AO 120 (Rev. 08/10)

TO:	REPORT ON THE FILING OR DETERMINATION OF AN ACTION REGARDING A PATENT OR TRADEMARK
Mail Stop 8 Director of the U.S. Patent and Trademark Office P.O. Box 1450 Alexandria, VA 22313-1450	

In Compliance with 35 U.S.C. § 290 and/or 15 U.S.C. § 1116 you are hereby advised that a court action has been filed in the U.S. District Court for the Western District of Texas on the following

☐ Trademarks or ☑ Patents. (☐ the patent action involves 35 U.S.C. § 292.):

DOCKET NO. 6-22-cv-299	DATE FILED 3/21/2022	U.S. DISTRICT COURT for the Western District of Texas
PLAINTIFF International Business Machines Corporation		DEFENDANT LzLabs GmbH Texas Wormhole, LLC

	PATENT OR TRADEMARK NO.	DATE OF PATENT OR TRADEMARK	HOLDER OF PATENT OR TRADEMARK
1	9,804,823	10/31/2017	International Business Machines Corporation
2	8,190,664	5/29/2012	International Business Machines Corporation
3	9,235,420	1/12/2016	International Business Machines Corporation
4	8,713,289	4/29/2014	International Business Machines Corporation
5	7,434,209	10/7/2009	International Business Machines Corporation

In the above—entitled case, the following patent(s)/ trademark(s) have been included:

DATE INCLUDED	INCLUDED BY ☐ Amendment ☐ Answer ☐ Cross Bill ☐ Other Pleading	
PATENT OR TRADEMARK NO.	DATE OF PATENT OR TRADEMARK	HOLDER OF PATENT OR TRADEMARK
1		
2		
3		
4		
5		

In the above—entitled case, the following decision has been rendered or judgement issued:

DECISION/JUDGEMENT

CLERK	(BY) DEPUTY CLERK	DATE

Copy 1—Upon initiation of action, mail this copy to Director Copy 3—Upon termination of action, mail this copy to Director
Copy 2—Upon filing document adding patent(s), mail this copy to Director Copy 4—Case file copy

AO 120 (Rev. 08/10)

TO:	REPORT ON THE FILING OR DETERMINATION OF AN ACTION REGARDING A PATENT OR TRADEMARK
Mail Stop 8 Director of the U.S. Patent and Trademark Office P.O. Box 1450 Alexandria, VA 22313-1450	

In Compliance with 35 U.S.C. § 290 and/or 15 U.S.C. § 1116 you are hereby advised that a court action has been filed in the U.S. District Court for the Western District of Texas on the following

☐ Trademarks or ☑ Patents. (☐ the patent action involves 35 U.S.C. § 292.):

DOCKET NO.	DATE FILED	U.S. DISTRICT COURT
6-22-cv-299	3/21/2022	for the Western District of Texas

PLAINTIFF	DEFENDANT
International Business Machines Corporation	LzLabs GmbH Texas Wormhole, LLC

	PATENT OR TRADEMARK NO.	DATE OF PATENT OR TRADEMARK	HOLDER OF PATENT OR TRADEMARK
1	9,804,823	10/31/2017	International Business Machines Corporation
2	8,190,664	5/29/2012	International Business Machines Corporation
3	9,235,420	1/12/2016	International Business Machines Corporation
4	8,713,289	4/29/2014	International Business Machines Corporation
5	7,434,209	10/7/2009	International Business Machines Corporation

In the above—entitled case, the following patent(s)/ trademark(s) have been included:

DATE INCLUDED	INCLUDED BY ☐ Amendment ☐ Answer ☐ Cross Bill ☐ Other Pleading	
PATENT OR TRADEMARK NO.	DATE OF PATENT OR TRADEMARK	HOLDER OF PATENT OR TRADEMARK
1		
2		
3		
4		
5		

In the above—entitled case, the following decision has been rendered or judgement issued:

DECISION/JUDGEMENT

CLERK	(BY) DEPUTY CLERK	DATE

Copy 1—Upon initiation of action, mail this copy to Director Copy 3—Upon termination of action, mail this copy to Director
Copy 2—Upon filing document adding patent(s), mail this copy to Director Copy 4—Case file copy

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	ISSUE DATE	PATENT NO.	ATTORNEY DOCKET NO.	CONFIRMATION NO.
11/668,623	04/29/2014	8713289	POU920060152US1	4180

33558 7590 04/09/2014

INTERNATIONAL BUSINESS MACHINES CORPORATION
Richard Lau
IPLAW DEPARTMENT / Bldg 008-2
2455 SOUTH ROAD - MS P386
POUGHKEEPSIE, NY 12601

ISSUE NOTIFICATION

The projected patent number and issue date are specified above.

Determination of Patent Term Adjustment under 35 U.S.C. 154 (b)
(application filed on or after May 29, 2000)

The Patent Term Adjustment is 1187 day(s). Any patent to issue from the above-identified application will include an indication of the adjustment on the front page.

If a Continued Prosecution Application (CPA) was filed in the above-identified application, the filing date that determines Patent Term Adjustment is the filing date of the most recent CPA.

Applicant will be able to obtain more detailed information by accessing the Patent Application Information Retrieval (PAIR) WEB site (http://pair.uspto.gov).

Any questions regarding the Patent Term Extension or Adjustment determination should be directed to the Office of Patent Legal Administration at (571)-272-7702. Questions relating to issue and publication fee payments should be directed to the Application Assistance Unit (AAU) of the Office of Data Management (ODM) at (571)-272-4200.

APPLICANT(s) (Please see PAIR WEB site http://pair.uspto.gov for additional applicants):

Reid T. Copeland, Toronto, CANADA;
Patrick R. Doyle, Toronto, CANADA;
Charles B. Hall, Calgary, CANADA;
Andrew Johnson, Hampshire, UNITED KINGDOM;
Ali I. Sheikh, Toronto, CANADA;

The United States represents the largest, most dynamic marketplace in the world and is an unparalleled location for business investment, innovation, and commercialization of new technologies. The USA offers tremendous resources and advantages for those who invest and manufacture goods here. Through SelectUSA, our nation works to encourage and facilitate business investment. To learn more about why the USA is the best country in the world to develop technology, manufacture products, and grow your business, visit SelectUSA.gov.

LzLabs GmbH. Ex. 1004-3

PART B - FEE(S) TRANSMITTAL

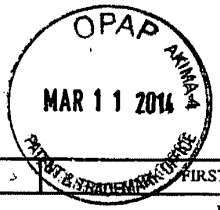
BES/jb 0560.669

Complete and send this form, together with applicable fee(s), to: Mail **Mail Stop ISSUE FEE**
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450
or Fax (571)-273-2885

INSTRUCTIONS: This form should be used for transmitting the ISSUE FEE and PUBLICATION FEE (if required). Blocks 1 through 5 should be completed where appropriate. All further correspondence including the Patent, advance orders and notification of maintenance fees will be mailed to the current correspondence address as indicated unless corrected below or directed otherwise in Block 1, by (a) specifying a new correspondence address; and/or (b) indicating a separate "FEE ADDRESS" for maintenance fee notifications.

CURRENT CORRESPONDENCE ADDRESS (Note: Use Block 1 for any change of address)

46369 7590 12/12/2013
HESLIN ROTHENBERG FARLEY & MESITI P.C.
5 COLUMBIA CIRCLE
ALBANY, NY 12203



Note: A certificate of mailing can only be used for domestic mailings of the Fee(s) Transmittal. This certificate cannot be used for any other accompanying papers. Each additional paper, such as an assignment or formal drawing, must have its own certificate of mailing or transmission.

Certificate of Mailing or Transmission
I hereby certify that this Fee(s) Transmittal is being deposited with the United States Postal Service with sufficient postage for first class mail in an envelope addressed to the Mail Stop ISSUE FEE address above, or being facsimile transmitted to the USPTO (571) 273-2885, on the date indicated below.

(Depositor's name)
(Signature)
(Date)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
11/668,623	01/30/2007	Reid T. Copeland	POU920060152US1	4180

TITLE OF INVENTION: EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS WITHOUT EXECUTING BRANCH INSTRUCTIONS

APPLN. TYPE	ENTITY STATUS	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	UNDISCOUNTED	$1780	$300	$0	$2080	03/12/2014

EXAMINER	ART UNIT	CLASS-SUBCLASS
HUISMAN, DAVID J	2183	712-220000

1. Change of correspondence address or indication of "Fee Address" (37 CFR 1.363).
☒ Change of correspondence address (or Change of Correspondence Address form PTO/SB/122) attached.
☐ "Fee Address" indication (or "Fee Address" Indication form PTO/SB/47; Rev 03-02 or more recent) attached. **Use of a Customer Number is required.**

2. For printing on the patent front page, list
(1) the names of up to 3 registered patent attorneys or agents OR, alternatively,
(2) the name of a single firm (having as a member a registered attorney or agent) and the names of up to 2 registered patent attorneys or agents. If no name is listed, no name will be printed.

1 Steven Chiu, Esq.
2 Blanche E. Schiller, Esq.
3 Heslin Rothenberg Farley & Mesiti P.C.

3. ASSIGNEE NAME AND RESIDENCE DATA TO BE PRINTED ON THE PATENT (print or type)

PLEASE NOTE: Unless an assignee is identified below, no assignee data will appear on the patent. If an assignee is identified below, the document has been filed for recordation as set forth in 37 CFR 3.11. Completion of this form is NOT a substitute for filing an assignment.

(A) NAME OF ASSIGNEE: INTERNATIONAL BUSINESS MACHINES CORPORATION

(B) RESIDENCE: (CITY and STATE OR COUNTRY): Armonk, New York

Please check the appropriate assignee category or categories (will not be printed on the patent) : ☐ Individual ☒ Corporation or other private group entity ☐ Government

4a. The following fee(s) are submitted:
☒ Issue Fee
☒ Publication Fee (No small entity discount permitted)
☐ Advance Order - # of Copies _____

4b. Payment of Fee(s): **(Please first reapply any previously paid issue fee shown above)**
☐ A check is enclosed.
☐ Payment by credit card. Form PTO-2038 is attached.
☒ The Director is hereby authorized to charge the required fee(s), any deficiency, or credit any overpayment, to Deposit Account Number 090463 (enclose an extra copy of this form).

03/12/2014 SMOHAMM1 00000023 090463 11668623
01 FC:1501 960.00 DA

PTOL-85 (Rev. 02/11)

LzLabs GmbH. Ex. 1004-4

5. **Change in Entity Status** (from status indicated above)

☐ Applicant certifying micro entity status. See 37 CFR 1.29

NOTE: Absent a valid certification of Micro Entity Status (see form PTO/SB/15A and 15B), issue fee payment in the micro entity amount will not be accepted at the risk of application abandonment.

☐ Applicant asserting small entity status. See 37 CFR 1.27

NOTE: If the application was previously under micro entity status, checking this box will be taken to be a notification of loss of entitlement to micro entity status.

☐ Applicant changing to regular undiscounted fee status.

NOTE: Checking this box will be taken to be a notification of loss of entitlement to small or micro entity status, as applicable.

NOTE: The Issue Fee and Publication Fee (if required) will not be accepted from anyone other than the applicant; a registered attorney or agent; or the assignee or other party in interest as shown by the records of the United States Patent and Trademark Office.

Authorized Signature Blanche E. Schiller Date January 2, 2014

Typed or printed name Blanche E. Schiller, Esq. Registration No. 35670

This collection of information is required by 37 CFR 1.311. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, Virginia 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450.

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

LzLabs GmbH. Ex. 1004-5

OPAP AKIMA-4
MAR 11 2014
PATENT & TRADEMARK OFFICE

PTO/SB/122 (11-08)
Approved for use through 11/30/2011. OMB 0651-0035
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

CHANGE OF CORRESPONDENCE ADDRESS
Application

Address to:
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Application Number	11/668,623
Filing Date	01/30/2007
First Named Inventor	Reid T. Copeland
Art Unit	2183
Examiner Name	David J. Huisman
Attorney Docket Number	POU920060152US1

Please change the Correspondence Address for the above-identified patent application to:

[X] The address associated with Customer Number: 33558

OR

[ ] Firm or Individual Name

Address

City	State	Zip

Country

Telephone	Email

This form cannot be used to change the data associated with a Customer Number. To change the data associated with an existing Customer Number use "Request for Customer Number Data Change" (PTO/SB/124).

I am the:

[ ] Applicant/Inventor

[ ] Assignee of record of the entire interest.
Statement under 37 CFR 3.73(b) is enclosed. (Form PTO/SB/96).

[X] Attorney or agent of record. Registration Number 35670.

[ ] Registered practitioner named in the application transmittal letter in an application without an executed oath or declaration. See 37 CFR 1.33(a)(1). Registration Number_____.

Signature Blanche E Schiller

Typed or Printed Name Blanche E. Schiller, Esq.

Date January 2, 2014 Telephone (518) 452-5600

NOTE: Signatures of all the inventors or assignees of record of the entire interest or their representative(s) are required. Submit multiple forms if more than one signature is required, see below*.

[ ] *Total of _____ forms are submitted.

This collection of information is required by 37 CFR 1.33. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 3 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

NOTICE OF ALLOWANCE AND FEE(S) DUE

46369 7590 12/12/2013

HESLIN ROTHENBERG FARLEY & MESITI P.C.
5 COLUMBIA CIRCLE
ALBANY, NY 12203

EXAMINER
HUISMAN, DAVID J

ART UNIT	PAPER NUMBER
2183	

DATE MAILED: 12/12/2013

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
11/668,623	01/30/2007	Reid T. Copeland	POU920060152US1	4180

TITLE OF INVENTION: EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS WITHOUT EXECUTING BRANCH INSTRUCTIONS

APPLN. TYPE	ENTITY STATUS	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	UNDISCOUNTED	$1780	$300	$0	$2080	03/12/2014

THE APPLICATION IDENTIFIED ABOVE HAS BEEN EXAMINED AND IS ALLOWED FOR ISSUANCE AS A PATENT. PROSECUTION ON THE MERITS IS CLOSED. THIS NOTICE OF ALLOWANCE IS NOT A GRANT OF PATENT RIGHTS. THIS APPLICATION IS SUBJECT TO WITHDRAWAL FROM ISSUE AT THE INITIATIVE OF THE OFFICE OR UPON PETITION BY THE APPLICANT. SEE 37 CFR 1.313 AND MPEP 1308.

THE ISSUE FEE AND PUBLICATION FEE (IF REQUIRED) MUST BE PAID WITHIN THREE MONTHS FROM THE MAILING DATE OF THIS NOTICE OR THIS APPLICATION SHALL BE REGARDED AS ABANDONED. THIS STATUTORY PERIOD CANNOT BE EXTENDED. SEE 35 U.S.C. 151. THE ISSUE FEE DUE INDICATED ABOVE DOES NOT REFLECT A CREDIT FOR ANY PREVIOUSLY PAID ISSUE FEE IN THIS APPLICATION. IF AN ISSUE FEE HAS PREVIOUSLY BEEN PAID IN THIS APPLICATION (AS SHOWN ABOVE), THE RETURN OF PART B OF THIS FORM WILL BE CONSIDERED A REQUEST TO REAPPLY THE PREVIOUSLY PAID ISSUE FEE TOWARD THE ISSUE FEE NOW DUE.

HOW TO REPLY TO THIS NOTICE:

I. Review the ENTITY STATUS shown above. If the ENTITY STATUS is shown as SMALL or MICRO, verify whether entitlement to that entity status still applies.

If the ENTITY STATUS is the same as shown above, pay the TOTAL FEE(S) DUE shown above.

If the ENTITY STATUS is changed from that shown above, on PART B - FEE(S) TRANSMITTAL, complete section number 5 titled "Change in Entity Status (from status indicated above)".

For purposes of this notice, small entity fees are 1/2 the amount of undiscounted fees, and micro entity fees are 1/2 the amount of small entity fees.

II. PART B - FEE(S) TRANSMITTAL, or its equivalent, must be completed and returned to the United States Patent and Trademark Office (USPTO) with your ISSUE FEE and PUBLICATION FEE (if required). If you are charging the fee(s) to your deposit account, section "4b" of Part B - Fee(s) Transmittal should be completed and an extra copy of the form should be submitted. If an equivalent of Part B is filed, a request to reapply a previously paid issue fee must be clearly made, and delays in processing may occur due to the difficulty in recognizing the paper as an equivalent of Part B.

III. All communications regarding this application must give the application number. Please direct all communications prior to issuance to Mail Stop ISSUE FEE unless advised to the contrary.

IMPORTANT REMINDER: Utility patents issuing on applications filed on or after Dec. 12, 1980 may require payment of maintenance fees. It is patentee's responsibility to ensure timely payment of maintenance fees when due.

LzLabs GmbH. Ex. 1004-7

PART B - FEE(S) TRANSMITTAL

Complete and send this form, together with applicable fee(s), to: Mail **Mail Stop ISSUE FEE**
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450
or Fax (571)-273-2885

INSTRUCTIONS: This form should be used for transmitting the ISSUE FEE and PUBLICATION FEE (if required). Blocks 1 through 5 should be completed where appropriate. All further correspondence including the Patent, advance orders and notification of maintenance fees will be mailed to the current correspondence address as indicated unless corrected below or directed otherwise in Block 1, by (a) specifying a new correspondence address; and/or (b) indicating a separate "FEE ADDRESS" for maintenance fee notifications.

CURRENT CORRESPONDENCE ADDRESS (Note: Use Block 1 for any change of address)

46369 7590 12/12/2013
HESLIN ROTHENBERG FARLEY & MESITI P.C.
5 COLUMBIA CIRCLE
ALBANY, NY 12203

Note: A certificate of mailing can only be used for domestic mailings of the Fee(s) Transmittal. This certificate cannot be used for any other accompanying papers. Each additional paper, such as an assignment or formal drawing, must have its own certificate of mailing or transmission.

Certificate of Mailing or Transmission

I hereby certify that this Fee(s) Transmittal is being deposited with the United States Postal Service with sufficient postage for first class mail in an envelope addressed to the Mail Stop ISSUE FEE address above, or being facsimile transmitted to the USPTO (571) 273-2885, on the date indicated below.

(Depositor's name)
(Signature)
(Date)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
11/668,623	01/30/2007	Reid T. Copeland	POU920060152US1	4180

TITLE OF INVENTION: EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS WITHOUT EXECUTING BRANCH INSTRUCTIONS

APPLN. TYPE	ENTITY STATUS	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	UNDISCOUNTED	$1780	$300	$0	$2080	03/12/2014

EXAMINER	ART UNIT	CLASS-SUBCLASS
HUISMAN, DAVID J	2183	712-220000

1. Change of correspondence address or indication of "Fee Address" (37 CFR 1.363).

❑ Change of correspondence address (or Change of Correspondence Address form PTO/SB/122) attached.

❑ "Fee Address" indication (or "Fee Address" Indication form PTO/SB/47; Rev 03-02 or more recent) attached. **Use of a Customer Number is required.**

2. For printing on the patent front page, list

(1) the names of up to 3 registered patent attorneys or agents OR, alternatively,

(2) the name of a single firm (having as a member a registered attorney or agent) and the names of up to 2 registered patent attorneys or agents. If no name is listed, no name will be printed.

1 _____
2 _____
3 _____

3. ASSIGNEE NAME AND RESIDENCE DATA TO BE PRINTED ON THE PATENT (print or type)

PLEASE NOTE: Unless an assignee is identified below, no assignee data will appear on the patent. If an assignee is identified below, the document has been filed for recordation as set forth in 37 CFR 3.11. Completion of this form is NOT a substitute for filing an assignment.

(A) NAME OF ASSIGNEE (B) RESIDENCE: (CITY and STATE OR COUNTRY)

Please check the appropriate assignee category or categories (will not be printed on the patent) : ❑ Individual ❑ Corporation or other private group entity ❑ Government

4a. The following fee(s) are submitted:

❑ Issue Fee

❑ Publication Fee (No small entity discount permitted)

❑ Advance Order - # of Copies _____

4b. Payment of Fee(s): **(Please first reapply any previously paid issue fee shown above)**

❑ A check is enclosed.

❑ Payment by credit card. Form PTO-2038 is attached.

❑ The Director is hereby authorized to charge the required fee(s), any deficiency, or credit any overpayment, to Deposit Account Number _____ (enclose an extra copy of this form).

LzLabs GmbH. Ex. 1004-8

5. **Change in Entity Status** (from status indicated above)

❑ Applicant certifying micro entity status. See 37 CFR 1.29	NOTE: Absent a valid certification of Micro Entity Status (see form PTO/SB/15A and 15B), issue fee payment in the micro entity amount will not be accepted at the risk of application abandonment.
❑ Applicant asserting small entity status. See 37 CFR 1.27	NOTE: If the application was previously under micro entity status, checking this box will be taken to be a notification of loss of entitlement to micro entity status.
❑ Applicant changing to regular undiscounted fee status.	NOTE: Checking this box will be taken to be a notification of loss of entitlement to small or micro entity status, as applicable.

NOTE: The Issue Fee and Publication Fee (if required) will not be accepted from anyone other than the applicant; a registered attorney or agent; or the assignee or other party in interest as shown by the records of the United States Patent and Trademark Office.

Authorized Signature _____ Date _____

Typed or printed name _____ Registration No. _____

This collection of information is required by 37 CFR 1.311. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, Virginia 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450.

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

LzLabs GmbH. Ex. 1004-9

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
11/668,623	01/30/2007	Reid T. Copeland	POU920060152US1	4180

46369 7590 12/12/2013

HESLIN ROTHENBERG FARLEY & MESITI P.C.
5 COLUMBIA CIRCLE
ALBANY, NY 12203

EXAMINER	
HUISMAN, DAVID J	

ART UNIT	PAPER NUMBER
2183	

DATE MAILED: 12/12/2013

Determination of Patent Term Adjustment under 35 U.S.C. 154 (b)
(application filed on or after May 29, 2000)

The Patent Term Adjustment to date is 1187 day(s). If the issue fee is paid on the date that is three months after the mailing date of this notice and the patent issues on the Tuesday before the date that is 28 weeks (six and a half months) after the mailing date of this notice, the Patent Term Adjustment will be 1187 day(s).

If a Continued Prosecution Application (CPA) was filed in the above-identified application, the filing date that determines Patent Term Adjustment is the filing date of the most recent CPA.

Applicant will be able to obtain more detailed information by accessing the Patent Application Information Retrieval (PAIR) WEB site (http://pair.uspto.gov).

Any questions regarding the Patent Term Extension or Adjustment determination should be directed to the Office of Patent Legal Administration at (571)-272-7702. Questions relating to issue and publication fee payments should be directed to the Customer Service Center of the Office of Patent Publication at 1-(888)-786-0101 or (571)-272-4200.

LzLabs GmbH. Ex. 1004-10

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether disclosure of these records is required by the Freedom of Information Act.
2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.
3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.
4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).
5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.
6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).
7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.
8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspection or an issued patent.
9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

Notices of Allowance and Fee(s) Due mailed between October 1, 2013 and December 31, 2013

(Addendum to PTOL-85)

If the "Notice of Allowance and Fee(s) Due" has a mailing date on or after October 1, 2013 and before January 1, 2014, the following information is applicable to this application.

If the issue fee is being timely paid on or after January 1, 2014, the amount due is the issue fee and publication fee in effect January 1, 2014. On January 1, 2014, the issue fees set forth in 37 CFR 1.18 decrease significantly and the publication fee set forth in 37 CFR 1.18(d)(1) decreases to $0.

If an issue fee or publication fee has been previously paid in this application, applicant is not entitled to a refund of the difference between the amount paid and the amount in effect on January 1, 2014.

Notice of Allowability	Application No. 11/668,623	Applicant(s) COPELAND ET AL.	
	Examiner David J. Huisman	Art Unit 2183	AIA (First Inventor to File) Status No

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☒ This communication is responsive to *the RCE filed on November 8, 2013*.
   ☐ A declaration(s)/affidavit(s) under **37 CFR 1.130(b)** was/were filed on_____.
2. ☐ An election was made by the applicant in response to a restriction requirement set forth during the interview on _____; the restriction requirement and election have been incorporated into this action.
3. ☒ The allowed claim(s) is/are *9-16, 18-22, 24-27, and 29-31 (hereafter collectively renumbered as claims 1-20)*. As a result of the allowed claim(s), you may be eligible to benefit from the **Patent Prosecution Highway** program at a participating intellectual property office for the corresponding application. For more information, please see http://www.uspto.gov/patents/init_events/pph/index.jsp or send an inquiry to PPHfeedback@uspto.gov .
4. ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

   Certified copies:

   a) ☐ All b) ☐ Some *c) ☐ None of the:
   1. ☐ Certified copies of the priority documents have been received.
   2. ☐ Certified copies of the priority documents have been received in Application No. _____ .
   3. ☐ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

   * Certified copies not received: _____.

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.
THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.

5. ☐ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
   ☐ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date _____.

   Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).
6. ☐ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Attachment(s)

1. ☐ Notice of References Cited (PTO-892)
2. ☒ Information Disclosure Statements (PTO/SB/08), Paper No./Mail Date 11/8/2013
3. ☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material
4. ☐ Interview Summary (PTO-413), Paper No./Mail Date _____ .
5. ☐ Examiner's Amendment/Comment
6. ☐ Examiner's Statement of Reasons for Allowance
7. ☐ Other _____.

/David J. Huisman/
Primary Examiner, Art Unit 2183

LzLabs GmbH. Ex. 1004-13

Issue Classification	Application/Control No.	Applicant(s)/Patent Under Reexamination
	11668623	COPELAND ET AL.
	Examiner	**Art Unit**
	DAVID J HUISMAN	2183

CPC		
Symbol	**Type**	**Version**

CPC Combination Sets				
Symbol	**Type**	**Set**	**Ranking**	**Version**

NONE		Total Claims Allowed:	
(Assistant Examiner)	(Date)	20	
/DAVID J HUISMAN/ Primary Examiner.Art Unit 2183	11/16/2013	O.G. Print Claim(s)	O.G. Print Figure
(Primary Examiner)	(Date)	1	2

LzLabs GmbH. Ex. 1004-14

Issue Classification	Application/Control No.	Applicant(s)/Patent Under Reexamination
	11668623	COPELAND ET AL.
	Examiner	**Art Unit**
	DAVID J HUISMAN	2183

US ORIGINAL CLASSIFICATION		INTERNATIONAL CLASSIFICATION										
CLASS	SUBCLASS	CLAIMED					NON-CLAIMED					
712	220	G	0	6	F	9 / 455 (2006.0)						
CROSS REFERENCE(S)												
CLASS	SUBCLASS (ONE SUBCLASS PER BLOCK)											
703	26											

NONE		**Total Claims Allowed:**	
(Assistant Examiner)	(Date)	20	
/DAVID J HUISMAN/ Primary Examiner.Art Unit 2183	11/16/2013	O.G. Print Claim(s)	O.G. Print Figure
(Primary Examiner)	(Date)	1	2

LzLabs GmbH. Ex. 1004-15

Issue Classification	Application/Control No.	Applicant(s)/Patent Under Reexamination
	11668623	COPELAND ET AL.
	Examiner	**Art Unit**
	DAVID J HUISMAN	2183

☐ Claims renumbered in the same order as presented by applicant ☐ CPA ☐ T.D. ☐ R.1.47

Final	Original	Final	Original	Final	Original	Final	Original	Final	Original	Final	Original	Final	Original	Final	Original
1	9	19	25												
2	10	20	26												
3	11	6	27												
4	12		28												
5	13	7	29												
9	14	8	30												
11	15	10	31												
12	16														
	17														
13	18														
14	19														
15	20														
16	21														
17	22														
	23														
18	24														

NONE		**Total Claims Allowed:**	
(Assistant Examiner)	(Date)	20	
/DAVID J HUISMAN/ Primary Examiner.Art Unit 2183	11/16/2013	O.G. Print Claim(s)	O.G. Print Figure
(Primary Examiner)	(Date)	1	2

LzLabs GmbH. Ex. 1004-16

Receipt date: 11/08/2013

SUPPLEMENTAL INFORMATION DISCLOSURE STATEMENT BY APPLICANT
(Not for submission under 37 CFR 1.99))

Application Number	11/668,623 11668623 - GAU: 2183
Filing Date	01/30/2007
First Named Inventor	Reid T. Copeland
Art Unit	2183
Examiner Name	David J. Huisman
Attorney Docket	POU920060152US1

U.S. PATENTS

Examiner Initial*	Cite No	Patent Number	Kind Code[1]	Issue Date	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines where Relevant Passages or Relevant Figures Appear

U.S. PATENT APPLICATION PUBLICATIONS

Examiner Initial*	Cite No	Publication Number	Kind Code[1]	Publication Date	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines where Relevant Passages or Relevant Figures Appear

FOREIGN PATENT DOCUMENTS

Examiner Initial*	Cite No	Publication Number[3]	Country Code[2]	Kind Code[4]	Publication Date	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines where Relevant Passages or Relevant Figures Appear	T[5]
/DJH/	B1	JPS58195963	JP	A	1983-11-15	Hitachi Ltd.		
/DJH/	B2	JP2004152201	JP	A	2004-05-27	Mitsubishi Electric Corp.		

NON-PATENT LITERATURE DOCUMENTS

Examiner Initial*	Cite No	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc), date, page(s), volume-issue number(s), publisher, city and/or country where published.	T[5]

EXAMINER SIGNATURE

Examiner Signature	/David J. Huisman/	Date Considered	11/16/2013

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through a citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [2] Enter office that issued the document, by the two-letter code (WIPO) Standard ST.3). [3] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[4] Kind of document by the appropriate symbols as indicate on the document under WIPO Standard ST.16 if possible. [5] Applicant is to place a check mark here if English language translation is attached.

LzLabs GmbH. Ex. 1004-17

EAST Search History

EAST Search History (Prior Art)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	1466	712/220.ccls.	US-PGPUB; USPAT	OR	ON	2013/11/16 15:13
L2	630	703/26.ccls.	US-PGPUB; USPAT	OR	ON	2013/11/16 15:13
S42	1	"20030093649".pn.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S43	86	emulat$3 with "condition code"	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S44	102	emulat$3 with ("condition code" "flag bit" "status bit") and (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S45	29	emulat$3 with ("condition code" "flag bit" "status bit") same branch and (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S46	68	("20020026633" | "4514803" | "4812975" | "5313614" | "5329471" | "5408637" | "5440702" | "5551015" | "5574873" | "5574927" | "5577233" | "5598560" | "5606696" | "5649203" | "5724590" | "5737631" | "5751982" | "5768593" | "5805792" | "5805895" | "5870575" | "5890000" | "5901308" | "5903760" | "5925124" | "5930509" | "5946491" | "5956495" | "5966539" | "5983335" | "6000028" | "6009261" | "6031994" | "6076155" | "6077314" | "6091897" | "6105124" | "6126329" | "6170083" | "6185675" | "6199095" | "6205545" | "6223339" | "6301705" | "6308318" | "6321379" | "6330691" | "6360194" | "6367067" | "6427234" | "6434741" | "6463582" | "6477641" | "6502237" | "6505296" | "6522985" | "6523173" | "6526572" | "6530078" | "6530079" | "6535903" | "6539471" | "6539541" | "6549987" | "6564297" | "6594824" | "6631514" | "6643630").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S47	6	(emulat$3 translat$3) same ("condition code" "flag bit" "status bit") same (without avoid$4) near2 branch$3 and (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S48	144	(emulat$3 translat$3) with (without avoid$4) near2 branch$3 and (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S49	12	(emulat$3 translat$3) with (without avoid$4) near2 branch$3 and (overflow carry sign) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S50	5	("4951195" | "5241678" | "5301325" | "5313614" | "5598560").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-18

S51	28	("5903760").URPN.	USPAT	OR	ON	2009/03/16 16:40
S52	68	("20020026633" | "4514803" | "4812975" | "5313614" | "5329471" | "5408637" | "5440702" | "5551015" | "5574873" | "5574927" | "5577233" | "5598560" | "5606696" | "5649203" | "5724590" | "5737631" | "5751982" | "5768593" | "5805792" | "5805895" | "5870575" | "5890000" | "5901308" | "5903760" | "5925124" | "5930509" | "5946491" | "5956495" | "5966539" | "5983335" | "6000028" | "6009261" | "6031994" | "6076155" | "6077314" | "6091897" | "6105124" | "6126329" | "6170083" | "6185675" | "6199095" | "6205545" | "6223339" | "6301705" | "6308318" | "6321379" | "6330691" | "6360194" | "6367067" | "6427234" | "6434741" | "6463582" | "6477641" | "6502237" | "6505296" | "6522985" | "6523173" | "6526572" | "6530078" | "6530079" | "6535903" | "6539471" | "6539541" | "6549987" | "6564297" | "6594824" | "6631514" | "6643630").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S53	68	("20020026633" | "4514803" | "4812975" | "5313614" | "5329471" | "5408637" | "5440702" | "5551015" | "5574873" | "5574927" | "5577233" | "5598560" | "5606696" | "5649203" | "5724590" | "5737631" | "5751982" | "5768593" | "5805792" | "5805895" | "5870575" | "5890000" | "5901308" | "5903760" | "5925124" | "5930509" | "5946491" | "5956495" | "5966539" | "5983335" | "6000028" | "6009261" | "6031994" | "6076155" | "6077314" | "6091897" | "6105124" | "6126329" | "6170083" | "6185675" | "6199095" | "6205545" | "6223339" | "6301705" | "6308318" | "6321379" | "6330691" | "6360194" | "6367067" | "6427234" | "6434741" | "6463582" | "6477641" | "6502237" | "6505296" | "6522985" | "6523173" | "6526572" | "6530078" | "6530079" | "6535903" | "6539471" | "6539541" | "6549987" | "6564297" | "6594824" | "6631514" | "6643630").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S54	723	(emulat$3 translat$3) with compar$4 with instruction	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S55	21	(emulat$3 translat$3) with compar$4 with instruction same (code flag) near2 (condition status bit)	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S56	412	712/209.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S57	487	712/221.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S58	867	712/227.ccls.	US-PGPUB; USPAT;	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-19

			USOCR			
S59	897	717/136.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S60	265	717/137.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S61	234	717/138.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S62	1	emulat$3 with ("condition code" "flag bit" "status bit") and (overflow carry sign)	USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2009/03/16 16:40
S63	1362	717/124.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S64	140	717/134.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S65	1560	(emulat$3 translat$3) same (arithmetic add$5 subtract$3) same (code flag) near2 (condition status bit)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S66	101	(emulat$3 translat$3) same (arithmetic add$5 subtract$3) with set$4 with (code flag) near2 (condition status bit)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S67	37	(emulat$3 translat$3) same (arithmetic add$5 subtract$3) with set$4 with (code flag) near2 (condition status bit) same (Z O S C sign zero overflow carry)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S68	29	(emulat$3 translat$3) same (arithmetic add$5 subtract$3) with set$4 with (code flag) near2 (condition status bit) same (Z O S C sign zero overflow carry) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S69	7	("20020046298" "20040176941" "20050015754" "20070011664" "5301302" "5678032" "5966515").pn.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S70	4	S69 and (code flag) near2 (condition status bit)	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S71	119	simulat$3 near4 (code flag) near2 (condition status bit)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S72	67	simulat$3 near4 (code flag) near2 (condition status bit) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S73	35	simulat$3 near4 (code flag) near2 (condition status bit) and (overflow carry) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S74	127	(emulat$3 translat$3) and (host target) with set$4 with (code flag) near2 (condition status bit) and ((overflow O) and (carry C) and (sign S negative N)) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S75	59	(emulat$3 translat$3) and (host target) with set$4 with (code flag) near2 (condition status bit) and ((overflow O) and (carry C) and (sign S negative N))	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-20

		and ("712"/$.ccls. "717"/$.ccls.)				
S76	32	emulat$3 and (host target) with set$4 with (code flag) near2 (condition status bit) and ((overflow O) and (carry C) and (sign S negative N)) and ("712"/$.ccls. "717"/$.ccls.)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S77	1	emulat$3 and (host target) with set$4 with (code flag) near2 (condition status bit) same ((overflow O) and (carry C) and (sign S negative N)) and ("712"/$.ccls. "717"/$.ccls.)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S78	1	emulat$3 and (host target) with set$4 with (code flag) near2 (condition status bit) same (((overflow O) and (carry C) and (sign S negative N))) and ("712"/$.ccls. "717"/$.ccls.)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S79	6	("4415969" | "5327537" | "5440749" | "5493687" | "5590359" | "5689720").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S80	2	("5774694").URPN.	USPAT	OR	ON	2009/03/16 16:40
S81	8	("5313614" | "5481693" | "5619664" | "5774694" | "5859999" | "5860017" | "5884059" | "6049864").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S82	1	"20060173931".pn.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:44
S83	446	703/26.ccls.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:46
S84	20	emulat$3 with ("condition code" "flag bit" "status bit") same (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/07/14 17:52
S85	0	emulat$3 with ("condition code" "flag bit" "status bit") same (overflow carry sign)	USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2009/07/14 17:53
S86	47	huisman.xp.	US-PGPUB; USPAT	OR	ON	2010/03/04 09:27
S87	1913	subtract$3 with carry with (bit flag)	US-PGPUB; USPAT	OR	ON	2010/03/04 11:03
S88	186	subtract$3 with set$4 with carry adj (bit flag)	US-PGPUB; USPAT	OR	ON	2010/03/04 11:03
S89	112	subtract$3 with set$4 with carry adj (bit flag) and "712"/$.ccls.	US-PGPUB; USPAT	OR	ON	2010/03/04 11:03
S90	2	("5574931" "5999976").pn.	US-PGPUB; USPAT	OR	ON	2010/03/04 15:04
S98	1	distinguish$3 with (outcome categor$4) with condition near2 (code flag)	US-PGPUB; USPAT	OR	ON	2012/07/03 14:35
S99	86	categor$4 with condition near2 (code flag)	US-PGPUB; USPAT	OR	ON	2012/07/03 14:35
S100	8	categor$4 with condition near2 (code flag) and "712"/$.ccls.	US-PGPUB; USPAT	OR	ON	2012/07/03 14:35
S102	8	categor$4 with condition near2 (code flag)	USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2012/07/03 14:37

LzLabs GmbH. Ex. 1004-21

S103	1297	712/220.ccls.	US-PGPUB; USPAT	OR	ON	2012/11/13 10:54
S104	591	703/26.ccls.	US-PGPUB; USPAT	OR	ON	2012/11/13 10:54
S105	93	(intermediate provisional temporary) near3 ("condition code" "flag register")	US-PGPUB; USPAT	OR	ON	2012/11/13 11:16
S106	35	(intermediate provisional temporary) near3 ("condition code" "flag register") and ("712"/$.ccls. "703"/$.ccls.)	US-PGPUB; USPAT	OR	ON	2012/11/13 11:28
S107	68	predict$3 near3 ("condition code" "flag register") and ("712"/$.ccls. "703"/$.ccls.)	US-PGPUB; USPAT	OR	ON	2012/11/13 11:40
S108	6	(intermediate provisional temporary) near3 ("condition code" "flag register")	USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2012/11/13 12:39
S109	0	predict$3 near3 ("condition code" "flag register") and ("712"/$.ccls. "703"/$.ccls.)	USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2012/11/13 12:42
S111	1409	712/220.ccls.	US-PGPUB; USPAT	OR	ON	2013/07/14 18:25
S112	623	703/26.ccls.	US-PGPUB; USPAT	OR	ON	2013/07/14 18:25

EAST Search History (Interference)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L3	16	((intermediate provisional temporary) near3 ("condition code" "flag register")).clm.	US-PGPUB; USPAT; UPAD	OR	ON	2013/11/16 15:13
S110	16	((intermediate provisional temporary) near3 ("condition code" "flag register")).clm.	US-PGPUB; USPAT; UPAD	OR	ON	2012/11/13 12:33
S113	16	((intermediate provisional temporary) near3 ("condition code" "flag register")).clm.	US-PGPUB; USPAT; UPAD	OR	ON	2013/07/14 18:26

11/16/2013 3:16:27 PM
C:\Users\dhuisman\Documents\EAST\Workspaces\11-668623.wsp

LzLabs GmbH. Ex. 1004-22

Search Notes	Application/Control No.	Applicant(s)/Patent Under Reexamination
	11668623	COPELAND ET AL.
	Examiner	Art Unit
	DAVID J HUISMAN	2183

CPC- SEARCHED		
Symbol	Date	Examiner

CPC COMBINATION SETS - SEARCHED		
Symbol	Date	Examiner

US CLASSIFICATION SEARCHED			
Class	Subclass	Date	Examiner
712	209, 221, 227	3-16-09	DJH
717	124, 134, 136, 137, 138	3-16-09	DJH
703	26	3-16-09	DJH
712	220	7-14-13	DJH
703	26	7-14-13	DJH
712	220	11-16-13	DJH
703	26	11-16-13	DJH

SEARCH NOTES		
Search Notes	Date	Examiner
EAST Search (all databases)	3/16/09	DJH
Inventor Name Search (PALM)	3/16/09	DJH
Updated EAST Search (all databases)	7-14-09	DJH
Updated EAST Search (all databases)	3-5-10	DJH
Updated EAST Search (all databases)	7-3-12	DJH
Updated EAST Search (all databases)	11-13-12	DJH
Inventor Name Search (PALM)	11-13-12	DJH
712/220 (cursory)	11-13-12	DJH
703/26 (cursory)	11-13-12	DJH
Inventor Name Search (PALM)	7-14-13	DJH
Inventor Name Search (PALM)	11-16-13	DJH

	/DAVID J HUISMAN/ Primary Examiner.Art Unit 2183

INTERFERENCE SEARCH			
US Class/ CPC Symbol	**US Subclass / CPC Group**	**Date**	**Examiner**
PGPub Text Search		11-13-12	DJH
PGPub Text Search		7-14-13	DJH
PGPub Text Search		11-16-13	DJH

	/DAVID J HUISMAN/ Primary Examiner.Art Unit 2183

LzLabs GmbH. Ex. 1004-24

Doc code: RCEX
Doc description: Request for Continued Examination (RCE)

PTO/SB/30EFS (07-09)
Approved for use through 07/31/2012. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

REQUEST FOR CONTINUED EXAMINATION(RCE)TRANSMITTAL
(Submitted Only via EFS-Web)

Application Number	11668623	Filing Date	2007-01-30	Docket Number (if applicable)	POU920060152US1	Art Unit	2183
First Named Inventor	Reid T. Copeland			Examiner Name	David J. Huisman		

This is a Request for Continued Examination (RCE) under 37 CFR 1.114 of the above-identified application.
Request for Continued Examination (RCE) practice under 37 CFR 1.114 does not apply to any utility or plant application filed prior to June 8, 1995, or to any design application. The Instruction Sheet for this form is located at WWW.USPTO.GOV

SUBMISSION REQUIRED UNDER 37 CFR 1.114

Note: If the RCE is proper, any previously filed unentered amendments and amendments enclosed with the RCE will be entered in the order in which they were filed unless applicant instructs otherwise. If applicant does not wish to have any previously filed unentered amendment(s) entered, applicant must request non-entry of such amendment(s).

☐ Previously submitted. If a final Office action is outstanding, any amendments filed after the final Office action may be considered as a submission even if this box is not checked.

☐ Consider the arguments in the Appeal Brief or Reply Brief previously filed on _____

☐ Other _____

☒ Enclosed

☐ Amendment/Reply

☒ Information Disclosure Statement (IDS)

☐ Affidavit(s)/ Declaration(s)

☒ Other Statement of Relevance

MISCELLANEOUS

☐ Suspension of action on the above-identified application is requested under 37 CFR 1.103(c) for a period of months _____ (Period of suspension shall not exceed 3 months; Fee under 37 CFR 1.17(i) required)

☐ Other _____

FEES

The RCE fee under 37 CFR 1.17(e) is required by 37 CFR 1.114 when the RCE is filed.

☒ The Director is hereby authorized to charge any underpayment of fees, or credit any overpayments, to Deposit Account No 090463

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT REQUIRED

☒ Patent Practitioner Signature

☐ Applicant Signature

LzLabs GmbH. Ex. 1004-25

PTO/SB/30EFS (07-09)
Approved for use through 07/31/2012. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Signature of Registered U.S. Patent Practitioner			
Signature	/Blanche E. Schiller/	Date (YYYY-MM-DD)	2013-11-08
Name	Blanche E. Schiller, Esq.	Registration Number	35670

This collection of information is required by 37 CFR 1.114. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450.

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

LzLabs GmbH. Ex. 1004-26

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these records.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

LzLabs GmbH. Ex. 1004-27

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant:	Reid T. Copeland et al.	Confirmation No.:	4180
Serial No.:	11/668,623	Group Art Unit:	2183
Filed:	01/30/2007	Examiner:	David J. Huisman
Title:	EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS WITHOUT EXECUTING BRANCH INSTRUCTIONS		

To: Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

STATEMENT OF RELEVANCE FOR INFORMATION DISCLOSED BY APPLICANT

Sir:

The following Statement of Relevance is submitted with the accompanying Supplemental Information Disclosure form.

Document Designation	Relevance
JPS58195963	Data Processor
JP2004152201	Processor Device

Full text copies of the art cited, or the pertinent portions thereof, are enclosed. It is respectfully requested that this art be considered by the Examiner in the above-entitled application and made of record therein. The information provided and references enclosed herewith shall not be construed as a representation that a search has been made or that no other art than that identified exists.

Respectfully submitted,

Blanche E. Schiller

Blanche E. Schiller
Attorney for Applicants
Registration No.: 35,670

Dated: November 8, 2013

HESLIN ROTHENBERG FARLEY & MESITI P.C.
5 Columbia Circle
Albany, NY 12203
Telephone: (518) 452-5600
Facsimile: (518) 452-5579

LzLabs GmbH. Ex. 1004-28

SUPPLEMENTAL INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99))		
	Application Number	11/668,623
	Filing Date	01/30/2007
	First Named Inventor	Reid T. Copeland
	Art Unit	2183
	Examiner Name	David J. Huisman
	Attorney Docket	POU920060152US1

U.S. PATENTS

Examiner Initial*	Cite No	Patent Number	Kind Code[1]	Issue Date	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines where Relevant Passages or Relevant Figures Appear

U.S. PATENT APPLICATION PUBLICATIONS

Examiner Initial*	Cite No	Publication Number	Kind Code[1]	Publication Date	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines where Relevant Passages or Relevant Figures Appear

FOREIGN PATENT DOCUMENTS

Examiner Initial*	Cite No	Publication Number[3]	Country Code[2]	Kind Code[4]	Publication Date	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines where Relevant Passages or Relevant Figures Appear	T[5]
	B1	JPS58195963	JP	A	1983-11-15	Hitachi Ltd.		
	B2	JP2004152201	JP	A	2004-05-27	Mitsubishi Electric Corp.		

NON-PATENT LITERATURE DOCUMENTS

Examiner Initial*	Cite No	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc), date, page(s), volume-issue number(s), publisher, city and/or country where published.	T[5]

EXAMINER SIGNATURE

Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through a citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [2] Enter office that issued the document, by the two-letter code (WIPO) Standard ST.3). [3] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document.

[4] Kind of document by the appropriate symbols as indicate on the document under WIPO Standard ST.16 if possible. [5] Applicant is to place a check mark here if English language translation is attached.

LzLabs GmbH. Ex. 1004-29

Espacenet

Bibliographic data: JPS58195963 (A) — 1983-11-15

DATA PROCESSOR

No documents available for this priority number.

Inventor(s): ADACHI SHIGEMI ± (ADACHI SHIGEMI)

Applicant(s): HITACHI LTD ± (HITACHI SEISAKUSHO KK)

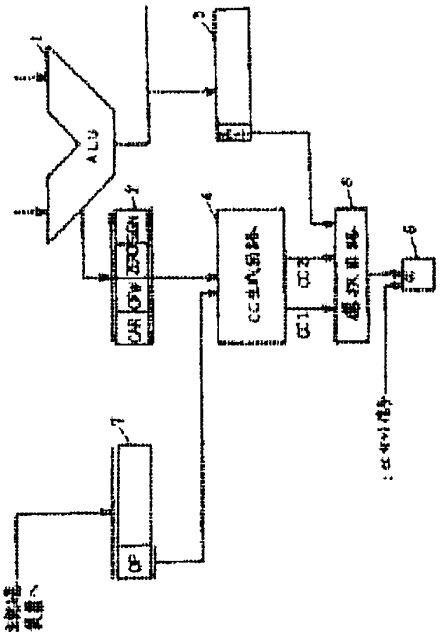
Classification: - **international:*G06F9/22; G06F9/30; G06F9/32; G06F9/455;*** (IPC1-7): G06F9/22; G06F9/30; G06F9/44
- **cooperative: G06F9/30094**

Application number: JP19820078215 19820512

Priority number (s): JP19820078215 19820512

Abstract of JPS58195963 (A)

PURPOSE:To set a condition code at a high speed by a hardware of a small quantity, by selecting one of plural condition codes generated by a kind of an instruction and a result of operation by a state control bit. CONSTITUTION:Each bit of a state flip-flop 2 set by a result of operation of an operator 1, and an instruction code (OP part) of an instruction register 7 are inputted to a condition code generating circuit 4. The condition code generating circuit 4 generates patterns CC1, CC2 of 2 kinds of condition codes of an emulation state and a non-emulation state corresponding to an input state, by an ROM pattern. The patterns of CC1 and CC2, and an EMLT bit in a control register 3 are inputted to a selecting circuit 5. Either one of the pattern CC1 or CC2 is selected by the EMLT fit, and it is set to a condition code flip- flop 6.



LzLabs GmbH. Ex. 1004-30

⑲ 日本国特許庁（JP）　　⑪ 特許出願公開

⑫ 公開特許公報（A）　　昭58—195963

⑤1Int. Cl.³	識別記号	庁内整理番号	⑬公開　昭和58年(1983)11月15日
G 06 F 9/30		7218—5B	
9/22		8120—5B	発明の数　1
9/44		7218—5B	審査請求　未請求

（全 4 頁）

⑤4データ処理装置

⑳特　　願　昭57—78215
㉒出　　願　昭57(1982)5月12日

㉒発 明 者　足立茂美
尾張旭市晴丘町池上1番地株式会社日立製作所旭工場内
㉑出 願 人　株式会社日立製作所
東京都千代田区丸の内1丁目5番1号
㉔代 理 人　弁理士　薄田利幸

明　細　書

1.発明の名称

データ処理装置

2.特許請求の範囲

(1) 命令を実行後、コンディションコードのセットを行うマイクロプログラム制御のデータ処理装置において、命令の種類、演算結果によつて複数のコンディションコードを生成し、マイクロプログラムによつてオン，オフ可能な状態制御ビットにより、前記生成した複数コンディションコードの一つを選択する手段を設け、装置状態によつてコンディションコードのセットを切換えることを特徴とするデータ処理装置。

3.発明の詳細な説明

発明の対象

本発明は、命令を実行後、コンディションコードのセットを行うマイクロプログラム制御のデータ処理装置に関し、特にあるデータ処理装置において、他のデータ処理装置の機械命令を実行させるエミュレーションに好適なコンディションコードのセット方式に関するものである。

従来技術

一般にデータ処理装置においては、ある命令を実行すると、その結果にもとづきコンディションコードがセットされるが、コンディションコードのパターンは、データ処理装置のアーキテクチャによつて異つているのが普通である。マイクロプログラム制御のデータ処理装置では、異るアーキテクチャのデータ処理装置の機械命令を実行するエミュレーションが可能であるが、コンディションコードのパターン発生方法がもとのデータ処理装置専用である場合には、エミュレーション時のコンディションコードのセットは困難となり、可能の場合にもマイクロプログラムの処理ステップ数が増加し、性能が出ないといつた問題がある。又、マイクロ命令の一部にコンディションコードのパターンの種類を指示するフィールドを設け、このフィールドのビットパターンによつて、複数個生成したコンディションコードのパターンの1つを選択するという方式も知られているが、融通

LzLabs GmbH. Ex. 1004-31

性がある反面マイクロ命令の上記フィールドを設けなければならないことや、該フィールドのデコーダを設けなければならないことなど、ハードウエア量が増加するという欠点があつた。

発明の目的

本発明の目的は、マイクロプログラム制御のデータ処理装置において、他データ処理装置のエミュレーションを行う場合にも、高速に、しかもハードウエアをあまり増加させることなく、コンディションコードをセットすることにある。

発明の総括的説明

一般に、マイクロプログラム制御のデータ処理装置において、コンディションコードをセットするには、機械命令の実行ステージでの演算結果によつてセットされるＺＥＲＯ，ＳＩＧＮ，ＣＡＲＲＹ等のビットと、処理している機械語の命令コードのビットパターンから、エンコード用ＲＯＭを用いて一意に生成したコンディションコードのパターンを、マイクロ命令中の１フィールドで指定されるコンディションコードセット命令に基づき、コンディションコード用レジスタにセットしている。この機構をできるだけそのまま用い、エミュレーション時の異るコンディションコードパターンをセットできる方法が望ましい。

一方、エミュレーションを実行する時には、エミュレーションを行うデータ処理装置を、エミュレーション専用の装置として使用する場合と、エミュレーションするプログラムと、自分自身の機械語で書かれたプログラムを割込み等の発生により交互に切換えて実行する場合がある。いずれの場合においても、エミュレーション実行状態と、非エミュレーション実行状態は、明確に区別されており、両状態はデータ処理装置のオペレーティングシステム，マイクロプログラムで管理される。

そこで、ハードウエアもこれに対応し、エミュレーション状態か否かを示す制御ビットを設け、両状態に応じて、マイクロプログラムによつて、該ビットをオン，オフできる機構を用意し、該ビットによつて、コンディションコードのセット方法を選択するようにすれば、わずかなハードウエア量の追加でエミュレーション時のコンディションコードのセットが可能である。

本発明では、機械語実行ステージで生成されるＺＥＲＯ，ＳＩＧＮ，ＣＡＲＲＹ等のビットと機械語の命令コードから生成するコンディションコードを、エミュレーション命令用と自分自身の命令用の２種類生成し、そのいずれかを上記エミュレーション状態を示すビットにより選択し、コンディションコードをセットするのである。エミュレーション状態の制御ビットのオン，オフは、データ処理装置内の他の制御ビットをオン，オフするハードウエアが使用できる。又、２種類のコンディションコードの生成についても、エンコード用ＲＯＭの容量増で解決可能であり、ハードウエア量の増加はわずかですむ。

発明の実施例

図は本発明の一実施例の構成図である。図中、１は演算器、２は演算結果によつて生成されるＣＡＲＲＹ，ＯＶＥＲＦＬＯＷ，ＺＥＲＯ，ＳＩＧＮを表示する状態フリップフロップである。３はデータ処理装置を制御するフリップフロップ類をまとめた制御レジスタで、この中のＥＭＬＴビットが、本発明によるエミュレーションモードであるか否かを表わすビット（１ビット）である。４はコンディションコード生成回路で、エミュレーション，非エミュレーション状態のそれぞれに対応したコンディションコードを生成する。５は生成された２種類のパターンのコンディションコードのいずれかを選択する選択回路、６は最終的にコンディションコードがセットされるコンディションコードフリップフロップ、７は主記憶装置から読み出された機械語を格納する命令レジスタである。

以下、本実施例における機械語の実行及びコンディションコードのセットについて説明する。まず、主記憶装置から命令レジスタ７に機械語が読み出され、該命令がデコードされて実行ステージに進み、演算器１によつて演算が行われる。なお、これに必要な構成は省略されている。この演算結果によつて、ＣＡＲＲＹ，ＺＥＲＯ，ＯＶＥＲＦ

LzLabs GmbH. Ex. 1004-32

LOW，SIGNの状態フリップフロップ2がセットされる。該状態フリップフロップ2の各ビット及び命令レジスタ7の命令コード（OP部）は、コンディションコード生成回路4に入力される。コンディションコード生成回路4では、あらかじめ設定されているROMパターンにより、入力状態に対応したエミュレーション状態と非エミュレーション状態の2種類のコンディションコードのパターンCC1，CC2を発生する。CC1，CC2のパターン及び制御レジスタ8中のEMLTビットは選択回路5に入力され、EMLTビットによって、CC1あるいはCC2のいずれかが選択される。制御レジスタ8は演算器1の出力バスに接続され、マイクロプログラムによって任意の値が設定できるものである。この中のEMLTビットは、エミュレーション状態，非エミュレーション状態の切換りに応じて、オン，オフされる。このようにして決定されたコンディションコードは、次のステップのマイクロ命令で出されるコンディションコードセット信号（CCセット信号）により、コンディションコードフリップフロップ6にセットされる。

以上のようにして、制御レジスタ8のEMLTビットを制御することで、エミュレーション状態，非エミュレーション状態に応じたコンディションコードを簡単にかつ、迅速にセットすることができる。

尚、上記説明では、コンディションコードは、2種類に限定したが、制御レジスタ8のEMLTビット，コンディションコード生成回路4を拡張すれば、更に多くのコンディションコードの種類をセットすることが可能である。又、制御レジスタ8のEMLTビットもコンディションコード生成回路4に入力し、一度に目的とするコンディションコードを出力するようにしてもよく、この場合、選択回路5が省略できる。

発明の効果

本発明によれば、異るアーキテクチャの計算機のエミュレーションを行う場合のコンディションコードのセットが、わずかなハードウエア量で、性能を損うことなく実現可能となり、エミュレーション実行の為の有効な方式である。

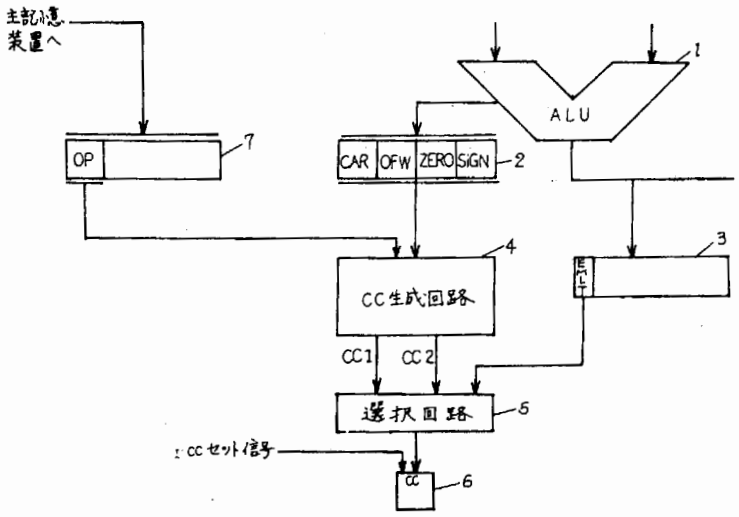
4図面の簡単な説明

図は本発明の一実施例の構成図である。

1…演算器、2…状態フリップフロップ、8…制御レジスタ、4…コンディションコード生成回路、5…選択回路、6…コンディションコードフリップフロップ、7…命令レジスタ。

代理人 弁理士 薄田利幸

LzLabs GmbH. Ex. 1004-33



LzLabs GmbH. Ex. 1004-34

Espacenet

Bibliographic data: JP2004152201 (A) — 2004-05-27

PROCESSOR DEVICE

No documents available for this priority number.

Inventor(s): ITO TAKEHIRO ± (ITO TAKEHIRO)

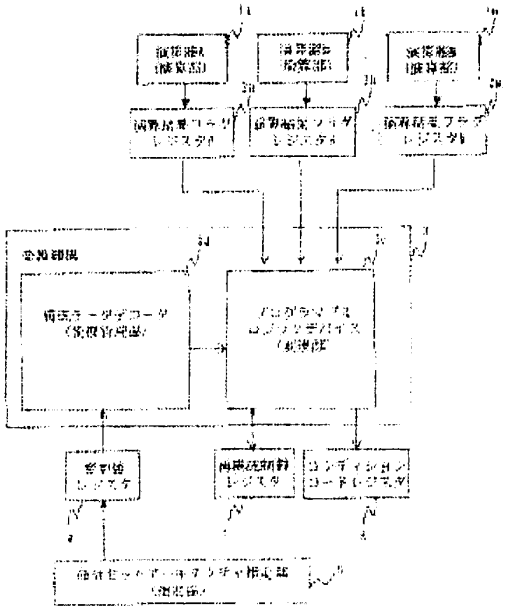
Applicant(s): MITSUBISHI ELECTRIC CORP ± (MITSUBISHI ELECTRIC CORP)

Classification: - international: ***G06F9/308;*** (IPC1-7): G06F9/308
- cooperative:

Application number: JP20020319234 20021101

Priority number(s): JP20020319234 20021101

Abstract of JP2004152201 (A)



LzLabs GmbH. Ex. 1004-35

PROBLEM TO BE SOLVED: To generate a highly versatile condition code by making a conversion mechanism converting a calculation result flag into the condition code variable. ; SOLUTION: An instruction set architecture specification part 9 specifies an instruction set architecture that is a conversion target in the conversion mechanism 3, and writes constitution data corresponding to the specified instruction set architecture into a changed value register 4. A constitution data decoder 3d reads a register value from the changed value register 4, decodes the constitution data, and outputs the decoded constitution data to a programmable logic device 3c. The programmable logic device 3c is reconstructed according to the constitution data, converts the calculation result flag into the condition code according to the instruction set architecture specified by the instruction set architecture specification part 9, and outputs the condition code to a condition code register 8. ; COPYRIGHT: (C)2004,JPO

Last updated: 13.03.2013 Worldwide Database 5.8.11.1; 92p

LzLabs GmbH. Ex. 1004-36

(19)日本国特許庁(JP)　(12)公開特許公報(A)　(11)特許出願公開番号
特開2004-152201
(P2004-152201A)
(43)公開日　平成16年5月27日(2004.5.27)

(51)Int.Cl.[7]	F I		テーマコード（参考）
G06F 9/308	G06F	9/30 340B	5B033

審査請求 未請求　請求項の数 7　OL　（全 11 頁）

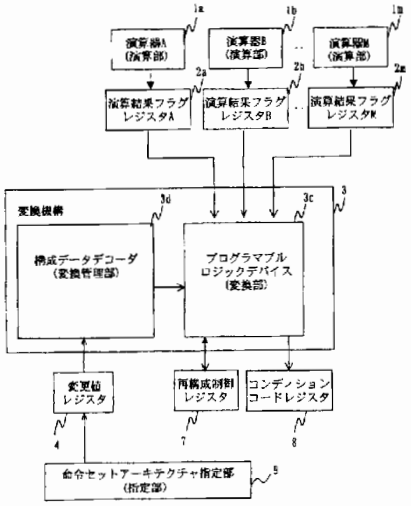
(21)出願番号　特願2002-319234(P2002-319234)
(22)出願日　平成14年11月1日(2002.11.1)

(71)出願人　000006013
三菱電機株式会社
東京都千代田区丸の内二丁目２番３号
(74)代理人　100099461
弁理士　溝井　章司
(74)代理人　100111497
弁理士　波田　啓子
(74)代理人　100111800
弁理士　竹内　三明
(74)代理人　100114878
弁理士　山地　博人
(72)発明者　伊藤　岳広
東京都千代田区丸の内二丁目２番３号　三菱電機株式会社内

Fターム(参考)　5B033 BD01 DD05

(54)【発明の名称】プロセッサ装置

(57)【要約】
【課題】演算結果フラグをコンディションコードに変換する変換機構を変更可能にすることで、汎用性の高いコンディションコードを生成出来るようにする。
【解決手段】命令セットアーキテクチャ指定部９が変換機構３での変換対象となる命令セットアーキテクチャを指定し、指定した命令セットアーキテクチャに対応する構成データを変更値レジスタ４に書き込み、構成データデコーダ３ｄは変更値レジスタ４からレジスタ値を読み込み、構成データをデコードしプログラマブルロジックデバイス３ｃに出力し、プログラマブルロジックデバイス３ｃは構成データに従って再構成され、演算結果フラグを命令セットアーキテクチャ指定部９により指定された命令セットアーキテクチャに従ったコンディションコードに変換し、コンディションコードレジスタ８に出力する。
【選択図】　図１



LzLabs GmbH. Ex. 1004-37

【特許請求の範囲】
【請求項１】
所定の演算を行い、演算結果を示す演算結果フラグを出力する演算部と、
複数種の命令セットアーキテクチャに対応し、演算結果フラグを複数種の命令セットアーキテクチャのうちの特定の命令セットアーキテクチャのコンディションコードに変換し、変換したコンディションコードを出力する変換部と、
前記変換部での変換の対象となる命令セットアーキテクチャを変換対象命令セットアーキテクチャとして指定する指定部とを有し、
前記変換部は、
演算結果フラグを、前記指定部により指定された変換対象命令セットアーキテクチャのコンディションコードに変換することを特徴とするプロセッサ装置。
【請求項２】
前記プロセッサ装置は、更に、
前記指定部により指定された変換対象命令セットアーキテクチャに対応する制御情報を前記変換部に対して出力する変換管理部を有し、
前記変換部は、
前記変換管理部からの制御情報に従い、演算結果フラグを変換対象命令セットアーキテクチャのコンディションコードに変換することを特徴とする請求項１に記載のプロセッサ装置。
【請求項３】
前記指定部は、
変換対象命令セットアーキテクチャに対応する制御情報を前記変換管理部に対して出力し、
前記変換管理部は、
前記指定部からの制御情報をデコードし、デコード後の制御情報を前記変換部に対して出力することを特徴とする請求項２に記載のプロセッサ装置。
【請求項４】
前記変換管理部は、
命令セットアーキテクチャごとにそれぞれの命令セットアーキテクチャに対応する制御情報を記憶する制御情報記憶部と、
前記制御情報記憶部に記憶された制御情報の中から変換対象命令セットアーキテクチャに対応する制御情報を選択する制御情報選択部とを有し、
前記指定部は、
変換対象命令セットアーキテクチャを示す変換対象命令セットアーキテクチャ情報を前記変換管理部に対して出力し、
前記変換管理部において、
前記制御情報選択部は、
前記指定部からの変換対象命令セットアーキテクチャ情報に基づき、前記制御情報記憶部に記憶された制御情報の中から変換対象命令セットアーキテクチャに対応する制御情報を選択し、選択した制御情報を前記変換部に対して出力することを特徴とする請求項２に記載のプロセッサ装置。
【請求項５】
前記変換部は、
命令セットアーキテクチャごとに内部回路を構成することにより、演算結果フラグを特定の命令セットアーキテクチャのコンディションコードに変換するプログラマブルロジックデバイスであり、
前記指定部により指定された変換対象命令セットアーキテクチャに対応させて内部回路を構成し、演算結果フラグを変換対象命令セットアーキテクチャのコンディションコードに変換することを特徴とする請求項１に記載のプロセッサ装置。
【請求項６】

LzLabs GmbH. Ex. 1004-38

前記プロセッサ装置は、更に、
前記指定部により指定された変換対象命令セットアーキテクチャに対応する制御情報を前記変換部に対して出力する変換管理部を有し、
前記変換部は、
前記変換管理部からの制御情報に従い、変換対象命令セットアーキテクチャに対応させて内部回路を構成し、演算結果フラグを変換対象命令セットアーキテクチャのコンディションコードに変換することを特徴とする請求項５に記載のプロセッサ装置。
【請求項７】
前記変換部は、
それぞれの命令セットアーキテクチャについて演算結果フラグの種類ごとに対応するコンディションコードを特定する変換テーブルを有し、
前記変換テーブルを検索して、前記演算部からの演算結果フラグ及び前記指定部により指定された変換対象命令セットアーキテクチャに対応するコンディションコードを選択し、選択したコンディションコードを出力することを特徴とする請求項１に記載のプロセッサ装置。
【発明の詳細な説明】
【０００１】
【発明の属する技術分野】
この発明は、マイクロプロセッサ等のプロセッサ装置上で様々なアーキテクチャの命令を命令エミュレーションにより実行する情報処理装置における汎用的なコンディションコードの生成技術に関するものである。
【０００２】
【従来の技術】
従来、特表平９－５０５４３０にあるように、複数のコンディションコードを保持し、そこからソフトウェアで設定した値によって必要なフラグを選択し、配列を変更することでプロセッサ間のコンディションコードの互換性を保障できる。
【０００３】
【特許文献１】
特表平９－５０５４３０号公報
【０００４】
【発明が解決しようとする課題】
上述したように従来の方式では、複数のコンディションコードから必要なフラグを選択することでプロセッサ間のコンディションコードの互換性を保障することができる。しかしコンディションコードの値を変えることはできないため、用意しているコンディションコード以外のフラグを使用するプロセッサの互換性を保障することができず、コンディションコードの汎用性が低いという問題があった。
【０００５】
本発明は上記問題点を解決するためになされたもので、演算結果フラグからコンディションコードを変換する変換手段を変更可能にすることで、汎用性の高いコンディションコードを生成出来るようにすることを目的とする。
【０００６】
【課題を解決するための手段】
本発明に係るプロセッサ装置は、
所定の演算を行い、演算結果を示す演算結果フラグを出力する演算部と、
複数種の命令セットアーキテクチャに対応し、演算結果フラグを複数種の命令セットアーキテクチャのうちの特定の命令セットアーキテクチャのコンディションコードに変換し、変換したコンディションコードを出力する変換部と、
前記変換部での変換の対象となる命令セットアーキテクチャを変換対象命令セットアーキテクチャとして指定する指定部とを有し、
前記変換部は、

LzLabs GmbH. Ex. 1004-39

演算結果フラグを、前記指定部により指定された変換対象命令セットアーキテクチャのコンディションコードに変換することを特徴とする。

【０００７】

【発明の実施の形態】

実施の形態１．

本発明の実施の形態について図面を参照して説明する。

【０００８】

図１は本発明の一実施の形態を示すブロック図である。

図において、１ａ～１ｍは演算器群、２ａ～２ｍは演算器群１ａ～１ｍが生成する演算結果フラグを保持するための演算結果フラグレジスタ群、４はソフトウェアから入力された プログラマブルロジックデバイス３ｃを変更する値を保持するための変更値レジスタ、３は演算結果フラグからコンディションフラグに変換する変換機構、３ｃはプログラマブルロジックデバイス、３ｄは変更値レジスタ値をデコードしてプログラマブルロジックデバイス３ｃを再構成する構成データを生成する構成データデコーダ、７は再構成処理の開始、終了を制御する再構成制御レジスタ、８は変換機構３で演算結果フラグから変換されたコンディションコードを保持するコンディションコードレジスタである。９は、変換機構３での変換の対象となる命令セットアーキテクチャを指定する命令セットアーキテクチャ指定部である。命令セットアーキテクチャ指定部９は、例えば、オペレーティングシステム（ＯＳ）等のソフトウェアである。

【０００９】

ここで、演算器１ａ～１ｍは、演算部の例であり、プログラマブルロジックデバイス３ｃは変換部の例であり、構成データデコーダ３ｄは変換管理部の例であり、命令セットアーキテクチャ指定部９は指定部の例である。また、構成データは、制御情報の例である。

【００１０】

次に動作について説明する。

マイクロプロセッサにて命令が実行されると各演算器１ａ～１ｍは演算を実行し、その演算結果の状態を表す演算結果フラグを各演算結果フラグレジスタ２ａ～２ｍに書き込む。変換機構３におけるプログラマブルロジックデバイス３ｃは各演算結果フラグレジスタ２ａ～２ｍから演算結果フラグ値を読み込み、コンディションコードに変換して、コンディションコードレジスタ８に書き込む。命令エミュレーション上で実行されるプログラムはコンディションコードレジスタ８の値によって動作を切替える。

【００１１】

次に変換機構３の変更手順について説明する。

まず、命令セットアーキテクチャ指定部（例えば、ソフトウェア）から変更値レジスタ４に対してプログラマブルロジックデバイスで構成された変換機構３を変更するための構成データ（デコード前の構成データ）を入力する。

この構成データ（デコード前の構成データ）は、プログラマブルロジックデバイス３ｃの内部回路を再構成するための制御情報であり、命令セットアーキテクチャごとに構成データが異なる。

このため、命令セットアーキテクチャ指定部９は、構成データ（デコード前の構成データ）を入力することによって、次にプログラマブルロジックデバイス３ｃが変換を行うべき命令セットアーキテクチャを指定する。

次にソフトウェアから再構成制御レジスタ７に再構成処理の開始を指示する値をセットすると構成データデコーダ３ｄは変更値レジスタ４からレジスタ値（デコード前の構成データ）を読み込む。

構成データデコーダ３ｄは読み込んだ変更値からプログラマブルロジックデバイス３ｃを再構成する構成データをデコードしプログラマブルロジックデバイス３ｃに出力しプログラマブルロジックデバイス３ｃは構成データに従って再構成される。そして、プログラマブルロジックデバイス３ｃは、演算結果フラグを命令セットアーキテクチャ指定部９により指定された命令セットアーキテクチャに従ったコンディションコードに変換し、コンデ

LzLabs GmbH. Ex. 1004-40

ィションコードレジスタ８に出力する。

【００１２】

次に、例えば図１において、命令によりプログラマブルロジックデバイスの再構成を実行する場合における再構成手順について、図２、図３を用いて説明する。

図２は再構成制御レジスタの構成図、図３は再構成のフローチャートである。

【００１３】

プログラマブルロジックデバイスの再構成を行う再構成実行命令が実行されると、図３のステップ３０１において、命令セットアーキテクチャ指定部９（例えばソフトウェア）から変更値レジスタ４に対しプログラマブルロジックデバイス３ｃを変更するための構成データ（デコード前の構成データ）が書き込まれる。書込み終了後ステップ３０２において、ソフトウェアから再構成制御レジスタ７の制御ビットがセットされる。

ステップ３０３において構成データデコーダ３ｄは制御ビットがセットされたことを検知すると変更値レジスタ４から値（デコード前の構成データ）を読み取り、ステップ３０４において再構成に使用する構成データをデコードし、再構成を開始する。

ステップ３０６においてプログラマブルロジックデバイス３ｃは再構成が開始されると再構成制御レジスタのステータスビットをセットする。

ステップ３０１からステップ３０６を、再構成が終了したことを表す再構成終了信号がプログラマブルロジックデバイス３ｃから出力されるまで繰り返す。

再構成終了信号を受けた構成データデコーダ３ｄは再構成制御レジスタ７のステータスビットをクリアする。ステータスビットがクリアされることによって再構成実行命令は終了する。

【００１４】

以上のように、演算結果フラグからコンディションコードに変換する変換機構をプログラマブルロジックデバイスで構成し、デコーダを介してプログラマブルロジックデバイスを再構成可能にすることで、汎用性の高いコンディションコードを生成することが可能になり、命令エミュレーション上で様々なアーキテクチャのプログラムを実行することが可能になる。

【００１５】

実施の形態２．

以上の実施の形態１では、プログラマブルロジックデバイスで構成した変換機構を、ソフトウェアにより入力された構成データからデコード回路でデコードして生成した構成データを用いて再構成するものであるが、次に複数の構成データをメモリ上に保持し例えばソフトウェアから再構成に使用する構成データを選択して変換機構を再構成する実施の形態を示す。

【００１６】

図４はこのような場合の一実施の形態を示すブロック図である。

図において、１ａ～１ｍは演算器群、２ａ～２ｍは演算器群１ａ～１ｍが生成する演算結果フラグを保持するための演算結果フラグレジスタ群、４はソフトウェアから入力された変換機構を変更する値を保持するための変更値レジスタ、３は演算結果フラグからコンディションフラグに変換する変換機構、３ｃはプログラマブルロジックデバイス、３ａはプログラマブルロジックデバイス３ｃを再構成するための構成データを保持する構成データメモリ、３ｂは変更値レジスタ４の値からプログラマブルロジックデバイスを再構成する構成データを構成データメモリ３ａから選択する構成データセレクタ、７は再構成処理の開始、終了を制御する再構成制御レジスタ、８は変換機構３で演算結果フラグから変換されたコンディションコードを保持するコンディションコードレジスタである。９は、変換機構３での変換の対象となる命令セットアーキテクチャを指定する命令セットアーキテクチャ指定部である。命令セットアーキテクチャ指定部９は、例えば、オペレーティングシステム（ＯＳ）等のソフトウェアである。

構成データメモリ３ａは、命令セットアーキテクチャごとに構成データを保有しており、構成データセレクタ３ｂは、命令セットアーキテクチャ指定部９により指定された命令セ

LzLabs GmbH. Ex. 1004-41

ットアーキテクチャに対応する構成データを構成データメモリ３ａが保有する構成データの中から選択する。

【００１７】

ここで、演算器１ａ～１ｍは、演算部の例であり、プログラマブルロジックデバイス３ｃは変換部の例であり、構成データメモリ３ａは制御情報記憶部の例であり、構成データセレクタ３ｂは制御情報選択部の例である。また、構成データメモリ３ａと構成データセレクタ３ｂを合わせたものが変換管理部の例である。命令セットアーキテクチャ指定部９は指定部の例である。また、構成データは、制御情報の例である。

【００１８】

次に動作について説明する。

マイクロプロセッサにて命令が実行されると各演算器１ａ～１ｍは演算を実行し、その演算結果の状態を表す演算結果フラグを各演算結果フラグレジスタ２ａ～２ｍに書き込む。変換機構３におけるプログラマブルロジックデバイス３ｃは各演算結果フラグレジスタ２ａ～２ｍから演算結果フラグ値を読み込み、コンディションコードに変換して、コンディションコードレジスタ８に書き込む。命令エミュレーション上で実行されるプログラムはコンディションコードレジスタ８の値によって動作を切替える。

【００１９】

次にソフトウェアによる変換機構３の変更手順について説明する。

まず、命令セットアーキテクチャ指定部９（例えば、ソフトウェア）から変更値レジスタ４に対してプログラマブルロジックデバイスで構成された変換機構３を変更するための変更値を入力する。

次にソフトウェアから再構成制御レジスタ７に再構成処理の開始を指示する値をセットすると構成データセレクタ３ｂは変更値レジスタ４からレジスタ値を読み込み、この値をもとに再構成に使用する構成データを構成データメモリ３ａから選択する。選択の結果、構成データメモリ３ａからプログラマブルロジックデバイス３ｃを再構成するための構成データが出力され、プログラマブルロジックデバイス３ｃは構成データに従って再構成される。そして、プログラマブルロジックデバイス３ｃは、演算結果フラグを命令セットアーキテクチャ指定部９により指定された命令セットアーキテクチャに従ったコンディションコードに変換し、コンディションコードレジスタ８に出力する。

【００２０】

次に、例えば図４において、構成データメモリがそれぞれ別々の構成データを保持した複数のメモリで構成されており、命令によりプログラマブルロジックデバイスの再構成を実行する場合におけるプログラマブルロジックデバイスの再構成手順の詳細について、図２、図５を用いて説明する。図２は再構成制御レジスタの構成図、図５は再構成のフローチャートである。

【００２１】

プログラマブルロジックデバイスの再構成を行う再構成実行命令が実行されると、図５のステップ２０１において、命令セットアーキテクチャ指定部９（例えばソフトウェア）から変更値レジスタ４に対しプログラマブルロジックデバイス３ｃを再構成する構成データが格納されたメモリを選択するための値が書き込まれる。

書込み終了後ステップ２０２において、ソフトウェアから再構成制御レジスタの制御ビットがセットされる。

ステップ２０３において構成データセレクタ３ｂは制御ビットがセットされたことを検知すると、変更値レジスタ４から値を読み取り、ステップ２０４において構成データメモリ３ａから変更値レジスタ値に対応する構成データが格納されたメモリを選択する。

ステップ２０５において選択されたメモリに格納された構成データによりプログラマブルロジックデバイス３ｃの再構成が開始される。

プログラマブルロジックデバイス３ｃは再構成が開始されると再構成制御レジスタのステータスビットをセットする。プログラマブルロジックデバイス３ｃは再構成が終了すると再構成制御レジスタ７のステータスビットをクリアする。ステータスビットがクリアされ

LzLabs GmbH. Ex. 1004-42

たことを検知すると再構成実行命令は終了する。

【００２２】

以上のように、演算結果フラグからコンディションコードに変換する変換機構をプログラマブルロジックデバイスで構成し、プログラマブルロジックデバイスの構成データを例えばソフトウェアから選択して再構成可能にすることで、汎用性の高いコンディションコードを生成することが可能になり、命令エミュレーション上で様々なアーキテクチャのプログラムを実行することが可能になる。

【００２３】

実施の形態３．

以上の実施の形態１、実施の形態２では、演算結果フラグからコンディションコードの変換機構をプログラマブルロジックデバイスで構成しソフトウェアから再構成していたが、次に変換機構をメモリ上のテーブルで構成する実施の形態を示す。

【００２４】

図６はこのような場合の一実施の形態を示すブロック図である。

図において、１ａ～１ｍは演算器群、２ａ～２ｍは演算器群１ａ～１ｍが生成する演算結果フラグを保持するための演算結果フラグレジスタ群、１０は実行する命令セットのアーキテクチャを表すアーキテクチャコードを保持するためのアーキテクチャコードレジスタ、１１は演算結果フラグレジスタ群２ａ～２ｍから読み込んだ各演算器１ａ～１ｍの演算結果フラグを保持するためのフラグレジスタ、１２はアーキテクチャコードレジスタ１０の値とフラグレジスタ１１の値から変換テーブル上の対応するコンディションコードのアドレスをデコードする変換テーブルアドレスデコーダ、１３は演算結果フラグとコンディションコードの対応を表すメモリ上の変換テーブル、８は変換テーブルから読み出されたコンディションコードを保持するためのコンディションコードレジスタ。９は、変換機構での変換の対象となる命令セットアーキテクチャを指定する命令セットアーキテクチャ指定部である。命令セットアーキテクチャ指定部９は、例えば、オペレーティングシステム（ＯＳ）等のソフトウェアである。

【００２５】

本実施の形態では、演算器１ａ～１ｍは、演算部の例であり、変換テーブルアドレスデコーダ１２と変換テーブル１３とを合わせたものが変換部の例であり、命令セットアーキテクチャ指定部９は指定部の例である。また、構成データは、制御情報の例である。

【００２６】

次に動作について説明する。

マイクロプロセッサにて命令が実行されると各演算器１ａ～１ｍは演算を実行し、その演算結果の状態を表す演算結果フラグを各演算結果フラグレジスタ２ａ～２ｍに書き込み、各演算結果フラグレジスタ２ａ～２ｍからフラグレジスタ１１に演算結果フラグが書き込まれる。また、命令セットアーキテクチャ指定部９は、次に変換の対象となる命令セットアーキテクチャを指定し、指定した命令セットアーキテクチャ（変換対象命令セットアーキテクチャ）を示す値をアーキテクチャコードレジスタ１０に書き込む。変換テーブルアドレスデコーダ１２はフラグレジスタ１１とアーキテクチャコードレジスタ１０のレジスタ値から変換テーブル１３上の対応するコンディションコードのアドレスを生成する。アドレス指定されたコンディションコードはコンディションコードレジスタ８に書き込まれ、命令エミュレーション上で実行されるプログラムはコンディションコードレジスタ８の値によって動作を切替える。

【００２７】

次に、例えば図６において、命令セットアーキテクチャ指定部９によってアーキテクチャコードをアーキテクチャコードレジスタ１０にセットする場合の、演算結果フラグからコンディションコードへの変換手順について、図７を用いて説明する。図７は変換手順のフローチャートである。

【００２８】

図７のステップ４０１において、命令セットアーキテクチャ指定部９はロードするプログ

LzLabs GmbH. Ex. 1004-43

ラムの命令セットアーキテクチャを認識し、対応するアーキテクチャコードをアーキテクチャコードレジスタ１０にセットする。ステップ４０２においてプログラムがロードされる。ステップ４０４において命令が実行されるたびに各演算結果フラグレジスタ２α～２ｍのレジスタ値がフラグレジスタ１１に書き込まれ、ステップ４０５において変換テーブルアドレスデコーダによって、変換テーブル上の対応するアドレスが生成される。ステップ４０６において変換テーブルの指定されたアドレスのコンディションコードが参照されコンディションコードレジスタ８に書き込まれる。以降ステップ４０３からステップ４０８を、プログラムが終了するまで繰り返す。

【００２９】

以上のように、演算結果フラグからコンディションコードに変換する変換機構を変換テーブルで構成することで、汎用性の高いコンディションコードを生成することが可能になり、命令エミュレーション上で様々なアーキテクチャのプログラムを実行することが可能になる。

【００３０】

ここで、実施の形態１～３で説明したマイクロプロセッサの特徴を再言する。

【００３１】

実施の形態１～３に示したマイクロプロセッサは、演算器で実行した演算結果により生成され演算器ごとの演算結果の状態を表す演算結果フラグをコンディションコードに変換する手段を備えたマイクロプロセッサであって、外部からの制御信号によって変更可能な、演算結果フラグからコンディションコードへの変換手段を備えることを特徴とする。

【００３２】

実施の形態１、２に示したマイクロプロセッサは、ソフトウェアにより変換手段を変更する値を入力する手段と、入力値を保持する手段と、入力値を読出し変換手段の制御信号を生成する手段を備えることを特徴とする。

【００３３】

実施の形態１に示したプロセッサは、デコード回路を備え、ソフトウェアからの入力値をデコードして変換手段への制御信号を生成することを特徴とする。

【００３４】

実施の形態２に示したプロセッサは、変換手段への制御信号パターンで構成したデータベースを備え、ソフトウェアによりデータベース上の制御信号パターンを選択することを特徴とする。

【００３５】

実施の形態１、２に示したプロセッサは、プログラマブルロジックデバイスと、プログラマブルロジックデバイスの構成データを保持する手段と、プログラマブルロジックデバイスを再構成する手段を備え、プログラマブルロジックデバイスを再構成することで変換手段を変更することを特徴とする。

【００３６】

実施の形態３に示したプロセッサは、演算結果フラグとコンディションコードの対応を表すメモリ上に構成されたテーブルと、各演算器の演算結果フラグレジスタ値を読み込む手段と、読み込んだ演算結果フラグレジスタ値と実行する命令セットのアーキテクチャを表すアーキテクチャコードから、対応するテーブル上のコンディションコードを選択する手段を備え、変換テーブルを参照することで演算結果フラグからコンディションコードを変換することを特徴とする。

【００３７】

【発明の効果】

このように、本発明によれば、演算結果フラグからコンディションコードへの変換を複数種類の命令セットアーキテクチャに対応させることができるため、命令エミュレーション上で様々なアーキテクチャのプログラムを実行することが可能になる。

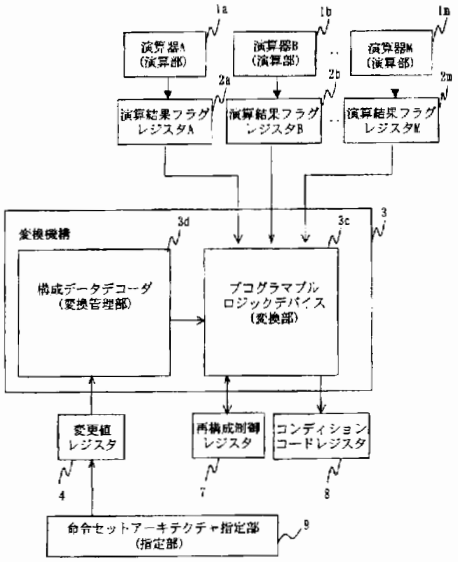
【図面の簡単な説明】

【図１】実施の形態１に係るマイクロプロセッサの構成例を示す図。

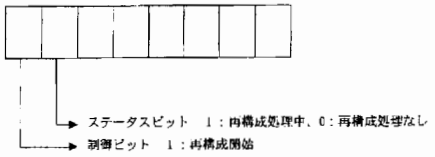
LzLabs GmbH. Ex. 1004-44

【図２】再構成制御レジスタの構成例を示す図。
【図３】実施の形態１に係る再構成手順の例を示すフローチャート図。
【図４】実施の形態２に係るマイクロプロセッサの構成例を示す図。
【図５】実施の形態２に係る再構成手順の例を示すフローチャート図。
【図６】実施の形態３に係るマイクロプロセッサの構成例を示す図。
【図７】実施の形態３に係る変換手順の例を示すフローチャート図。
【符号の説明】
１　演算器、２　演算結果フラグレジスタ、３　変換機構、４　変更値レジスタ、７　再構成制御レジスタ、８　コンディションコードレジスタ、９　命令セットアーキテクチャ指定部、１０　アーキテクチャコードレジスタ、１１　フラグレジスタ、１２　変換テーブルアドレスデコーダ、１３　変換テーブル。

【図１】

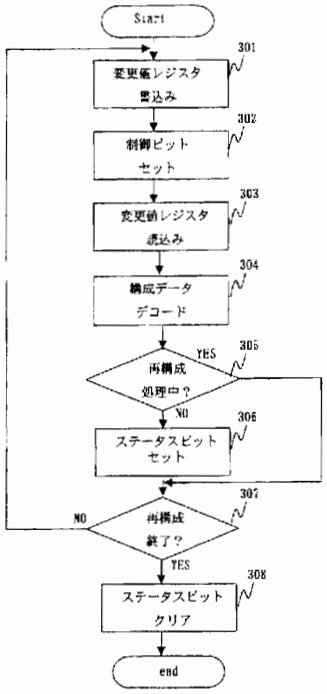


【図２】

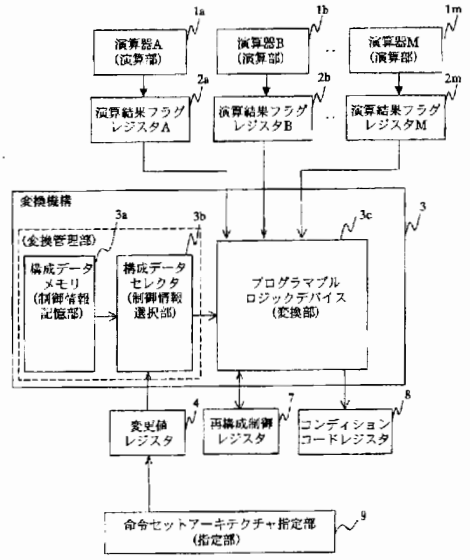


LzLabs GmbH. Ex. 1004-45

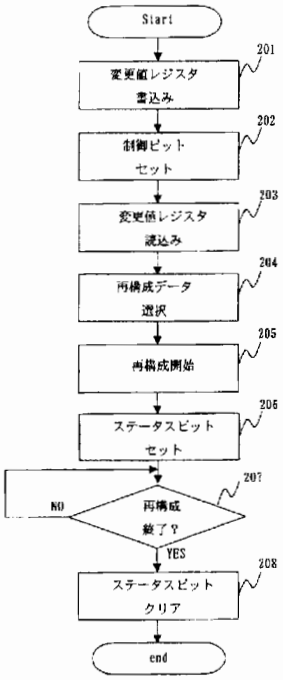
【図３】


Start
301
変更値レジスタ
書込み
302
制御ビット
セット
303
変更値レジスタ
読込み
304
構成データ
デコード
305
再構成
処理中？
YES
NO
306
ステータスビット
セット
307
再構成
終了？
NO
YES
308
ステータスビット
クリア
end

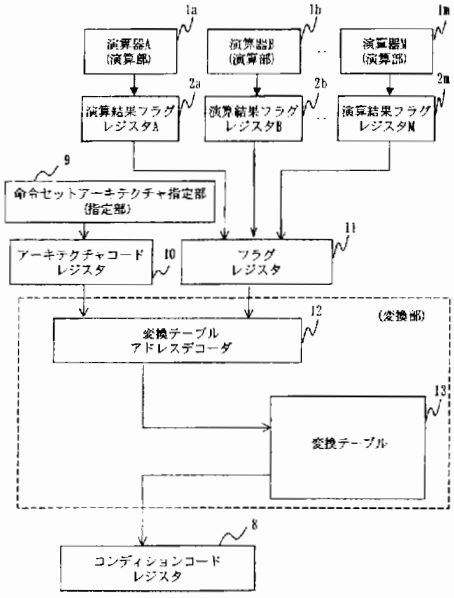
【図４】


1a
演算器A
(演算部)
1b
演算器B
(演算部)
1m
演算器M
(演算部)
2a
演算結果フラグ
レジスタA
2b
演算結果フラグ
レジスタB
2m
演算結果フラグ
レジスタM
変換機構
3a
3b
3c
3
(変換管理部)
構成データ
メモリ
(制御情報
記憶部)
構成データ
セレクタ
(制御情報
選択部)
プログラマブル
ロジックデバイス
(変換部)
4
変更値
レジスタ
7
再構成制御
レジスタ
8
コンディション
コードレジスタ
9
命令セットアーキテクチャ指定部
(指定部)

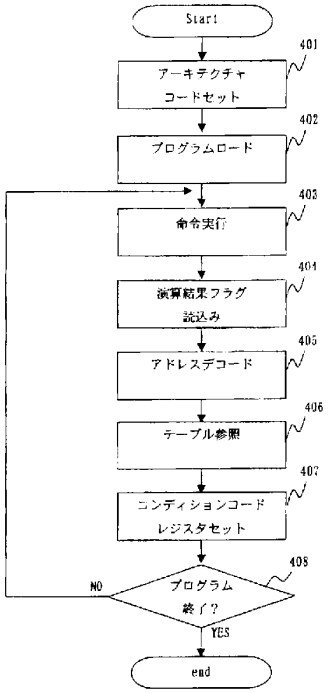
【図５】


Start
201
変更値レジスタ
書込み
202
制御ビット
セット
203
変更値レジスタ
読込み
204
再構成データ
選択
205
再構成開始
206
ステータスビット
セット
207
再構成
終了？
NO
YES
208
ステータスビット
クリア
end

【図６】


1a
演算器A
(演算部)
1b
演算器B
(演算部)
1m
演算器M
(演算部)
2a
演算結果フラグ
レジスタA
2b
演算結果フラグ
レジスタB
2m
演算結果フラグ
レジスタM
9
命令セットアーキテクチャ指定部
(指定部)
11
アーキテクチャコード
レジスタ
10
フラグ
レジスタ
(変換部)
12
変換テーブル
アドレスデコーダ
13
変換テーブル
8
コンディションコード
レジスタ

【図７】


Start
401
アーキテクチャ
コードセット
402
プログラムロード
403
命令実行
404
演算結果フラグ
読込み
405
アドレスデコード
406
テーブル参照
407
コンディションコード
レジスタセット
408
プログラム
終了？
NO
YES
end

Electronic Patent Application Fee Transmittal

Application Number:	11668623
Filing Date:	30-Jan-2007
Title of Invention:	EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS WITHOUT EXECUTING BRANCH INSTRUCTIONS
First Named Inventor/Applicant Name:	Reid T. Copeland
Filer:	Blanche E. Schiller/Jill Becker
Attorney Docket Number:	POU920060152US1

Filed as Large Entity

Utility under 35 USC 111(a) Filing Fees

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Basic Filing:				
Pages:				
Claims:				
Miscellaneous-Filing:				
Petition:				
Patent-Appeals-and-Interference:				
Post-Allowance-and-Post-Issuance:				
Extension-of-Time:				

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Miscellaneous:				
RCE - 2nd and Subsequent Request	1820	1	1700	1700
		Total in USD ($)		**1700**

Electronic Acknowledgement Receipt

EFS ID:	17352645
Application Number:	11668623
International Application Number:	
Confirmation Number:	4180
Title of Invention:	EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS WITHOUT EXECUTING BRANCH INSTRUCTIONS
First Named Inventor/Applicant Name:	Reid T. Copeland
Customer Number:	46369
Filer:	Blanche E. Schiller/Jill Becker
Filer Authorized By:	Blanche E. Schiller
Attorney Docket Number:	POU920060152US1
Receipt Date:	08-NOV-2013
Filing Date:	30-JAN-2007
Time Stamp:	14:13:27
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	yes
Payment Type	Deposit Account
Payment was successfully received in RAM	$1700
RAM confirmation Number	524
Deposit Account	090463
Authorized User	
The Director of the USPTO is hereby authorized to charge indicated fees and credit any overpayment as follows: Charge any Additional Fees required under 37 C.F.R. Section 1.17 (Patent application and reexamination processing fees)	

File Listing:					
Document Number	**Document Description**	**File Name**	**File Size(Bytes)/ Message Digest**	**Multi Part /.zip**	**Pages (if appl.)**
1	Request for Continued Examination (RCE)	POU920060152US1_RCE3.pdf	697827 7eeae0d8be5123365b8169595982835975 4393aa	no	3
Warnings:					
Information:					
2	Transmittal Letter	POU920060152US1_IDS02_SOR.pdf	44660 84fbc8ee362582c5e7499d65d3cf2d5b715 400c0	no	1
Warnings:					
Information:					
3	Information Disclosure Statement (IDS) Form (SB08)	POU920060152US1_IDS02.pdf	84700 43c05f1f63fdfd766430f62d348e362968667 569	no	1
Warnings:					
Information:					
This is not an USPTO supplied IDS fillable form					
4	Foreign Reference	JPS58195963.pdf	201051 c4f227dc1528e27c4bccbdf6df73bbcbd828 dde2	no	5
Warnings:					
Information:					
5	Foreign Reference	JP2004152201.pdf	650414 6ab5f3b669931a4196b284434d8b6352145 a4010	no	13
Warnings:					
Information:					
6	Fee Worksheet (SB06)	fee-info.pdf	30646 1f84c7e56074c37bcf41655c0bbdc11824f8 6bde	no	2
Warnings:					
Information:					
		Total Files Size (in bytes):	1709298		

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office
If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

NOTICE OF ALLOWANCE AND FEE(S) DUE

46369 7590 08/13/2013

HESLIN ROTHENBERG FARLEY & MESITI P.C.
5 COLUMBIA CIRCLE
ALBANY, NY 12203

EXAMINER
HUISMAN, DAVID J

ART UNIT	PAPER NUMBER
2183	

DATE MAILED: 08/13/2013

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
11/668,623	01/30/2007	Reid T. Copeland	POU920060152US1	4180

TITLE OF INVENTION: EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS WITHOUT EXECUTING BRANCH INSTRUCTIONS

APPLN. TYPE	ENTITY STATUS	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	UNDISCOUNTED	$1780	$300	$0	$2080	11/13/2013

THE APPLICATION IDENTIFIED ABOVE HAS BEEN EXAMINED AND IS ALLOWED FOR ISSUANCE AS A PATENT. PROSECUTION ON THE MERITS IS CLOSED. THIS NOTICE OF ALLOWANCE IS NOT A GRANT OF PATENT RIGHTS. THIS APPLICATION IS SUBJECT TO WITHDRAWAL FROM ISSUE AT THE INITIATIVE OF THE OFFICE OR UPON PETITION BY THE APPLICANT. SEE 37 CFR 1.313 AND MPEP 1308.

THE ISSUE FEE AND PUBLICATION FEE (IF REQUIRED) MUST BE PAID WITHIN THREE MONTHS FROM THE MAILING DATE OF THIS NOTICE OR THIS APPLICATION SHALL BE REGARDED AS ABANDONED. THIS STATUTORY PERIOD CANNOT BE EXTENDED. SEE 35 U.S.C. 151. THE ISSUE FEE DUE INDICATED ABOVE DOES NOT REFLECT A CREDIT FOR ANY PREVIOUSLY PAID ISSUE FEE IN THIS APPLICATION. IF AN ISSUE FEE HAS PREVIOUSLY BEEN PAID IN THIS APPLICATION (AS SHOWN ABOVE), THE RETURN OF PART B OF THIS FORM WILL BE CONSIDERED A REQUEST TO REAPPLY THE PREVIOUSLY PAID ISSUE FEE TOWARD THE ISSUE FEE NOW DUE.

HOW TO REPLY TO THIS NOTICE:

I. Review the ENTITY STATUS shown above. If the ENTITY STATUS is shown as SMALL or MICRO, verify whether entitlement to that entity status still applies.

If the ENTITY STATUS is the same as shown above, pay the TOTAL FEE(S) DUE shown above.

If the ENTITY STATUS is changed from that shown above, on PART B - FEE(S) TRANSMITTAL, complete section number 5 titled "Change in Entity Status (from status indicated above)".

For purposes of this notice, small entity fees are 1/2 the amount of undiscounted fees, and micro entity fees are 1/2 the amount of small entity fees.

II. PART B - FEE(S) TRANSMITTAL, or its equivalent, must be completed and returned to the United States Patent and Trademark Office (USPTO) with your ISSUE FEE and PUBLICATION FEE (if required). If you are charging the fee(s) to your deposit account, section "4b" of Part B - Fee(s) Transmittal should be completed and an extra copy of the form should be submitted. If an equivalent of Part B is filed, a request to reapply a previously paid issue fee must be clearly made, and delays in processing may occur due to the difficulty in recognizing the paper as an equivalent of Part B.

III. All communications regarding this application must give the application number. Please direct all communications prior to issuance to Mail Stop ISSUE FEE unless advised to the contrary.

IMPORTANT REMINDER: Utility patents issuing on applications filed on or after Dec. 12, 1980 may require payment of maintenance fees. It is patentee's responsibility to ensure timely payment of maintenance fees when due.

LzLabs GmbH. Ex. 1004-53

PART B - FEE(S) TRANSMITTAL

Complete and send this form, together with applicable fee(s), to: Mail **Mail Stop ISSUE FEE**
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450
or Fax (571)-273-2885

INSTRUCTIONS: This form should be used for transmitting the ISSUE FEE and PUBLICATION FEE (if required). Blocks 1 through 5 should be completed where appropriate. All further correspondence including the Patent, advance orders and notification of maintenance fees will be mailed to the current correspondence address as indicated unless corrected below or directed otherwise in Block 1, by (a) specifying a new correspondence address; and/or (b) indicating a separate "FEE ADDRESS" for maintenance fee notifications.

CURRENT CORRESPONDENCE ADDRESS (Note: Use Block 1 for any change of address)

46369 7590 08/13/2013
HESLIN ROTHENBERG FARLEY & MESITI P.C.
5 COLUMBIA CIRCLE
ALBANY, NY 12203

Note: A certificate of mailing can only be used for domestic mailings of the Fee(s) Transmittal. This certificate cannot be used for any other accompanying papers. Each additional paper, such as an assignment or formal drawing, must have its own certificate of mailing or transmission.

Certificate of Mailing or Transmission

I hereby certify that this Fee(s) Transmittal is being deposited with the United States Postal Service with sufficient postage for first class mail in an envelope addressed to the Mail Stop ISSUE FEE address above, or being facsimile transmitted to the USPTO (571) 273-2885, on the date indicated below.

(Depositor's name)
(Signature)
(Date)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
11/668,623	01/30/2007	Reid T. Copeland	POU920060152US1	4180

TITLE OF INVENTION: EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS WITHOUT EXECUTING BRANCH INSTRUCTIONS

APPLN. TYPE	ENTITY STATUS	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	UNDISCOUNTED	$1780	$300	$0	$2080	11/13/2013

EXAMINER	ART UNIT	CLASS-SUBCLASS
HUISMAN, DAVID J	2183	712-220000

1. Change of correspondence address or indication of "Fee Address" (37 CFR 1.363).

❑ Change of correspondence address (or Change of Correspondence Address form PTO/SB/122) attached.

❑ "Fee Address" indication (or "Fee Address" Indication form PTO/SB/47; Rev 03-02 or more recent) attached. **Use of a Customer Number is required.**

2. For printing on the patent front page, list

(1) the names of up to 3 registered patent attorneys or agents OR, alternatively,

(2) the name of a single firm (having as a member a registered attorney or agent) and the names of up to 2 registered patent attorneys or agents. If no name is listed, no name will be printed.

1 _____
2 _____
3 _____

3. ASSIGNEE NAME AND RESIDENCE DATA TO BE PRINTED ON THE PATENT (print or type)

PLEASE NOTE: Unless an assignee is identified below, no assignee data will appear on the patent. If an assignee is identified below, the document has been filed for recordation as set forth in 37 CFR 3.11. Completion of this form is NOT a substitute for filing an assignment.

(A) NAME OF ASSIGNEE (B) RESIDENCE: (CITY and STATE OR COUNTRY)

Please check the appropriate assignee category or categories (will not be printed on the patent) : ❑ Individual ❑ Corporation or other private group entity ❑ Government

4a. The following fee(s) are submitted:

❑ Issue Fee

❑ Publication Fee (No small entity discount permitted)

❑ Advance Order - # of Copies _____

4b. Payment of Fee(s): **(Please first reapply any previously paid issue fee shown above)**

❑ A check is enclosed.

❑ Payment by credit card. Form PTO-2038 is attached.

❑ The Director is hereby authorized to charge the required fee(s), any deficiency, or credit any overpayment, to Deposit Account Number _____ (enclose an extra copy of this form).

LzLabs GmbH. Ex. 1004-54

5. **Change in Entity Status** (from status indicated above)

❑ Applicant certifying micro entity status. See 37 CFR 1.29	NOTE: Absent a valid certification of Micro Entity Status (see form PTO/SB/15A and 15B), issue fee payment in the micro entity amount will not be accepted at the risk of application abandonment.
❑ Applicant asserting small entity status. See 37 CFR 1.27	NOTE: If the application was previously under micro entity status, checking this box will be taken to be a notification of loss of entitlement to micro entity status.
❑ Applicant changing to regular undiscounted fee status.	NOTE: Checking this box will be taken to be a notification of loss of entitlement to small or micro entity status, as applicable.

NOTE: The Issue Fee and Publication Fee (if required) will not be accepted from anyone other than the applicant; a registered attorney or agent; or the assignee or other party in interest as shown by the records of the United States Patent and Trademark Office.

Authorized Signature _____ Date _____

Typed or printed name _____ Registration No. _____

This collection of information is required by 37 CFR 1.311. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, Virginia 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450.

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

LzLabs GmbH. Ex. 1004-55

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
11/668,623	01/30/2007	Reid T. Copeland	POU920060152US1	4180

46369 7590 08/13/2013

HESLIN ROTHENBERG FARLEY & MESITI P.C.
5 COLUMBIA CIRCLE
ALBANY, NY 12203

EXAMINER	
HUISMAN, DAVID J	

ART UNIT	PAPER NUMBER
2183	

DATE MAILED: 08/13/2013

Determination of Patent Term Adjustment under 35 U.S.C. 154 (b)
(application filed on or after May 29, 2000)

The Patent Term Adjustment to date is 1187 day(s). If the issue fee is paid on the date that is three months after the mailing date of this notice and the patent issues on the Tuesday before the date that is 28 weeks (six and a half months) after the mailing date of this notice, the Patent Term Adjustment will be 1187 day(s).

If a Continued Prosecution Application (CPA) was filed in the above-identified application, the filing date that determines Patent Term Adjustment is the filing date of the most recent CPA.

Applicant will be able to obtain more detailed information by accessing the Patent Application Information Retrieval (PAIR) WEB site (http://pair.uspto.gov).

Any questions regarding the Patent Term Extension or Adjustment determination should be directed to the Office of Patent Legal Administration at (571)-272-7702. Questions relating to issue and publication fee payments should be directed to the Customer Service Center of the Office of Patent Publication at 1-(888)-786-0101 or (571)-272-4200.

LzLabs GmbH. Ex. 1004-56

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether disclosure of these records is required by the Freedom of Information Act.
2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.
3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.
4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).
5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.
6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).
7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.
8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspection or an issued patent.
9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

Notice of Allowability	Application No.	Applicant(s)	
	11/668,623	COPELAND ET AL.	
	Examiner	Art Unit	
	DAVID HUISMAN	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☒ This communication is responsive to *the RCE filed on October 29, 2012*.

2. ☐ An election was made by the applicant in response to a restriction requirement set forth during the interview on _____; the restriction requirement and election have been incorporated into this action.

3. ☒ The allowed claim(s) is/are *9-16, 18-22, 24-27, and 29-31 (hereafter collectively renumbered as claims 1-20)*. As a result of the allowed claim(s), you may be eligible to benefit from the **Patent Prosecution Highway** program at a participating intellectual property office for the corresponding application. For more information, please see http://www.uspto.gov/patents/init_events/pph/index.jsp or send an inquiry to PPHfeedback@uspto.gov .

4. ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
   a) ☐ All b) ☐ Some* c) ☐ None of the:
      1. ☐ Certified copies of the priority documents have been received.
      2. ☐ Certified copies of the priority documents have been received in Application No. _____ .
      3. ☐ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

   * Certified copies not received: _____.

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.
THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.

5. ☐ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
   ☐ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date _____.

Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).

6. ☐ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Attachment(s)

1. ☐ Notice of References Cited (PTO-892)
2. ☐ Information Disclosure Statements (PTO/SB/08), Paper No./Mail Date _____
3. ☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material
4. ☐ Interview Summary (PTO-413), Paper No./Mail Date _____ .
5. ☒ Examiner's Amendment/Comment
6. ☐ Examiner's Statement of Reasons for Allowance
7. ☐ Other _____.

/David J. Huisman/
Primary Examiner, Art Unit 2183

LzLabs GmbH. Ex. 1004-58

Continued Examination Under 37 CFR 1.114

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on October 29, 2012, has been entered.

Interview Request

2. Applicant's filed an interview request on October 29, 2012. MPEP 713.01 states that "An interview should be had only when the nature of the case is such that the interview could serve to develop and clarify specific issues and lead to a mutual understanding between the examiner and the applicant, and thereby advance the prosecution of the application." Since the examiner did not need clarification on the amendment filed in conjunction with the interview request, any granted interview would only delay prosecution, and would not be appropriate at this time. Therefore, the interview request is denied.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to DAVID HUISMAN whose telephone number is (571) 272-4168. The examiner can normally be reached on Monday-Friday (9:00-6:00).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Andrew Caldwell can be reached on (571) 272-3702. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/David J. Huisman/
Primary Examiner, Art Unit 2183

EAST Search History

EAST Search History (Prior Art)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
S42	1	"20030093649".pn.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S43	86	emulat$3 with "condition code"	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S44	102	emulat$3 with ("condition code" "flag bit" "status bit") and (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S45	29	emulat$3 with ("condition code" "flag bit" "status bit") same branch and (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S46	68	("20020026633" | "4514803" | "4812975" | "5313614" | "5329471" | "5408637" | "5440702" | "5551015" | "5574873" | "5574927" | "5577233" | "5598560" | "5606696" | "5649203" | "5724590" | "5737631" | "5751982" | "5768593" | "5805792" | "5805895" | "5870575" | "5890000" | "5901308" | "5903760" | "5925124" | "5930509" | "5946491" | "5956495" | "5966539" | "5983335" | "6000028" | "6009261" | "6031994" | "6076155" | "6077314" | "6091897" | "6105124" | "6126329" | "6170083" | "6185675" | "6199095" | "6205545" | "6223339" | "6301705" | "6308318" | "6321379" | "6330691" | "6360194" | "6367067" | "6427234" | "6434741" | "6463582" | "6477641" | "6502237" | "6505296" | "6522985" | "6523173" | "6526572" | "6530078" | "6530079" | "6535903" | "6539471" | "6539541" | "6549987" | "6564297" | "6594824" | "6631514" | "6643630").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S47	6	(emulat$3 translat$3) same ("condition code" "flag bit" "status bit") same (without avoid$4) near2 branch$3 and (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S48	144	(emulat$3 translat$3) with (without avoid$4) near2 branch$3 and (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S49	12	(emulat$3 translat$3) with (without avoid$4) near2 branch$3 and (overflow carry sign) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S50	5	("4951195" | "5241678" | "5301325" | "5313614" | "5598560").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S51	28	("5903760").URPN.	USPAT	OR	ON	2009/03/16 16:40
S52	68	("20020026633" | "4514803" | "4812975" | "5313614" | "5329471" | "5408637" |	US-PGPUB; USPAT;	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-61

		"5440702" | "5551015" | "5574873" | "5574927" | "5577233" | "5598560" | "5606696" | "5649203" | "5724590" | "5737631" | "5751982" | "5768593" | "5805792" | "5805895" | "5870575" | "5890000" | "5901308" | "5903760" | "5925124" | "5930509" | "5946491" | "5956495" | "5966539" | "5983335" | "6000028" | "6009261" | "6031994" | "6076155" | "6077314" | "6091897" | "6105124" | "6126329" | "6170083" | "6185675" | "6199095" | "6205545" | "6223339" | "6301705" | "6308318" | "6321379" | "6330691" | "6360194" | "6367067" | "6427234" | "6434741" | "6463582" | "6477641" | "6502237" | "6505296" | "6522985" | "6523173" | "6526572" | "6530078" | "6530079" | "6535903" | "6539471" | "6539541" | "6549987" | "6564297" | "6594824" | "6631514" | "6643630").PN.	USOCR			
S53	68	("20020026633" | "4514803" | "4812975" | "5313614" | "5329471" | "5408637" | "5440702" | "5551015" | "5574873" | "5574927" | "5577233" | "5598560" | "5606696" | "5649203" | "5724590" | "5737631" | "5751982" | "5768593" | "5805792" | "5805895" | "5870575" | "5890000" | "5901308" | "5903760" | "5925124" | "5930509" | "5946491" | "5956495" | "5966539" | "5983335" | "6000028" | "6009261" | "6031994" | "6076155" | "6077314" | "6091897" | "6105124" | "6126329" | "6170083" | "6185675" | "6199095" | "6205545" | "6223339" | "6301705" | "6308318" | "6321379" | "6330691" | "6360194" | "6367067" | "6427234" | "6434741" | "6463582" | "6477641" | "6502237" | "6505296" | "6522985" | "6523173" | "6526572" | "6530078" | "6530079" | "6535903" | "6539471" | "6539541" | "6549987" | "6564297" | "6594824" | "6631514" | "6643630").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S54	723	(emulat$3 translat$3) with compar$4 with instruction	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S55	21	(emulat$3 translat$3) with compar$4 with instruction same (code flag) near2 (condition status bit)	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S56	412	712/209.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S57	487	712/221.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S58	867	712/227.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S59	897	717/136.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-62

S60	265	717/137.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S61	234	717/138.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S62	1	emulat$3 with ("condition code" "flag bit" "status bit") and (overflow carry sign)	USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2009/03/16 16:40
S63	1362	717/124.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S64	140	717/134.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S65	1560	(emulat$3 translat$3) same (arithmetic add$5 subtract$3) same (code flag) near2 (condition status bit)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S66	101	(emulat$3 translat$3) same (arithmetic add$5 subtract$3) with set$4 with (code flag) near2 (condition status bit)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S67	37	(emulat$3 translat$3) same (arithmetic add$5 subtract$3) with set$4 with (code flag) near2 (condition status bit) same (Z O S C sign zero overflow carry)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S68	29	(emulat$3 translat$3) same (arithmetic add$5 subtract$3) with set$4 with (code flag) near2 (condition status bit) same (Z O S C sign zero overflow carry) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S69	7	("20020046298" "20040176941" "20050015754" "20070011664" "5301302" "5678032" "5966515").pn.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S70	4	S69 and (code flag) near2 (condition status bit)	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S71	119	simulat$3 near4 (code flag) near2 (condition status bit)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S72	67	simulat$3 near4 (code flag) near2 (condition status bit) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S73	35	simulat$3 near4 (code flag) near2 (condition status bit) and (overflow carry) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S74	127	(emulat$3 translat$3) and (host target) with set$4 with (code flag) near2 (condition status bit) and ((overflow O) and (carry C) and (sign S negative N)) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S75	59	(emulat$3 translat$3) and (host target) with set$4 with (code flag) near2 (condition status bit) and ((overflow O) and (carry C) and (sign S negative N)) and ("712"/$.ccls. "717"/$.ccls.)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S76	32	emulat$3 and (host target) with set$4 with (code flag) near2 (condition status bit) and ((overflow O) and (carry C) and	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-63

		(sign S negative N)) and ("712"/$.ccls. "717"/$.ccls.)				
S77	1	emulat$3 and (host target) with set$4 with (code flag) near2 (condition status bit) same ((overflow O) and (carry C) and (sign S negative N)) and ("712"/$.ccls. "717"/$.ccls.)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S78	1	emulat$3 and (host target) with set$4 with (code flag) near2 (condition status bit) same (((overflow O) and (carry C) and (sign S negative N))) and ("712"/$.ccls. "717"/$.ccls.)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S79	6	("4415969" | "5327537" | "5440749" | "5493687" | "5590359" | "5689720").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S80	2	("5774694").URPN.	USPAT	OR	ON	2009/03/16 16:40
S81	8	("5313614" | "5481693" | "5619664" | "5774694" | "5859999" | "5860017" | "5884059" | "6049864").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S82	1	"20060173931".pn.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:44
S83	446	703/26.ccls.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:46
S84	20	emulat$3 with ("condition code" "flag bit" "status bit") same (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/07/14 17:52
S85	0	emulat$3 with ("condition code" "flag bit" "status bit") same (overflow carry sign)	USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2009/07/14 17:53
S86	47	huisman.xp.	US-PGPUB; USPAT	OR	ON	2010/03/04 09:27
S87	1913	subtract$3 with carry with (bit flag)	US-PGPUB; USPAT	OR	ON	2010/03/04 11:03
S88	186	subtract$3 with set$4 with carry adj (bit flag)	US-PGPUB; USPAT	OR	ON	2010/03/04 11:03
S89	112	subtract$3 with set$4 with carry adj (bit flag) and "712"/$.ccls.	US-PGPUB; USPAT	OR	ON	2010/03/04 11:03
S90	2	("5574931" "5999976").pn.	US-PGPUB; USPAT	OR	ON	2010/03/04 15:04
S98	1	distinguish$3 with (outcome categor$4) with condition near2 (code flag)	US-PGPUB; USPAT	OR	ON	2012/07/03 14:35
S99	86	categor$4 with condition near2 (code flag)	US-PGPUB; USPAT	OR	ON	2012/07/03 14:35
S100	8	categor$4 with condition near2 (code flag) and "712"/$.ccls.	US-PGPUB; USPAT	OR	ON	2012/07/03 14:35
S102	8	categor$4 with condition near2 (code flag)	USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2012/07/03 14:37
S103	1297	712/220.ccls.	US-PGPUB; USPAT	OR	ON	2012/11/13 10:54
S104	591	703/26.ccls.	US-PGPUB; USPAT	OR	ON	2012/11/13 10:54

LzLabs GmbH. Ex. 1004-64

S105	93	(intermediate provisional temporary) near3 ("condition code" "flag register")	US-PGPUB; USPAT	OR	ON	2012/11/13 11:16
S106	35	(intermediate provisional temporary) near3 ("condition code" "flag register") and ("712"/$.ccls. "703"/$.ccls.)	US-PGPUB; USPAT	OR	ON	2012/11/13 11:28
S107	68	predict$3 near3 ("condition code" "flag register") and ("712"/$.ccls. "703"/$.ccls.)	US-PGPUB; USPAT	OR	ON	2012/11/13 11:40
S108	6	(intermediate provisional temporary) near3 ("condition code" "flag register")	USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2012/11/13 12:39
S109	0	predict$3 near3 ("condition code" "flag register") and ("712"/$.ccls. "703"/$.ccls.)	USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2012/11/13 12:42
S111	1409	712/220.ccls.	US-PGPUB; USPAT	OR	ON	2013/07/14 18:25
S112	623	703/26.ccls.	US-PGPUB; USPAT	OR	ON	2013/07/14 18:25

EAST Search History (Interference)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
S110	16	((intermediate provisional temporary) near3 ("condition code" "flag register")).clm.	US-PGPUB; USPAT; UPAD	OR	ON	2012/11/13 12:33
S113	16	((intermediate provisional temporary) near3 ("condition code" "flag register")).clm.	US-PGPUB; USPAT; UPAD	OR	ON	2013/07/14 18:26

7/14/2013 6:36:00 PM
C:\Users\dhuisman\Documents\EAST\Workspaces\11-668623.wsp

LzLabs GmbH. Ex. 1004-65

Issue Classification	**Application/Control No.** 11668623	**Applicant(s)/Patent Under Reexamination** COPELAND ET AL.
	Examiner DAVID J HUISMAN	**Art Unit** 2183

CPC

Symbol					Type	Version
			/			
			/			
			/			
			/			
			/			
			/			
			/			
			/			
			/			
			/			
			/			
			/			
			/			
			/			
			/			

CPC Combination Sets

Symbol					Type	Set	Ranking	Version
			/					
			/					

NONE (Assistant Examiner)	(Date)	**Total Claims Allowed:** 20	
/DAVID J HUISMAN/ Primary Examiner.Art Unit 2183 (Primary Examiner)	07/14/2013 (Date)	O.G. Print Claim(s) 1	O.G. Print Figure 2

LzLabs GmbH. Ex. 1004-66

Issue Classification	Application/Control No.	Applicant(s)/Patent Under Reexamination
	11668623	COPELAND ET AL.
	Examiner	**Art Unit**
	DAVID J HUISMAN	2183

US ORIGINAL CLASSIFICATION		INTERNATIONAL CLASSIFICATION	
CLASS	SUBCLASS	CLAIMED	NON-CLAIMED
712	220	G 0 6 F 9 / 455 (2006.0)	

CROSS REFERENCE(S)					
CLASS	SUBCLASS (ONE SUBCLASS PER BLOCK)				
703	26				

NONE	Total Claims Allowed:
(Assistant Examiner) (Date)	20
/DAVID J HUISMAN/ Primary Examiner.Art Unit 2183 07/14/2013	O.G. Print Claim(s): 1 / O.G. Print Figure: 2
(Primary Examiner) (Date)	

LzLabs GmbH. Ex. 1004-67

Issue Classification	Application/Control No.	Applicant(s)/Patent Under Reexamination
	11668623	COPELAND ET AL.
	Examiner	**Art Unit**
	DAVID J HUISMAN	2183

☐ Claims renumbered in the same order as presented by applicant ☐ CPA ☐ T.D. ☐ R.1.47

Final	Original	Final	Original	Final	Original	Final	Original	Final	Original	Final	Original	Final	Original	Final	Original
1	9	19	25												
2	10	20	26												
3	11	6	27												
4	12		28												
5	13	7	29												
9	14	8	30												
11	15	10	31												
12	16														
	17														
13	18														
14	19														
15	20														
16	21														
17	22														
	23														
18	24														

NONE		**Total Claims Allowed:**	
(Assistant Examiner)	(Date)	20	
/DAVID J HUISMAN/ Primary Examiner.Art Unit 2183	07/14/2013	O.G. Print Claim(s)	O.G. Print Figure
(Primary Examiner)	(Date)	1	2

LzLabs GmbH. Ex. 1004-68

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

BIB DATA SHEET

CONFIRMATION NO. 4180

SERIAL NUMBER	FILING or 371(c) DATE RULE	CLASS	GROUP ART UNIT	ATTORNEY DOCKET NO.
11/668,623	01/30/2007	712	2183	POU920060152US1

APPLICANTS

Reid T. Copeland, Toronto, CANADA;
Patrick R. Doyle, Toronto, CANADA;
Charles B. Hall, Calgary, CANADA;
Andrew Johnson, Hampshire, UNITED KINGDOM;
Ali I. Sheikh, Toronto, CANADA;

**** CONTINUING DATA ***************************

**** FOREIGN APPLICATIONS ***************************

**** IF REQUIRED, FOREIGN FILING LICENSE GRANTED ****
02/22/2007

Foreign Priority claimed ☐ Yes ☑ No 35 USC 119(a-d) conditions met ☐ Yes ☑ No Verified and Acknowledged /DAVID J HUISMAN/ Examiner's Signature	☐ Met after Allowance Initials	STATE OR COUNTRY	SHEETS DRAWINGS	TOTAL CLAIMS	INDEPENDENT CLAIMS
		CANADA	4	~~8~~ 20	3

DJH
11-13-12

ADDRESS

HESLIN ROTHENBERG FARLEY & MESITI P.C.
5 COLUMBIA CIRCLE
ALBANY, NY 12203
UNITED STATES

TITLE

EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS WITHOUT EXECUTING BRANCH INSTRUCTIONS

FILING FEE RECEIVED		
1000	FEES: Authority has been given in Paper No._____ to charge/credit DEPOSIT ACCOUNT No._____ for following:	☐ All Fees ☐ 1.16 Fees (Filing) ☐ 1.17 Fees (Processing Ext. of time) ☐ 1.18 Fees (Issue) ☐ Other _____ ☐ Credit

LzLabs GmbH. Ex. 1004-69

Search Notes	Application/Control No.	Applicant(s)/Patent Under Reexamination
	11668623	COPELAND ET AL.
	Examiner	Art Unit
	DAVID J HUISMAN	2183

CPC- SEARCHED		
Symbol	Date	Examiner

CPC COMBINATION SETS - SEARCHED		
Symbol	Date	Examiner

US CLASSIFICATION SEARCHED			
Class	Subclass	Date	Examiner
712	209, 221, 227	3-16-09	DJH
717	124, 134, 136, 137, 138	3-16-09	DJH
703	26	3-16-09	DJH
712	220	7-14-13	DJH
703	26	7-14-13	DJH

SEARCH NOTES		
Search Notes	Date	Examiner
EAST Search (all databases)	3/16/09	DJH
Inventor Name Search (PALM)	3/16/09	DJH
Updated EAST Search (all databases)	7-14-09	DJH
Updated EAST Search (all databases)	3-5-10	DJH
Updated EAST Search (all databases)	7-3-12	DJH
Updated EAST Search (all databases)	11-13-12	DJH
Inventor Name Search (PALM)	11-13-12	DJH
712/220 (cursory)	11-13-12	DJH
703/26 (cursory)	11-13-12	DJH
Inventor Name Search (PALM)	7-14-13	DJH

INTERFERENCE SEARCH

	/DAVID J HUISMAN/ Primary Examiner.Art Unit 2183

LzLabs GmbH. Ex. 1004-70

US Class/ CPC Symbol	US Subclass / CPC Group	Date	Examiner
PGPub Text Search		11-13-12	DJH
PGPub Text Search		7-14-13	DJH

	/DAVID J HUISMAN/ Primary Examiner.Art Unit 2183

PTO/SB/30 (07-09)
Approved for use through 07/31/2012. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Request for Continued Examination (RCE) Transmittal

Address to:
Mail Stop RCE
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Application Number	11/668,623
Filing Date	January 30, 2007
First Named Inventor	Reid T. COPELAND et al.
Art Unit	2183
Examiner Name	David J. Huisman
Attorney Docket Number	POU920060152US1

This is a Request for Continued Examination (RCE) under 37 CFR 1.114 of the above-identified application.
Request for Continued Examination (RCE) practice under 37 CFR 1.114 does not apply to any utility or plant application filed prior to June 8, 1995, or to any design application. See Instruction Sheet for RCEs (not to be submitted to the USPTO) on page 2.

1. Submission required under 37 CFR 1.114 Note: If the RCE is proper, any previously filed unentered amendments and amendments enclosed with the RCE will be entered in the order in which they were filed unless applicant instructs otherwise. If applicant does not wish to have any previously filed unentered amendment(s) entered, applicant must request non-entry of such amendment(s).

   a. ☐ Previously submitted. If a final Office action is outstanding, any amendments filed after the final Office action may be considered as a submission even if this box is not checked.

      i. ☐ Consider the arguments in the Appeal Brief or Reply Brief previously filed on _____

      ii. ☐ Other _____

   b. ☑ Enclosed

      i. ☑ Amendment/Reply

      ii. ☐ Affidavit(s)/ Declaration(s)

      iii. ☐ Information Disclosure Statement (IDS)

      iv. ☐ Other _____

2. Miscellaneous

   a. ☐ Suspension of action on the above-identified application is requested under 37 CFR 1.103(c) for a period of _____ months. (Period of suspension shall not exceed 3 months; Fee under 37 CFR 1.17(i) required)

   b. ☐ Other _____

3. Fees The RCE fee under 37 CFR 1.17(e) is required by 37 CFR 1.114 when the RCE is filed.

   a. ☑ The Director is hereby authorized to charge the following fees, any underpayment of fees, or credit any overpayments, to Deposit Account No. 090463.

      i. ☐ RCE fee required under 37 CFR 1.17(e)

      ii. ☐ Extension of time fee (37 CFR 1.136 and 1.17)

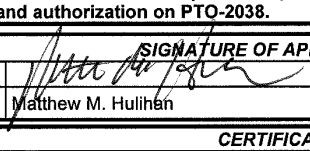
      iii. ☐ Other _____

   b. ☐ Check in the amount of $ _____ enclosed

   c. ☐ Payment by credit card (Form PTO-2038 enclosed)

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT REQUIRED

Signature		Date	October 29, 2012
Name (Print/Type)	Matthew M. Hulihan	Registration No.	68483

CERTIFICATE OF MAILING OR TRANSMISSION

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Mail Stop RCE, Commissioner for Patents, P. O. Box 1450, Alexandria, VA 22313-1450 or facsimile transmitted to the U.S. Patent and Trademark Office on the date shown below.

Signature			
Name (Print/Type)		Date	

This collection of information is required by 37 CFR 1.114. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Mail Stop RCE, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**
If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants: Reid T. COPELAND et al. Confirmation No.: 4180

Serial No.: 11/668,623 Group Art Unit: 2183

Filed: January 30, 2007 Examiner: David J. Huisman

Title: EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS WITHOUT EXECUTING BRANCH INSTRUCTIONS

To: Mail Stop RCE
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Amendment Accompanying Filing of Request for Continued Examination (RCE)

Dear Sir:

This paper is filed responsive to the final Office Action mailed July 31, 2012, in connection with the above-designated U.S. patent application and in support of the accompanying filed Request for Continued Examination. In addition, an Applicant Initiated Interview Request Form is co-filed herewith. Should any issues remain unresolved by this Amendment, a*pplicants respectfully request an Examiner Interview in order to discuss the Amendment in the hope of facilitating prosecution of the application.* Allowance of the application is respectfully requested in view of the Amendments and Remarks below.

Amendments to the Claims begin on page 2 of this paper.

Remarks begin on page 10 of this paper.

LzLabs GmbH. Ex. 1004-73

Amendments to the Claims

Kindly amend claims 9, 14, 15, 20, 21, 26, 27 & 30, cancel claim 17, and add new claim 31, as set forth below. All pending claims are reproduced below, with changes in the amended claims shown by underlining for added matter, and strikethrough/double brackets for deleted matter.

1-8. (Canceled)

9. (Currently Amended) A computer program product for emulating computer instructions from a source machine to produce sequences of instructions on a target machine, said computer program product comprising:

a non-transitory computer readable storage medium readable by a processor and storing instructions for execution by the processor for performing a method comprising:

obtaining by ~~a~~ the target machine a computer instruction from ~~a~~ the source machine, said source machine having a different architecture from said target machine; and

generating a sequence of target machine instructions which together operate to derive an encoding for a target machine condition code for the computer instruction, wherein the sequence of target machine instructions provides distinguishing information to distinguish between a plurality of possible outcomes for the target machine condition code, and directly calculates the target machine condition code ~~based, at least in part, on the distinguishing information~~ without using branch instructions, the directly calculating comprising:

determining an intermediate condition code value, the intermediate condition code value being a provisional value for the target machine condition code and subject to change based on the distinguishing information; and

determining, based on the intermediate condition code value and based on the distinguishing information, the target machine condition

LzLabs GmbH. Ex. 1004-74

code, wherein at least part of said distinguishing information ~~being~~ is separate from the intermediate condition code value.

10. (Previously Presented) The computer program product of claim 9, wherein said calculation of the target machine condition code comprises using at least one of carry, sign, and overflow codes in the calculation.

11. (Previously Presented) The computer program product of claim 10, wherein said calculation employs temporary locations for storing intermediate results.

12. (Previously Presented) The computer program product of claim 9, wherein said method further includes executing said generated sequence of target machine instructions.

13. (Previously Presented) The computer program product of claim 12, wherein said executing occurs at substantially the same time as said generating.

14. (Currently Amended) The computer program product of claim 30, wherein said first set of one or more possible outcomes comprises one case result and said second set of one or more possible outcomes comprises at least two other case results, and wherein said generating comprises:

mimicking an instruction present in a source instruction stream in a manner which sets one or more target ~~CPU~~ central processing unit (CPU) flag bits and which stores a first result in a first storage location;

executing an instruction which uses said stored first result to set the distinguishing information, the distinguishing information comprising at least one bit in a second storage location;

using said stored first result to provide a second result in the first storage location;

executing an instruction which employs said second result to produce a third result in the first storage location, the third result being the determined intermediate ~~result~~ condition code value which distinguishes between the one case result and the at least two other case results; and

LzLabs GmbH. Ex. 1004-75

executing an instruction which uses the third result in said first storage location and the at least one bit in the second storage location to provide an indication in which the one case result and the at least two other case results are distinguished, wherein using the at least one bit in the second storage location distinguishes between possible outcomes within the at least two other case results, and wherein the one case result and the at least two other case results represent condition codes of said source machine.

15. (Currently Amended) A computer system for emulating computer instructions from a source machine to produce sequences of instructions on a target machine, the computer system comprising:

a memory; and

a processor in communication with the memory, wherein the computer system is capable of performing a method, the method comprising:

obtaining by ~~a~~ the target machine a computer instruction from ~~a~~ the source machine, said source machine having a different architecture from said target machine; and

generating a sequence of target machine instructions which together operate to derive an encoding for a target machine condition code for the computer instruction, wherein the sequence of target machine instructions provides distinguishing information to distinguish between a plurality of possible outcomes for the target machine condition code, and directly calculates the target machine condition code ~~based, at least in part, on the distinguishing information~~ without using branch instructions, the directly calculating comprising:

determining an intermediate condition code value, the intermediate condition code value being a provisional value for the target machine condition code and subject to change based on the distinguishing information; and

LzLabs GmbH. Ex. 1004-76

determining, based on the intermediate condition code value and based on the distinguishing information, the target machine condition code, wherein at least part of said distinguishing information ~~being~~ is separate from the intermediate condition code value.

16. (Previously Presented) The computer system of claim 15, wherein said calculation of the target machine condition code comprises using at least one of carry, sign, and overflow codes in the calculation.

17. (Canceled)

18. (Previously Presented) The computer system of claim 15, wherein said method further includes executing said generated sequence of target machine instructions.

19. (Previously Presented) The computer system of claim 18, wherein said executing occurs at substantially the same time as said generating.

20. (Currently Amended) The computer system of claim 15, wherein the plurality of possible outcomes comprise a first set of one or more possible outcomes and a second set of one or more possible outcomes, the first set of one or more possible outcomes comprising one case result and the second set of one or more possible outcomes comprising at least two other case results, wherein the ~~sequence of target machine instructions provides an~~determined intermediate ~~result to distinguish~~condition code value distinguishes between the one case result and the at least two other case results, and wherein said generating comprises:

mimicking an instruction present in a source instruction stream in a manner which sets one or more target ~~CPU~~ central processing unit (CPU) flag bits and which stores a first result in a first storage location;

executing an instruction which uses said stored first result to set the distinguishing information, the distinguishing information comprising at least one bit in a second storage location;

using said stored first result to provide a second result in the first storage location;

LzLabs GmbH. Ex. 1004-77

executing an instruction which employs said second result to produce a third result in the first storage location, the third result being the determined intermediate ~~result~~ condition code value which distinguishes between the one case result and the at least two other case results; and

executing an instruction which uses the third result in said first storage location and the at least one bit in the second storage location to provide an indication in which the one case result and the at least two other case results are distinguished, wherein using the at least one bit in the second storage location distinguishes between possible outcomes within the at least two other case results, and wherein the one case result and the at least two other case results represent condition codes of said source machine.

21. (Currently Amended) A method for emulating computer instructions from a source machine to produce sequences of instructions on a target machine, said method comprising:

obtaining by ~~a~~ the target machine a computer instruction from ~~a~~ the source machine, said source machine having a different architecture from said target machine; and

generating a sequence of target machine instructions which together operate to derive an encoding for a target machine condition code for the computer instruction, wherein the sequence of target machine instructions provides distinguishing information to distinguish between a plurality of possible outcomes for the target machine condition code, and directly calculates the target machine condition code ~~based, at least in part, on the distinguishing information~~ without using branch instructions, the directly calculating comprising:

determining an intermediate condition code value, the intermediate condition code value being a provisional value for the target machine condition code and subject to change based on the distinguishing information; and

determining, based on the intermediate condition code value and based on the distinguishing information, the target machine condition code, wherein at least

LzLabs GmbH. Ex. 1004-78

part of said distinguishing information ~~being~~ is separate from the intermediate condition code value.

22. (Previously Presented) The method of claim 21, wherein said calculation of the target machine condition code comprises using at least one of carry, sign, and overflow codes in the calculation.

23. (Canceled)

24. (Previously Presented) The method of claim 21, wherein said method further includes executing said generated sequence of target machine instructions.

25. (Previously Presented) The method of claim 24, wherein said executing occurs at substantially the same time as said generating.

26. (Currently Amended) The method of claim 21, wherein the plurality of possible outcomes comprise a first set of one or more possible outcomes and a second set of one or more possible outcomes, the first set of one or more possible outcomes comprising one case result and the second set of one or more possible outcomes comprising at least two other case results, wherein the ~~sequence of target machine instructions provides an~~determined intermediate ~~result to distinguish~~condition code value distinguishes between the one case result and the at least two other case results, and wherein said generating comprises:

mimicking an instruction present in a source instruction stream in a manner which sets one or more target ~~CPU~~ central processing unit (CPU) flag bits and which stores a first result in a first storage location;

executing an instruction which uses said stored first result to set the distinguishing information, the distinguishing information comprising at least one bit in a second storage location;

using said stored first result to provide a second result in the first storage location;

executing an instruction which employs said second result to produce a third result in the first storage location, the third result being the determined intermediate ~~result~~

LzLabs GmbH. Ex. 1004-79

condition code value which distinguishes between the one case result and the at least two other case results; and

executing an instruction which uses the third result in said first storage location and the at least one bit in the second storage location to provide an indication in which the one case result and the at least two other case results are distinguished, wherein using the at least one bit in the second storage location distinguishes between possible outcomes within the at least two other case results, and wherein the one case result and the at least two other case results represent condition codes of said source machine.

27. (Currently Amended) The computer program product of claim 9, wherein the sequence of target machine instructions together operate to directly calculate the target machine condition code using at least one of addition, subtraction, and multiplication.

28. (Canceled)

29. (Previously Presented) The computer program product of claim 9, wherein the distinguishing information distinguishes between outcomes for the following cases: a first operand and a second operand of a target machine instruction of the sequence of target machine instructions are equal; the first operand is greater than the second operand; and the first operand is less than the second operand.

30. (Currently Amended) The computer program product of claim 9, wherein the plurality of possible outcomes comprise a first set of one or more possible outcomes and a second set of one or more possible outcomes, wherein the ~~sequence of target machine instructions provides an~~ determined intermediate ~~result to distinguish~~condition code value distinguishes between the first set of one or more possible outcomes and the second set of one or more possible outcomes, and wherein the distinguishing information ~~distinguishes~~ is applied to the intermediate condition code value to distinguish between possible outcomes within the first set of possible outcomes or within the second set of possible outcomes.

31. (New) The computer program product of claim 9, wherein determining the intermediate condition code value comprises storing the intermediate condition code value in a condition code storage location as the provisional target machine condition code, and wherein

LzLabs GmbH. Ex. 1004-80

determining the target machine condition code based on the distinguishing information and the intermediate condition code value comprises performing at least one operation on the intermediate condition code value stored in the condition code storage location to obtain the target machine condition code and storing the target machine condition code in the condition code storage location.

* * * * *

LzLabs GmbH. Ex. 1004-81

Remarks

Entry of the amendments, favorable consideration and allowance of the claims presented are respectfully requested. Claims 9-16, 18-22, 24-27 & 29-31 are now pending.

Initially, applicants thank the Examiner for the indication of allowability of claims 14, 20 & 26, if rewritten in independent form including all of the limitations of the base claim and any intervening claims. Presently, these dependent claims have not been rewritten into independent form since the amended independent claims from which they depend are believed to be in condition for allowance for the reasons explained below.

By this paper, claims 9, 14, 15, 20, 21, 26, 27 & 30 are amended, and claim 17 is canceled (without prejudice) for clarification purposes and not in acquiescence to the rejections. Additionally, new claim 31 is added. Support for these claim amendments and the new claim can be found throughout the application as filed. For example, reference specification paragraphs [0024]-[0032], and FIG. 2, in which a target machine condition code is calculated by, for instance, determining an intermediate condition code value (stored in rC just before execution of the last instruction in the sequence, in the examples presented). The intermediate condition code value is provisionally the target machine condition code, subject to being changed based on the distinguishing information. In the examples noted above, the intermediate condition code value might be incremented or decremented depending on the distinguishing information. No new matter is added to the application by any amendment presented.

Claim Objections:

Claims 9, 14, 15, 20, 21, 26 & 27 were objected to for informalities. These claims have been amended, and the amendments are believed to overcome these objections.

Claim Rejections – 35 U.S.C. §102, 103:

Substantively, prior claims 9-12, 15-18, 21, 22, 24, and 27-30 were rejected under 35 U.S.C. §102(b) as being anticipated by Liu et al. (U.S. Patent No. 6,049,864; hereinafter Liu), while claims 13, 19 & 25 were rejected under 35 U.S.C. §103(a) as being unpatentable over Liu in view of the Examiner's taking Official Notice. These rejections are respectfully traversed to

LzLabs GmbH. Ex. 1004-82

any extent deemed applicable to the claims presented herewith, and reconsideration thereof is requested.

In accordance with independent claim 1 presented herewith, a storage medium is disclosed which stores instructions for execution by a processor for performing a method that includes, for instance: generating a sequence of target machine instructions which together operate to derive an encoding for a target machine condition code for a computer instruction, wherein the sequence of target machine instructions provides distinguishing information to distinguish between a plurality of possible outcomes for the target machine condition code, and directly calculates the target machine condition code without using branch instructions. The directly calculating includes: determining an intermediate condition code value, the intermediate condition code value being a provisional value for the target machine condition code and subject to change based on the distinguishing information; and determining, based on the intermediate condition code value and based on the distinguishing information, the target machine condition code, wherein at least part of said distinguishing information is separate from the intermediate condition code value.

Thus, according to an aspect of applicants' recited invention, a target machine condition code is calculated without using branch instructions. The calculation includes determining an intermediate value for the target machine condition code. The intermediate value is subject to change based on distinguishing information which distinguishes between multiple possible outcomes for the target machine condition code. The target machine condition code is determined based on the distinguishing information and based on the intermediate condition code. To illustrate, refer to the tables presented in paragraphs [0025] and [0031] in applicants' specification. The target machine condition code, as a result of the sequence of instructions, will be held in register rC. An intermediate condition code value for that target machine condition code is determined (e.g. by performing the first four instructions in these examples) and placed in rC. That intermediate condition code value is a provisional value for the target machine condition code – that is, it is not the target machine condition code, *per se*, but rather a provisional value for the target machine condition code because it is subject to change based on the distinguishing information (the value stored in register rX in these examples). Based on performing the final instruction, in these examples, the target machine condition code is

LzLabs GmbH. Ex. 1004-83

determined and stored in rC, which may or may not be a modified version of the intermediate condition code value that was determined and stored in rC prior to performance of the final instruction.

A careful reading of the applied art fails to uncover any teaching or suggestion of an aspect of applicants' recited protocol. For instance, a careful reading of Liu fails to uncover any teaching or suggestion of determining an intermediate condition code value which is a provisional value for the target machine condition code and which is subject to change based on distinguishing information. Information in bits 32-38 of the X1 register, for example, is not a determined intermediate condition code value. The bits are not a provisional value for a target machine condition code, and they are not subject to change based on distinguishing information, as recited in an aspect of applicants' protocol. Instead, (some of) the bits in the X1 register are used to set, via the prodf32 instruction, carry, overflow, sign, and other flag bits in the AFLAGS register, which the Office Action interprets as being a condition code. However, without acquiescing to the Office Action's interpretation, the bits in the AFLAGS register are also not a determined intermediate condition code value, e.g. are not a provisional value for a target machine condition code which are subject to change based on distinguishing information. Instead, the bits in AFLAGS are set directly from the bits in the X1 register as part of the prodf32 instruction. As such, to the extent that the value(s) in the AFLAGS register are condition codes (as indicated in the Office Action, but without acquiescing to this interpretation) those values are set and are not intermediate (i.e. provisional values for the target machine condition code) which are subject to further change based on distinguishing information. As soon as the AFLAGS values are set by the prodf32 instruction, the final value for AFLAGS (and therefore the condition code, using the Office Action's interpretation), is determined.

Furthermore, a careful reading of Liu also fails to teach or suggest determining, based on an intermediate condition code value as recited above and based on the distinguishing information, a target machine condition code, as is further recited in claim 1 presented herewith. Since a careful reading of Liu fails to teach or suggest an intermediate condition code value as described above, a careful reading thereof cannot possibly teach or suggest determining a target machine condition code value based on such an intermediate condition code value and on other (distinguishing) information.

LzLabs GmbH. Ex. 1004-84

Since at least these features are missing from Liu, a careful reading thereof fails to describe, teach or suggest at least these aspects of applicants' claimed invention.

In accordance with independent claim 15 presented herewith, a computer system is provided for emulating computer instructions from a source machine to produce sequences of instructions on a target machine. The computer system includes a memory and a processor in communications with the memory, and is configured to perform a method which includes one or more of the following aspects, one or more of which is not described, taught, or suggested by the combination of references: obtaining by the target machine a computer instruction from the source machine, said source machine having a different architecture from said target machine; and generating a sequence of target machine instructions which together operate to derive an encoding for a target machine condition code for the computer instruction, wherein the sequence of target machine instructions provides distinguishing information to distinguish between a plurality of possible outcomes for the target machine condition code, and directly calculates the target machine condition code based without using branch instructions, the directly calculating comprising: determining an intermediate condition code value, the intermediate condition code value being a provisional value for the target machine condition code and subject to change based on the distinguishing information; and determining, based on the intermediate condition code value and based on the distinguishing information, the target machine condition code, wherein at least part of said distinguishing information is separate from the intermediate condition code value. Therefore, applicants respectfully request an indication of allowance for independent claim 15.

Additionally, in accordance with independent claim 21presented herewith, a method is provided for emulating computer instructions from a source machine to produce sequences of instructions on a target machine. T method includes one or more of the following aspects, one or more of which is not described, taught, or suggested by the combination of references: obtaining by the target machine a computer instruction from the source machine, said source machine having a different architecture from said target machine; and generating a sequence of target machine instructions which together operate to derive an encoding for a target machine condition code for the computer instruction, wherein the sequence of target machine instructions provides distinguishing information to distinguish between a plurality of possible outcomes for the target

LzLabs GmbH. Ex. 1004-85

machine condition code, and directly calculates the target machine condition code without using branch instructions, the directly calculating comprising: determining an intermediate condition code value, the intermediate condition code value being a provisional value for the target machine condition code and subject to change based on the distinguishing information; and determining, based on the intermediate condition code value and based on the distinguishing information, the target machine condition code, wherein at least part of said distinguishing information is separate from the intermediate condition code value. Therefore, applicants respectfully request an indication of allowance for independent claim 21.

The dependent claims are believed allowable for the same reasons as the independent claims, as well as for their own additional characterizations. For example, dependent claim 31 further recites:

- "wherein determining the intermediate condition code value comprises storing the intermediate condition code value in a condition code storage location as the provisional target machine condition code, and wherein determining the target machine condition code based on the distinguishing information and the intermediate condition code value comprises performing at least one operation on the intermediate condition code value stored in the condition code storage location to obtain the target machine condition code and storing the target machine condition code in the condition code storage location."

According to an aspect of applicants' recited protocol, the intermediate condition code value is stored as the provisional condition code (subject to change) in a condition code storage location, and the target machine condition code is determined by performing at least one operation on the intermediate condition code value (stored in the condition code storage location) to obtain the target machine condition code which is then stored in that condition code storage location. A careful reading of Liu fails to uncover any such teaching or suggestion. To the extent that Liu teaches applicants' recited target machine condition code, as alleged in the Office Action but without acquiescing to such allegation, the code is stored in the AFLAGS register. However, a careful reading of Liu fails to uncover any teaching or suggestion of storing an intermediate condition code value in that AFLAGS location, and fails to uncover any teaching or suggestion of determining a target machine condition code by performing at least one operation on an intermediate value stored in the AFLAGS location to obtain a target machine condition

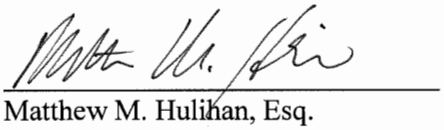
LzLabs GmbH. Ex. 1004-86

code stored in that location. For at least this additional reason, applicants respectfully submit that dependent claim 31 distinguishes over the cited art.

All claims are believed to be in condition for allowance, and such action is respectfully requested.

As noted, applicants respectfully request an Examiner Interview at the time the Examiner is considering this Response, in order to discuss the Response and facilitate prosecution of the application. An Applicant-Initiated Interview Request Form accompanies filing of this paper.

Respectfully submitted,



Matthew M. Hulihan, Esq.
Attorney for Applicants
Registration No.: 68,483

Dated: October 29, 2012.

HESLIN ROTHENBERG FARLEY & MESITI P.C.
5 Columbia Circle
Albany, New York 12203-5160
Telephone: (518) 452-5600
Facsimile: (518) 452-5579

LzLabs GmbH. Ex. 1004-87

Doc Code: M865 or FAI.REQ.INTV

PTOL-413A (08-10)
Approved for use through 07/31/2012. OMB 0651-0031
U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

Applicant Initiated Interview Request Form

Application No.: 12/668,623 First Named Applicant: Reid T. COPELAND, et al.
Examiner: David J. Huisman Art Unit: 2444 Status of Application: Pending

Tentative Participants:
(1) Blanche E. Schiller (2) Matthew M. Hulihan
(3) Examiner David J. Huisman (4) _____

Proposed Date of Interview: _____ **Proposed Time:** _____ **(AM/PM)**

Type of Interview Requested:
(1) [✓] Telephonic (2) [ ] Personal (3) [ ] Video Conference

Exhibit To Be Shown or Demonstrated: [ ] YES [✓] NO
If yes, provide brief description: _____

Issues To Be Discussed

Issues (Rej., Obj., etc)	Claims/ Fig. #s	Prior Art	Discussed	Agreed	Not Agreed
(1) 102(b) Rej.		Liu	[ ]	[ ]	[ ]
(2) 103(a)		Liu	[ ]	[ ]	[ ]
(3)			[ ]	[ ]	[ ]
(4)			[ ]	[ ]	[ ]

[ ] Continuation Sheet Attached [✓] Proposed Amendment or Arguments Attached
Brief Description of Arguments to be Presented: Discuss claim Amendments and any remaining issues the Examiner may have.

An interview was conducted on the above-identified application on _____

NOTE: This form should be completed and filed by applicant in advance of the interview (see MPEP § 713.01). If this form is signed by a registered practitioner not of record, the Office will accept this as an indication that he or she is authorized to conduct an interview on behalf of the principal (37 CFR 1.32(a)(3)) pursuant to 37 CFR 1.34. This is not a power of attorney to any above named practitioner. See the Instruction Sheet for this form, which is incorporated by reference. By signing this form, applicant or practitioner is certifying that he or she has read the Instruction Sheet. After the interview is conducted, applicant is advised to file a statement of the substance of this interview (37 CFR 1.133(b)) as soon as possible. This application will not be delayed from issue because of applicant's failure to submit a written record of this interview.

Matthew M. Hulihan
Applicant/Applicant's Representative Signature

Examiner/SPE Signature

Matthew M. Hulihan
Typed/Printed Name of Applicant or Representative

68,483
Registration Number, if applicable

This collection of information is required by 37 CFR 1.133. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 24 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

Electronic Patent Application Fee Transmittal

Application Number:	11668623
Filing Date:	30-Jan-2007
Title of Invention:	METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS
First Named Inventor/Applicant Name:	Reid T. Copeland
Filer:	Matthew M. Hulihan/Lynn Comproski
Attorney Docket Number:	POU920060152US1

Filed as Large Entity

Utility under 35 USC 111(a) Filing Fees

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Basic Filing:				
Pages:				
Claims:				
Miscellaneous-Filing:				
Petition:				
Patent-Appeals-and-Interference:				
Post-Allowance-and-Post-Issuance:				
Extension-of-Time:				

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Miscellaneous:				
Request for continued examination	1801	1	930	930
		Total in USD ($)		**930**

Electronic Acknowledgement Receipt	
EFS ID:	14097604
Application Number:	11668623
International Application Number:	
Confirmation Number:	4180
Title of Invention:	METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS
First Named Inventor/Applicant Name:	Reid T. Copeland
Customer Number:	46369
Filer:	Matthew M. Hulihan/Lynn Comproski
Filer Authorized By:	Matthew M. Hulihan
Attorney Docket Number:	POU920060152US1
Receipt Date:	29-OCT-2012
Filing Date:	30-JAN-2007
Time Stamp:	16:08:45
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	yes
Payment Type	Deposit Account
Payment was successfully received in RAM	$930
RAM confirmation Number	1901
Deposit Account	090463
Authorized User	
The Director of the USPTO is hereby authorized to charge indicated fees and credit any overpayment as follows: Charge any Additional Fees required under 37 C.F.R. Section 1.16 (National application filing, search, and examination fees) Charge any Additional Fees required under 37 C.F.R. Section 1.17 (Patent application and reexamination processing fees)	

Charge any Additional Fees required under 37 C.F.R. Section 1.21 (Miscellaneous fees and charges)					
File Listing:					
Document Number	**Document Description**	**File Name**	**File Size(Bytes)/ Message Digest**	**Multi Part /.zip**	**Pages (if appl.)**
1	Request for Continued Examination (RCE)	POU920060152US1_PR04_RCE_Transmittal.pdf	91549 e757a42262040b73e05d07f7dba0dc881c40630a	no	1
Warnings:					
This is not a USPTO supplied RCE SB30 form.					
Information:					
2		POU920060152US1_PR04_RCE_Amendment.pdf	789825 136bf9bc5fdcc2b2766d5d11ed48c5b3dffd2e2f	yes	15
	Multipart Description/PDF files in .zip description				
	Document Description		**Start**	**End**	
	Amendment Submitted/Entered with Filing of CPA/RCE		1	1	
	Claims		2	9	
	Applicant Arguments/Remarks Made in an Amendment		10	15	
Warnings:					
Information:					
3	First Action Interview - Schedule Interview request	POU920060152US1_PR04_InterviewRequestForm.pdf	76194 ac2336755a4afaf417ab976fadd5ad4de5d1a776	no	1
Warnings:					
Information:					
4	Fee Worksheet (SB06)	fee-info.pdf	30656 36d6dda1b63bca288cf6b62c33e31c666496b6ee	no	2
Warnings:					
Information:					
		Total Files Size (in bytes):	988224		

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office
If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

PTO/SB/06 (07-06)
Approved for use through 1/31/2007. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

PATENT APPLICATION FEE DETERMINATION RECORD

Substitute for Form PTO-875

Application or Docket Number	Filing Date	☐ To be Mailed
11/668,623	01/30/2007	

APPLICATION AS FILED – PART I

	(Column 1)	(Column 2)	SMALL ENTITY ☐		OR	OTHER THAN SMALL ENTITY	
FOR	NUMBER FILED	NUMBER EXTRA	RATE ($)	FEE ($)		RATE ($)	FEE ($)
☒ BASIC FEE (37 CFR 1.16(a), (b), or (c))	N/A	N/A	N/A			N/A	300
☐ SEARCH FEE (37 CFR 1.16(k), (i), or (m))	N/A	N/A	N/A			N/A	
☐ EXAMINATION FEE (37 CFR 1.16(o), (p), or (q))	N/A	N/A	N/A			N/A	
TOTAL CLAIMS (37 CFR 1.16(i))	minus 20 =	*	X $ =		OR	X $ =	
INDEPENDENT CLAIMS (37 CFR 1.16(h))	minus 3 =	*	X $ =			X $ =	
☐ APPLICATION SIZE FEE (37 CFR 1.16(s))	If the specification and drawings exceed 100 sheets of paper, the application size fee due is $250 ($125 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).						
☐ MULTIPLE DEPENDENT CLAIM PRESENT (37 CFR 1.16(j))							
* If the difference in column 1 is less than zero, enter "0" in column 2.			TOTAL			TOTAL	300

APPLICATION AS AMENDED – PART II

AMENDMENT	10/29/2012	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA	SMALL ENTITY RATE ($)	ADDITIONAL FEE ($)	OR	OTHER THAN SMALL ENTITY RATE ($)	ADDITIONAL FEE ($)
	Total (37 CFR 1.16(i))	* 20	Minus	** 20	= 0	X $ =		OR	X $62=	0
	Independent (37 CFR 1.16(h))	* 3	Minus	***3	= 0	X $ =		OR	X $250=	0
	☐ Application Size Fee (37 CFR 1.16(s))									
	☐ FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							OR		
						TOTAL ADD'L FEE		OR	TOTAL ADD'L FEE	0

AMENDMENT		(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA	RATE ($)	ADDITIONAL FEE ($)	OR	RATE ($)	ADDITIONAL FEE ($)
	Total (37 CFR 1.16(i))	*	Minus	**	=	X $ =		OR	X $ =	
	Independent (37 CFR 1.16(h))	*	Minus	***	=	X $ =		OR	X $ =	
	☐ Application Size Fee (37 CFR 1.16(s))									
	☐ FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							OR		
						TOTAL ADD'L FEE		OR	TOTAL ADD'L FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.

** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20".

*** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3".

The "Highest Number Previously Paid For" (Total or Independent) is the highest number found in the appropriate box in column 1.

Legal Instrument Examiner:
/TAMMY MCBETH BROWN/

This collection of information is required by 37 CFR 1.16. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
11/668,623	01/30/2007	Reid T. Copeland	POU920060152US1	4180

46369 7590 07/31/2012
HESLIN ROTHENBERG FARLEY & MESITI P.C.
5 COLUMBIA CIRCLE
ALBANY, NY 12203

EXAMINER	
HUISMAN, DAVID J	
ART UNIT	PAPER NUMBER
2183	
MAIL DATE	DELIVERY MODE
07/31/2012	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

LzLabs GmbH. Ex. 1004-95

Office Action Summary	Application No.	Applicant(s)	
	11/668,623	COPELAND ET AL.	
	Examiner	Art Unit	
	DAVID HUISMAN	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *13 June 2012*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ An election was made by the applicant in response to a restriction requirement set forth during the interview on _____; the restriction requirement and election have been incorporated into this action.

4)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

5)☒ Claim(s) *9-22,24-27,29 and 30* is/are pending in the application.

5a) Of the above claim(s) _____ is/are withdrawn from consideration.

6)☐ Claim(s) _____ is/are allowed.

7)☒ Claim(s) *9-13,15-19,21,22,24,25,27,29 and 30* is/are rejected.

8)☒ Claim(s) *14,20 and 26* is/are objected to.

9)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

10)☐ The specification is objected to by the Examiner.

11)☒ The drawing(s) filed on *16 June 2009* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

12)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

13)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

LzLabs GmbH. Ex. 1004-96

DETAILED ACTION

1. Claims 9-22, 24-27, and 29-30 have been examined.

Claim Objections

2. Claim 9 is objected to because of the following informalities:

- In line 7, there is a lack of antecedent basis for "said source machine" because applicant previously recites "a source machine" twice.
- In lines 7-8, there is a lack of antecedent basis for "said target machine" because applicant previously recites "a target machine" twice.

3. Claim 14 is objected to because of the following informalities:

- In line 6, replace "CPU" with --central processing unit (CPU)--. Applicant should avoid using initialisms/acronyms without first writing out what they stand for (since acronyms can take on different meanings over time).

4. Claim 15 is objected to because of the following informalities:

- In line 7, there is a lack of antecedent basis for "said source machine" because applicant previously recites "a source machine" twice.
- In lines 7-8, there is a lack of antecedent basis for "said target machine" because applicant previously recites "a target machine" twice.

5. Claim 20 is objected to because of the following informalities:

- In line 9, replace "CPU" with --central processing unit (CPU)--. Applicant should avoid using initialisms/acronyms without first writing out what they stand for (since acronyms can take on different meanings over time).

6. Claim 21 is objected to because of the following informalities:

- In lines 4-5, there is a lack of antecedent basis for "said source machine" because applicant previously recites "a source machine" twice.
- In line 5, there is a lack of antecedent basis for "said target machine" because applicant previously recites "a target machine" twice.

7. Claim 26 is objected to because of the following informalities:

- In line 9, replace "CPU" with --central processing unit (CPU)--. Applicant should avoid using initialisms/acronyms without first writing out what they stand for (since acronyms can take on different meanings over time).

8. Claim 27 is objected to because of the following informalities:

- In line 3, insert a comma after "subtraction".

Appropriate correction is required.

Claim Rejections - 35 USC § 102

9. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

10. Claims 9-12, 15-18, 21-22, 24, and 27-30 are rejected under 35 U.S.C. 102(b) as being anticipated by Liu et al., U.S. Patent No. 6,049,864 (herein referred to as Liu).

11. Referring to claim 21, Liu has taught a method for emulating computer instructions from a source machine to produce sequences of instructions on a target machine, said method comprising:

a) obtaining by a target machine a computer instruction from a source machine, said source machine having a different architecture from said target machine. See Fig.3 and claims 1 and 3 of Liu, and note that instruction ADDF32 X1 = X2 + X3 is a CISC instruction to be executed on a RISC machine.

b) generating a sequence of target machine instructions which together operate to derive an encoding for a target machine condition code for the computer instruction. See Fig.3 and column 3, lines 4-6, and note that a sequence of addf32 and prodf32 instructions are generated to emulate the CISC instruction and also set at least some of a series of condition codes shown in Figs.1A, 1B, and 3.

c) wherein the sequence of target machine instructions provides distinguishing information to distinguish between a plurality of possible outcomes for the condition code and directly calculates the condition code based, at least in part, on the distinguishing information, without using branch instructions, at least part of said distinguishing information being separate from the condition code. See Fig.3, and note that the emulation of the ADDF32 source instruction results in the execution of a sequence of multiple non-branch instructions that provide distinguishing information (bits 32-38 of X1), which are then used, at least in part, to set carry, overflow, sign, and other flags, any one or more of which may be considered the condition code. Bits 32-38 of X1 are clearly separate from the condition code(s), and since the condition codes may vary based

on the values of the distinguishing information (bits 32-38 of X1), the distinguishing information distinguishes between a plurality of possible outcomes for the condition code(s).

12. Referring to claim 22, Liu has taught the method of claim 21 wherein said calculation of the target machine condition code comprises using at least one of carry, sign, and overflow codes in the calculation. See Fig.3 and note, in the bottom left box, that carry codes (e.g. X1{32}, X1{33}, …) are used in calculation the condition code shown in the bottom right box of Fig.3.

13. Referring to claim 24, Liu has taught the method of claim 21 wherein said method further includes executing said generated sequence of target machine instructions. See Fig.3.

14. Referring to claims 9-10 and 12, claims 9-10 and 12 are rejected for the same reasons set forth in the rejections of claims 21-22 and 24, respectively.

15. Referring to claim 11, Liu has taught the computer program product of claim 10, wherein said calculation employs temporary locations for storing intermediate results. See Fig.3, and note that in response to the addf32 instruction, not only is the result produced and stored in X1{31:0}, but other intermediate values are stored in X1{32:38}. These intermediate results are then used by the prodf32 instruction to set flags.

16. Referring to claims 15-16 and 18, the systems of claims 15-16 and 18 perform the methods of claims 21-22 and 24, respectively. Consequently, claims 15-16 and 18 are rejected for the same reasons set forth in the rejections of claims 21-22 and 24, respectively. Also, it should be noted that Liu has taught a memory, both inherently and explicitly. That is, in Fig.3, Liu shows a memory X1[38:0] for storing data. Also, program instructions are inherently stored on some medium so that the processor may fetch and process them.

17. Referring to claim 17, Liu has taught the computer system of claim 16. Furthermore, claim 17 is rejected for the same reasons set forth in the rejection of claim 11.

18. Referring to claim 27, Liu has taught the computer program product of claim 9, wherein the sequence of target machine instructions together operate to directly calculate the target machine condition code using at least one of addition, subtraction and multiplication. See Fig.3, and note the addition operation. Also, one of ordinary skill in the art would recognize that a subtraction and multiplication would be handled in a similar fashion.

19. Referring to claim 29, Liu has taught the computer program product of claim 9, wherein the distinguishing information distinguishes between outcomes for the following cases: a first operand and a second operand of a target machine instruction of the sequence of target machine instructions are equal; the first operand is greater than the second operand; and the first operand is less than the second operand. Note that the distinguishing information could include bits 31:0 of X1. Using a subtract instruction to compare two operands, if X1[31:0] = 0 after performing operand1 minus operand2, then this distinguishing information indicates equality between the operands. If X1[31] is 1, then operand1 is less than operand2 (since the subtraction result would be negative), and if X1[31] is 0 and any of X1[30:0] is 1, then operand1 is greater than operand2. Hence, the X1 information distinguishes between these cases.

20. Referring to claim 30, Liu has taught the computer program product of claim 9, wherein the plurality of possible outcomes comprise a first set of one or more possible outcomes (Fig.3, a first set of outcomes could be at least when the aux and carry flags take on the values 00 or 01, combined) and a second set of one or more possible outcomes (Fig.3, a second set of outcomes could be at least when the aux and carry flags take on the values 10 or 11, combined), wherein

the sequence of target machine instructions provides an intermediate result (Fig.3, left box, bits 32-38 of X1) to distinguish between the first set of one or more possible outcomes and the second set of one or more possible outcomes (from Fig.3, it is clear that bits 32 and 38 of X1, i.e., at least some of the distinguishing information, distinguish between outcomes. Based on the values of these bits, different outcomes will occur), and wherein the distinguishing information distinguishes between possible outcomes within the first set of possible outcomes or within the second set of possible outcomes (see Fig.3. Again, based on how bits 32 and 28 are set, the distinguishing information will distinguish between the first and second outcomes in each set).

Claim Rejections - 35 USC § 103

21. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

22. Claims 13, 19, and 25 are rejected under 35 U.S.C. 103(a) as being unpatentable over Liu in view of the examiner's taking of Official Notice.

23. Referring to claim 25, Liu has taught the method of claim 24. Liu has not explicitly taught that said executing occurs at substantially the same time as said generating. However, the examiner asserts that interpretive environments, in which instructions are translated at run-time, are well known and accepted in the art. Such systems reduce the amount of work done by the compiler and therefore, reduce compilation time. As a result, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Liu such that the generating of

the sequence of instructions occurs at run-time when the source instruction is encountered, thereby resulting in execution occurring at substantially the same time as said generating.

24. Referring to claim 13, claim 13 is rejected for the same reasons set forth in the rejection of claim 25.

25. Referring to claim 19, the system of claim 19 performs the method of claim 25. Consequently, claim 19 is rejected for the same reasons set forth in the rejection of claim 25.

Allowable Subject Matter

26. Claims 14, 20, and 26 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

Response to Arguments

27. Most of applicant's arguments, though fully considered, are moot in view of the new grounds of rejection necessitated by applicant's amendments. However, on page 12, applicant does argue that Liu has not taught "obtaining by a target machine a computer instruction from a source machine." In response, the examiner asserts that applicant is reading this limitation too narrowly. The target machine is the machine of Liu, and it obtains computer instruction ADDF32. This instruction is not native to Liu's machine, which is why it is emulated. However, ADDF32 is native to a CISC machine. Hence, while Liu may not inherently teach transmission of the ADDF32 instruction from a source machine to Liu's machine, transmission is not required by the claims. All that is required is that Liu's machine obtain the instruction, which it does, and

that the obtained instruction is from some source machine (meaning, it is native to some other machine in existence), which is the case, otherwise it wouldn't exist in the first place. Alternatively, it could be said that wherever the program was written and/or compiled/assembled into CISC code is the source machine. Please note that even if transmission was required by the claim, the examiner asserts that this is not a patentable feature as transmission of code between computers is known in the art. Therefore, the examiner recommends focusing more on the inventive concept involving the condition code and distinguishing information, than on minor details such as how the instruction is obtained.

Conclusion

28. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to DAVID HUISMAN whose telephone number is (571)272-4168. The examiner can normally be reached on Monday-Friday (9:00-6:00).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/David J. Huisman/
Primary Examiner, Art Unit 2183

Search Notes	Application/Control No.	Applicant(s)/Patent Under Reexamination
	11668623	COPELAND ET AL.
	Examiner	**Art Unit**
	DAVID J HUISMAN	2183

SEARCHED			
Class	**Subclass**	**Date**	**Examiner**
712	209, 221, 227	3-16-09	DJH
717	124, 134, 136, 137, 138	3-16-09	DJH
703	26	3-16-09	DJH

SEARCH NOTES		
Search Notes	**Date**	**Examiner**
EAST Search (all databases)	3/16/09	DJH
Inventor Name Search (PALM)	3/16/09	DJH
Updated EAST Search (all databases)	7-14-09	DJH
Updated EAST Search (all databases)	3-5-10	DJH
Updated EAST Search (all databases)	7-3-12	DJH

INTERFERENCE SEARCH			
Class	**Subclass**	**Date**	**Examiner**

	/DAVID HUISMAN/ Primary Examiner.Art Unit 2183

LzLabs GmbH. Ex. 1004-106

EAST Search History

EAST Search History (Prior Art)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	1	distinguish$3 with (outcome categor$4) with condition near2 (code flag)	US-PGPUB; USPAT	OR	ON	2012/07/03 14:35
L2	86	categor$4 with condition near2 (code flag)	US-PGPUB; USPAT	OR	ON	2012/07/03 14:35
L3	8	categor$4 with condition near2 (code flag) and "712"/$.ccls.	US-PGPUB; USPAT	OR	ON	2012/07/03 14:35
L5	8	categor$4 with condition near2 (code flag)	USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2012/07/03 14:37
S42	1	"20030093649".pn.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S43	86	emulat$3 with "condition code"	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S44	102	emulat$3 with ("condition code" "flag bit" "status bit") and (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S45	29	emulat$3 with ("condition code" "flag bit" "status bit") same branch and (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S46	68	("20020026633" | "4514803" | "4812975" | "5313614" | "5329471" | "5408637" | "5440702" | "5551015" | "5574873" | "5574927" | "5577233" | "5598560" | "5606696" | "5649203" | "5724590" | "5737631" | "5751982" | "5768593" | "5805792" | "5805895" | "5870575" | "5890000" | "5901308" | "5903760" | "5925124" | "5930509" | "5946491" | "5956495" | "5966539" | "5983335" | "6000028" | "6009261" | "6031994" | "6076155" | "6077314" | "6091897" | "6105124" | "6126329" | "6170083" | "6185675" | "6199095" | "6205545" | "6223339" | "6301705" | "6308318" | "6321379" | "6330691" | "6360194" | "6367067" | "6427234" | "6434741" | "6463582" | "6477641" | "6502237" | "6505296" | "6522985" | "6523173" | "6526572" | "6530078" | "6530079" | "6535903" | "6539471" | "6539541" | "6549987" | "6564297" | "6594824" | "6631514" | "6643630").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S47	6	(emulat$3 translat$3) same ("condition code" "flag bit" "status bit") same (without avoid$4) near2 branch$3 and (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S48	144	(emulat$3 translat$3) with (without avoid$4) near2 branch$3 and (overflow	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-107

		carry sign)				
S49	12	(emulat$3 translat$3) with (without avoid$4) near2 branch$3 and (overflow carry sign) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S50	5	("4951195" | "5241678" | "5301325" | "5313614" | "5598560").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S51	28	("5903760").URPN.	USPAT	OR	ON	2009/03/16 16:40
S52	68	("20020026633" | "4514803" | "4812975" | "5313614" | "5329471" | "5408637" | "5440702" | "5551015" | "5574873" | "5574927" | "5577233" | "5598560" | "5606696" | "5649203" | "5724590" | "5737631" | "5751982" | "5768593" | "5805792" | "5805895" | "5870575" | "5890000" | "5901308" | "5903760" | "5925124" | "5930509" | "5946491" | "5956495" | "5966539" | "5983335" | "6000028" | "6009261" | "6031994" | "6076155" | "6077314" | "6091897" | "6105124" | "6126329" | "6170083" | "6185675" | "6199095" | "6205545" | "6223339" | "6301705" | "6308318" | "6321379" | "6330691" | "6360194" | "6367067" | "6427234" | "6434741" | "6463582" | "6477641" | "6502237" | "6505296" | "6522985" | "6523173" | "6526572" | "6530078" | "6530079" | "6535903" | "6539471" | "6539541" | "6549987" | "6564297" | "6594824" | "6631514" | "6643630").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S53	68	("20020026633" | "4514803" | "4812975" | "5313614" | "5329471" | "5408637" | "5440702" | "5551015" | "5574873" | "5574927" | "5577233" | "5598560" | "5606696" | "5649203" | "5724590" | "5737631" | "5751982" | "5768593" | "5805792" | "5805895" | "5870575" | "5890000" | "5901308" | "5903760" | "5925124" | "5930509" | "5946491" | "5956495" | "5966539" | "5983335" | "6000028" | "6009261" | "6031994" | "6076155" | "6077314" | "6091897" | "6105124" | "6126329" | "6170083" | "6185675" | "6199095" | "6205545" | "6223339" | "6301705" | "6308318" | "6321379" | "6330691" | "6360194" | "6367067" | "6427234" | "6434741" | "6463582" | "6477641" | "6502237" | "6505296" | "6522985" | "6523173" | "6526572" | "6530078" | "6530079" | "6535903" | "6539471" | "6539541" | "6549987" | "6564297" | "6594824" | "6631514" | "6643630").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S54	723	(emulat$3 translat$3) with compar$4 with instruction	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S55	21	(emulat$3 translat$3) with compar$4 with instruction same (code flag) near2 (condition status bit)	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-108

S56	412	712/209.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S57	487	712/221.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S58	867	712/227.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S59	897	717/136.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S60	265	717/137.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S61	234	717/138.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S62	1	emulat$3 with ("condition code" "flag bit" "status bit") and (overflow carry sign)	USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2009/03/16 16:40
S63	1362	717/124.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S64	140	717/134.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S65	1560	(emulat$3 translat$3) same (arithmetic add$5 subtract$3) same (code flag) near2 (condition status bit)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S66	101	(emulat$3 translat$3) same (arithmetic add$5 subtract$3) with set$4 with (code flag) near2 (condition status bit)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S67	37	(emulat$3 translat$3) same (arithmetic add$5 subtract$3) with set$4 with (code flag) near2 (condition status bit) same (Z O S C sign zero overflow carry)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S68	29	(emulat$3 translat$3) same (arithmetic add$5 subtract$3) with set$4 with (code flag) near2 (condition status bit) same (Z O S C sign zero overflow carry) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S69	7	("20020046298" "20040176941" "20050015754" "20070011664" "5301302" "5678032" "5966515").pn.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S70	4	S69 and (code flag) near2 (condition status bit)	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S71	119	simulat$3 near4 (code flag) near2 (condition status bit)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S72	67	simulat$3 near4 (code flag) near2 (condition status bit) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S73	35	simulat$3 near4 (code flag) near2 (condition status bit) and (overflow carry) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-109

S74	127	(emulat$3 translat$3) and (host target) with set$4 with (code flag) near2 (condition status bit) and ((overflow O) and (carry C) and (sign S negative N)) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S75	59	(emulat$3 translat$3) and (host target) with set$4 with (code flag) near2 (condition status bit) and ((overflow O) and (carry C) and (sign S negative N)) and ("712"/$.ccls. "717"/$.ccls.)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S76	32	emulat$3 and (host target) with set$4 with (code flag) near2 (condition status bit) and ((overflow O) and (carry C) and (sign S negative N)) and ("712"/$.ccls. "717"/$.ccls.)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S77	1	emulat$3 and (host target) with set$4 with (code flag) near2 (condition status bit) same ((overflow O) and (carry C) and (sign S negative N)) and ("712"/$.ccls. "717"/$.ccls.)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S78	1	emulat$3 and (host target) with set$4 with (code flag) near2 (condition status bit) same (((overflow O) and (carry C) and (sign S negative N))) and ("712"/$.ccls. "717"/$.ccls.)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S79	6	("4415969" | "5327537" | "5440749" | "5493687" | "5590359" | "5689720").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S80	2	("5774694").URPN.	USPAT	OR	ON	2009/03/16 16:40
S81	8	("5313614" | "5481693" | "5619664" | "5774694" | "5859999" | "5860017" | "5884059" | "6049864").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S82	1	"20060173931".pn.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:44
S83	446	703/26.ccls.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:46
S84	20	emulat$3 with ("condition code" "flag bit" "status bit") same (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/07/14 17:52
S85	0	emulat$3 with ("condition code" "flag bit" "status bit") same (overflow carry sign)	USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2009/07/14 17:53
S86	47	huisman.xp.	US-PGPUB; USPAT	OR	ON	2010/03/04 09:27
S87	1913	subtract$3 with carry with (bit flag)	US-PGPUB; USPAT	OR	ON	2010/03/04 11:03
S88	186	subtract$3 with set$4 with carry adj (bit flag)	US-PGPUB; USPAT	OR	ON	2010/03/04 11:03
S89	112	subtract$3 with set$4 with carry adj (bit flag) and "712"/$.ccls.	US-PGPUB; USPAT	OR	ON	2010/03/04 11:03
S90	2	("5574931" "5999976").pn.	US-PGPUB; USPAT	OR	ON	2010/03/04 15:04

EAST Search History (Interference)

<This search history is empty>

LzLabs GmbH. Ex. 1004-110

7/3/2012 2:39:06 PM
C:\Users\dhuisman\Documents\EAST\Workspaces\11-668623.wsp

LzLabs GmbH. Ex. 1004-111

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants: Reid COPELAND et al. Confirmation No.: 4180

Serial No.: 11/668,623 Group Art Unit: 2183

Filed: January 30, 2007 Examiner: David Huisman

Title: METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS

To: Mail Stop Amendment
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Response to Office Action

Dear Sir:

This paper is filed responsive to the Office Action mailed March 14, 2012, in connection with the above-designated U.S. patent application. In addition, this paper is filed in accordance with the May 31, 2012, telephone conference between the Examiner and applicants' undersigned representative. *Should any issue remain unresolved by this paper, applicants respectfully request that the Examiner telephone their undersigned representative in order to discuss the issue in the hope of facilitating prosecution of the application.* Allowance of the application is respectfully requested in view of the Amendments and Remarks below.

Amendments to the Specification appear on page 2 of this paper.

Amendments to the Claims begin on page 3 of this paper.

Remarks begin on page 10 of this paper.

LzLabs GmbH. Ex. 1004-112

Amendments to the Specification

Please amend specification paragraph [0036] as follows:

[0036] The typical emulation environment in which the present invention is employed is illustrated in Figure 5. Emulators such as 320 ~~accepts~~accept, as input, instruction streams 305 representing machine or assembly language instructions which are designed to operate on source machine 300. Emulator 320 employees memory 315 in target machine 310 to produce a stream of instructions which are capable of executing on target machine 310. While Figure 5 particularly shows operation within an emulation environment, it is also noted that the present invention contemplates a situation in which emulator 320 operates essentially as an interpreter in which the instructions are not only translated to the new architecture but in which they are also executed at essentially the same time.

LzLabs GmbH. Ex. 1004-113

Amendments to the Claims

Kindly amend claims 9, 10, 14-16, 20-22 & 26, cancel claim 28 (without prejudice), and add new claim 30, as set forth below. All pending claims are reproduced below, with changes in the amended claims shown by underlining for added matter, and strikethrough/double brackets for deleted matter.

1-8. (Canceled)

9. (Currently Amended) A computer program product for emulating computer instructions from a source machine to produce sequences of instructions on a target machine, said computer program product comprising:

a non-transitory computer readable storage medium readable by a processor and storing instructions for execution by the processor for performing a method comprising:

obtaining by a target machine a computer instruction from a source machine, said source machine having a different architecture from said target machine; and

generating a sequence of target machine instructions which together operate to derive an encoding for a target machine condition code for the computer instruction, wherein the sequence of target machine instructions ~~directly calculates the condition code and~~ provides distinguishing information to distinguish between a plurality of possible outcomes for the condition code and directly calculates the condition code based, at least in part, on the distinguishing information without using branch instructions, at least part of said distinguishing information being separate from the condition code.

10. (Currently Amended) The computer program product of claim 9, wherein said calculation of the target machine condition code comprises using at least one of carry, sign, and overflow codes in the calculation.

11. (Previously Presented) The computer program product of claim 10, wherein said calculation employs temporary locations for storing intermediate results.

LzLabs GmbH. Ex. 1004-114

12. (Previously Presented) The computer program product of claim 9, wherein said method further includes executing said generated sequence of target machine instructions.

13. (Previously Presented) The computer program product of claim 12, wherein said executing occurs at substantially the same time as said generating.

14. (Currently Amended) The computer program product of claim ~~9~~30, wherein said first set of one or more possible outcomes comprises one case result and said second set of one or more possible outcomes comprises at least two other case results, and wherein said generating comprises:

mimicking an instruction present in a source instruction stream in a manner which sets one or more target CPU flag bits and which ~~places~~ stores a first result in a first storage location;

executing an instruction which uses said stored first result to set the distinguishing information, the distinguishing information comprising at least one bit in a second storage location ~~so as to distinguish one or more case results~~;

using said stored first result to provide a ~~shorter bit configuration which serves to distinguish at least one of said case results~~second result in the first storage location;

executing an instruction which employs said ~~first storage location~~second result to produce a third result in the first storage location, the third result being the intermediate result which distinguishes ~~a different set of~~between the one case result and the at least two other case results; and

executing an instruction which uses the ~~results~~ third result in said first ~~and~~ storage location and the at least one bit in the second storage ~~locations~~ location to provide an indication in which ~~at least three cases~~the one case result and the at least two other case results are distinguished, ~~said cases representing~~wherein using the at least one bit in the second storage location distinguishes between possible outcomes within the at least two other case results, and wherein the one case result and the at least two other case results represent condition codes of said source machine.

LzLabs GmbH. Ex. 1004-115

15. (Currently Amended) A computer system for emulating computer instructions from a source machine to produce sequences of instructions on a target machine, the computer system comprising:

a memory; and

a processor in ~~communications~~ communication with the memory, wherein the computer system is capable of performing a method, the method comprising:

obtaining by a target machine a computer instruction from a source machine, said source machine having a different architecture from said target machine; and

generating a sequence of target machine instructions which together operate to derive an encoding for a target machine condition code for the computer instruction, wherein the sequence of target machine instructions ~~directly calculates the condition code and~~ provides distinguishing information to distinguish between a plurality of possible outcomes for the condition code and directly calculates the condition code based, at least in part, on the distinguishing information without using branch instructions, at least part of said distinguishing information being separate from the condition code.

16. (Currently Amended) The computer system of claim 15, wherein said calculation of the target machine condition code comprises using at least one of carry, sign, and overflow codes in the calculation.

17. (Previously Presented) The computer system of claim 16, wherein said calculation employs temporary locations for storing intermediate results.

18. (Previously Presented) The computer system of claim 15, wherein said method further includes executing said generated sequence of target machine instructions.

19. (Previously Presented) The computer system of claim 18, wherein said executing occurs at substantially the same time as said generating.

LzLabs GmbH. Ex. 1004-116

20. (Currently Amended) The computer system of claim 15, wherein the plurality of possible outcomes comprise a first set of one or more possible outcomes and a second set of one or more possible outcomes, the first set of one or more possible outcomes comprising one case result and the second set of one or more possible outcomes comprising at least two other case results, wherein the sequence of target machine instructions provides an intermediate result to distinguish between the one case result and the at least two other case results, and wherein said generating comprises:

mimicking an instruction present in a source instruction stream in a manner which sets one or more target CPU flag bits and which ~~places~~ stores a first result in a first storage location;

executing an instruction which uses said stored first result to set the distinguishing information, the distinguishing information comprising at least one bit in a second storage location ~~so as to distinguish one or more case results~~;

using said stored first result to provide a ~~shorter bit configuration which serves to distinguish at least one of said case results~~second result in the first storage location;

executing an instruction which employs said ~~first storage location~~second result to produce a third result in the first storage location, the third result being the intermediate result which distinguishes ~~a different set of~~between the one case result and the at least two other case results; and

executing an instruction which uses the ~~results~~ third result in said first ~~and~~ storage location and the at least one bit in the second storage ~~locations~~ location to provide an indication in which ~~at least three cases~~the one case result and the at least two other case results are distinguished, ~~said cases representing~~wherein using the at least one bit in the second storage location distinguishes between possible outcomes within the at least two other case results, and wherein the one case result and the at least two other case results represent condition codes of said source machine.

LzLabs GmbH. Ex. 1004-117

21. (Currently Amended) A method for emulating computer instructions from a source machine to produce sequences of instructions on a target machine, said method comprising:

obtaining by a target machine a computer instruction from a source machine, said source machine having a different architecture from said target machine; and

generating a sequence of target machine instructions which together operate to derive an encoding for a target machine condition code for the computer instruction, wherein the sequence of target machine instructions ~~directly calculates the condition code and~~ provides distinguishing information to distinguish between a plurality of possible outcomes for the condition code and directly calculates the condition code based, at least in part, on the distinguishing information without using branch instructions, at least part of said distinguishing information being separate from the condition code.

22. (Currently Amended) The method of claim 21, wherein said calculation of the target machine condition code comprises using at least one of carry, sign, and overflow codes in the calculation.

23. (Canceled)

24. (Previously Presented) The method of claim 21, wherein said method further includes executing said generated sequence of target machine instructions.

25. (Previously Presented) The method of claim 24, wherein said executing occurs at substantially the same time as said generating.

26. (Currently Amended) The method of claim 21, wherein the plurality of possible outcomes comprise a first set of one or more possible outcomes and a second set of one or more possible outcomes, the first set of one or more possible outcomes comprising one case result and the second set of one or more possible outcomes comprising at least two other case results, wherein the sequence of target machine instructions provides an intermediate result to distinguish between the one case result and the at least two other case results, and wherein said generating comprises:

LzLabs GmbH. Ex. 1004-118

mimicking an instruction present in a source instruction stream in a manner which sets one or more target CPU flag bits and which ~~places~~ stores a first result in a first storage location;

executing an instruction which uses said stored first result to set the distinguishing information, the distinguishing information comprising at least one bit in a second storage location ~~so as to distinguish one or more case results~~;

using said stored first result to provide a ~~shorter bit configuration which serves to distinguish at least one of said case results~~second result in the first storage location;

executing an instruction which employs said ~~first storage location~~second result to produce a third result in the first storage location, the third result being the intermediate result which distinguishes ~~a different set of~~between the one case result and the at least two other case results; and

executing an instruction which uses the ~~results~~ third result in said first ~~and~~ storage location and the at least one bit in the second storage ~~locations~~ location to provide an indication in which ~~at least three cases~~ the one case result and the at least two other case results are distinguished, ~~said cases representing~~ wherein using the at least one bit in the second storage location distinguishes between possible outcomes within the at least two other case results, and wherein the one case result and the at least two other case results represent condition codes of said source machine.

27. (Previously Presented) The computer program product of claim 9, wherein the sequence of target machine instructions together operate to directly calculate the target machine condition code using at least one of addition, subtraction and multiplication.

28. (Canceled)

29. (Previously Presented) The computer program product of claim 9, wherein the distinguishing information distinguishes between outcomes for the following cases: a first operand and a second operand of a target machine instruction of the sequence of target machine

LzLabs GmbH. Ex. 1004-119

instructions are equal; the first operand is greater than the second operand; and the first operand is less than the second operand.

30. (New) The computer program product of claim 9, wherein the plurality of possible outcomes comprise a first set of one or more possible outcomes and a second set of one or more possible outcomes, wherein the sequence of target machine instructions provides an intermediate result to distinguish between the first set of one or more possible outcomes and the second set of one or more possible outcomes, and wherein the distinguishing information distinguishes between possible outcomes within the first set of possible outcomes or within the second set of possible outcomes.

* * * * *

LzLabs GmbH. Ex. 1004-120

Remarks

Entry of the amendments, favorable consideration, and allowance of the claims presented are respectfully requested. Claims 9-22, 24-27, 29 & 30 are now pending.

Applicants gratefully acknowledge the time afforded applicants' attorney, Blanche Schiller, and Matthew Hulihan during the telephone interview with Examiner Huisman on May 31, 2012. During the interview, the Examiner indicated that claim 1 presented herewith appears to overcome the current rejection and that further consideration would be given to the applied art. *Should any issue remain unresolved by this paper, applicants respectfully request that the Examiner telephone their undersigned representative to discuss the matter in the hope of advancing prosecution of the subject application.*

By this paper, claims 9, 10, 14-16, 20-22 & 26 are amended for clarification purposes and not in acquiescence to the rejections. Additionally, claim 28 is canceled (without prejudice) and claim 30 is added. Support for these claim amendments and the new claim can be found throughout the application as filed. For example, reference specification paragraphs [0021], [0023], [0025], [0027]-[0029], [0031] & [0032], and FIG. 2. No new matter is added to the application by any amendment presented.

Specification Objection:

Applicants have amended paragraph [0036] of the specification to replace "accepts" with --accept--, as was suggested by the Examiner. The amendment to paragraph [0036] is believed to address the specification objection on page 2 of the Office Action.

Claim Objections:

Applicants have amended claim 15 to replace "communications" with --communication-- and have amended claims 10, 15 & 22 to insert a comma after "sign", as suggested by the Examiner. These amendments to claims 10, 15 & 22 are believed to address the claim objections on page 2 of the Office Action.

LzLabs GmbH. Ex. 1004-121

Claim Rejections – 35 U.S.C. §112:

Claims 14, 20 & 26 are amended and no longer recite "using said result to provide a shorter bit configuration". As such, reconsideration and withdrawal of the rejection to claims 14, 20 & 26 are respectfully requested.

Claim Rejections – 35 U.S.C. §101:

Claims 9-14 were rejected under 35 U.S.C. §101 as being directed to nonstatutory subject matter. Without acquiescing to this rejection, independent claim 9 is amended to recite, "A computer program product…comprising: a non-transitory computer readable storage medium…", which further recites statutory subject matter. Reconsideration and withdrawal of the rejection to claims 9-14 under 35 U.S.C. §101 are respectfully requested.

Claim Rejections – 35 U.S.C. §102:

Substantively, prior claims 9-12, 14-18, 20-22, 24, and 26-28 were rejected under 35 U.S.C. §102(b) as being anticipated by Liu et al. (U.S. Patent No. 6,046,864; hereinafter Liu), while claims 13, 19, 25 & 29 were rejected under 35 U.S.C. §103(a) as being unpatentable over Liu in view of the Examiner's taking of Official Notice. These rejections are respectfully traversed to any extent deemed applicable to the claims presented herewith, and reconsideration thereof is requested.

In accordance with claim 9 presented herewith, a computer program product is recited for emulating computer instructions from a source machine to produce sequences of instructions on a target machine. The computer program product includes a non-transitory computer readable storage medium readable by a processor and storing instructions for execution by the processor for performing a method that includes, for instance, obtaining by a target machine a computer instruction from a source machine, the source machine having a different architecture from said target machine, and generating a sequence of target machine instructions which together operate to derive an encoding for a target machine condition code for the computer instruction, wherein the sequence of target machine instructions provides distinguishing information to distinguish between a plurality of possible outcomes for the condition code and directly calculates the condition code based, at least in part, on the distinguishing information without using branch

LzLabs GmbH. Ex. 1004-122

instructions, at least part of said distinguishing information being separate from the condition code.

Thus, in accordance with claim 9 presented herewith, a sequence of target machine instructions provides distinguishing information to distinguish between a plurality of possible outcomes for a condition code and directly calculates the condition code, based, at least in part, on this distinguishing information. Applicants respectfully submit that a careful reading of the applied art fails to uncover any such teaching or suggestion. Liu places indications of carry, sign, overflow, etc in an AFLAGS register (see FIG. 3). Cited in the Office Action for an alleged teaching of distinguishing information is column 4, lines 53-63. Applicants respectfully traverse this citation as applied to claim 9 presented herewith. A careful reading of Liu fails to uncover any teaching or suggestion of providing distinguishing information to distinguish between a plurality of possible outcomes for a condition code and directly calculating the condition code, based, at least in part, on this distinguishing information, as is recited in claim 9 presented herewith.

Claim 9 presented herewith further recites that the method includes "obtaining by a target machine a computer instruction from a source machine". Applicants respectfully submit that a careful reading of Liu fails to uncover any teaching or suggestion of a target machine obtaining a computer instruction from a source machine. Cited in the Office Action in this regard are Claims 1 and 3, and FIG. 3 of Liu wherein it discloses translating a flag generating CISC instruction. Applicants respectfully traverse this citation. Disclosure of a translation of instructions in one format to another format provides no teaching or suggestion per se of "obtaining by a target machine a computer instruction from a source machine" A careful reading of Liu fails to uncover any teaching or suggestion of target and source machines, let alone of obtaining an instruction by a target machine from a source machine, as is recited in claim 9 presented herewith.

For at least the above reasons, applicants respectfully submit that claim 9 presented herewith patentably distinguishes over the applied art of Liu.

Additionally, in accordance with independent claim 15 presented herewith, a computer system is provided for emulating computer instructions from a source machine to produce

LzLabs GmbH. Ex. 1004-123

sequences of instructions on a target machine. The computer system includes a memory and a processor in communication with the memory, and is configured to perform a method which includes one or more of the following aspects, one or more of which is not described, taught, or suggested by the combination of references: obtaining by a target machine a computer instruction from a source machine, the source machine having a different architecture from said target machine, and generating a sequence of target machine instructions which together operate to derive an encoding for a target machine condition code for the computer instruction, wherein the sequence of target machine instructions provides distinguishing information to distinguish between a plurality of possible outcomes for the condition code and directly calculates the condition code based, at least in part, on the distinguishing information without using branch instructions, at least part of said distinguishing information being separate from the condition code. Therefore, applicants respectfully request an indication of allowance for independent claim 15.

Further, in accordance with independent claim 21 presented herewith, a method is provided for emulating computer instructions from a source machine to produce sequences of instructions on a target machine. The method includes one or more of the following aspects, one or more of which is not described, taught, or suggested by the combination of references: obtaining by a target machine a computer instruction from a source machine, the source machine having a different architecture from said target machine, and generating a sequence of target machine instructions which together operate to derive an encoding for a target machine condition code for the computer instruction, wherein the sequence of target machine instructions provides distinguishing information to distinguish between a plurality of possible outcomes for the condition code and directly calculates the condition code based, at least in part, on the distinguishing information without using branch instructions, at least part of said distinguishing information being separate from the condition code. Therefore, applicants respectfully request an indication of allowance for independent claim 21.

LzLabs GmbH. Ex. 1004-124

The dependent claims are believed allowable for the same reasons as the independent claims, as well as for their own additional characterizations. For example, dependent claim 30 further recites:

- "wherein the plurality of possible outcomes comprise a first set of one or more possible outcomes and a second set of one or more possible outcomes, wherein the sequence of target machine instructions provides an intermediate result to distinguish between the first set of one or more possible outcomes and the second set of one or more possible outcomes, and wherein the distinguishing information distinguishes between possible outcomes within the first set of possible outcomes or within the second set of possible outcomes."

In accordance with applicants' recited protocol, the sequence of target machine instructions provides an intermediate result to distinguish between a first set of possible outcome(s) for the condition code and a second set of possible outcome(s) for the condition code, and the distinguishing information distinguishes between possible outcomes within the first set of outcome(s) or the second set of outcome(s). For instance, in the two examples provided in applicants' specification (see paragraphs [0025]-[0029], and [0030]), the intermediate result stored in register rC by instruction [4] distinguishes between the first and second sets of possible outcome(s), and the distinguishing information stored in register rX distinguishes the two outcomes within the first set of possible outcomes. No such teaching or suggestion is provided in Liu.

Furthermore, dependent claim 14 further characterizes dependent claim 30, by reciting, "wherein said first set of one or more possible outcomes comprises one case result and said second set of one or more possible outcomes comprises at least two other case results, and wherein said generating comprises: mimicking an instruction present in a source instruction stream in a manner which sets one or more target CPU flag bits and which stores a first result in a first storage location; executing an instruction which uses said stored first result to set the distinguishing information, the distinguishing information comprising at least one bit in a second storage location; using said stored first result to provide a second result in the first storage location; executing an instruction which employs said second result to produce a third result in the first storage location, the third result being the intermediate result which distinguishes between the one case result and the at least two other case results; and executing an instruction

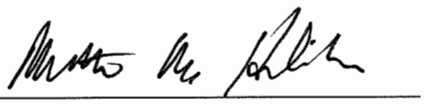
LzLabs GmbH. Ex. 1004-125

which uses the third result in said first storage location and the at least one bit in the second storage location to provide an indication in which the one case result and the at least two other case results are distinguished, wherein using the at least one bit in the second storage location distinguishes between possible outcomes within the at least two other case results, and wherein the one case result and the at least two other case results represent condition codes of said source machine." Attention is directed to paragraphs [0025]-[0029] and [0032] of the application for an example of the recited protocol. Applicants respectfully submit that no such teaching or suggestion is provided in Liu.

All claims are believed to be in condition for allowance, and such action is respectfully requested.

Should any issue remain unresolved, applicants' undersigned representative respectfully requests a telephone interview with the Examiner at the time that the Examiner is considering this Amendment in order to discuss the issue and facilitate prosecution of the subject application.

Respectfully submitted,



Matthew M. Hulihan, Esq.
Attorney for Applicants
Registration No.: 68,483

Dated: June 13, 2012.

HESLIN ROTHENBERG FARLEY & MESITI P.C.
5 Columbia Circle
Albany, New York 12203-5160
Telephone: (518) 452-5600
Facsimile: (518) 452-5579

LzLabs GmbH. Ex. 1004-126

Electronic Acknowledgement Receipt

EFS ID:	13002124
Application Number:	11668623
International Application Number:	
Confirmation Number:	4180
Title of Invention:	METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS
First Named Inventor/Applicant Name:	Reid T. Copeland
Customer Number:	46369
Filer:	Matthew M. Hulihan
Filer Authorized By:	
Attorney Docket Number:	POU920060152US1
Receipt Date:	13-JUN-2012
Filing Date:	30-JAN-2007
Time Stamp:	14:56:14
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1		POU920060152US1_PR03_20120613.pdf	782490 5e5a73fea8d8ab1d0cf34d087de429b60ac55616	yes	15

	Multipart Description/PDF files in .zip description		
	Document Description	**Start**	**End**
	Amendment/Req. Reconsideration-After Non-Final Reject	1	1
	Specification	2	2
	Claims	3	9
	Applicant Arguments/Remarks Made in an Amendment	10	15
Warnings:			
Information:			
	Total Files Size (in bytes):	782490	

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office
If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

PTO/SB/06 (07-06)
Approved for use through 1/31/2007. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

PATENT APPLICATION FEE DETERMINATION RECORD Substitute for Form PTO-875	Application or Docket Number 11/668,623	Filing Date 01/30/2007	☐ To be Mailed

APPLICATION AS FILED – PART I

	(Column 1)	(Column 2)	SMALL ENTITY ☐		OR	OTHER THAN SMALL ENTITY	
FOR	NUMBER FILED	NUMBER EXTRA	RATE ($)	FEE ($)		RATE ($)	FEE ($)
☐ BASIC FEE (37 CFR 1.16(a), (b), or (c))	N/A	N/A	N/A			N/A	
☐ SEARCH FEE (37 CFR 1.16(k), (i), or (m))	N/A	N/A	N/A			N/A	
☐ EXAMINATION FEE (37 CFR 1.16(o), (p), or (q))	N/A	N/A	N/A			N/A	
TOTAL CLAIMS (37 CFR 1.16(i))	minus 20 =	*	X $ =		OR	X $ =	
INDEPENDENT CLAIMS (37 CFR 1.16(h))	minus 3 =	*	X $ =			X $ =	
☐ APPLICATION SIZE FEE (37 CFR 1.16(s))	If the specification and drawings exceed 100 sheets of paper, the application size fee due is $250 ($125 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).						
☐ MULTIPLE DEPENDENT CLAIM PRESENT (37 CFR 1.16(j))							
* If the difference in column 1 is less than zero, enter "0" in column 2.			TOTAL			TOTAL	

APPLICATION AS AMENDED – PART II

AMENDMENT	06/13/2012	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA	SMALL ENTITY RATE ($)	ADDITIONAL FEE ($)	OR	OTHER THAN SMALL ENTITY RATE ($)	ADDITIONAL FEE ($)
	Total (37 CFR 1.16(i))	* 20	Minus	** 20	= 0	X $ =		OR	X $60=	0
	Independent (37 CFR 1.16(h))	* 3	Minus	***3	= 0	X $ =		OR	X $250=	0
	☐ Application Size Fee (37 CFR 1.16(s))									
	☐ FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							OR		
						TOTAL ADD'L FEE		OR	TOTAL ADD'L FEE	0

AMENDMENT		(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA	RATE ($)	ADDITIONAL FEE ($)	OR	RATE ($)	ADDITIONAL FEE ($)
	Total (37 CFR 1.16(i))	*	Minus	**	=	X $ =		OR	X $ =	
	Independent (37 CFR 1.16(h))	*	Minus	***	=	X $ =		OR	X $ =	
	☐ Application Size Fee (37 CFR 1.16(s))									
	☐ FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							OR		
						TOTAL ADD'L FEE		OR	TOTAL ADD'L FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.
** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20".
*** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3".
The "Highest Number Previously Paid For" (Total or Independent) is the highest number found in the appropriate box in column 1.

Legal Instrument Examiner:
/ANGELONA JONES/

This collection of information is required by 37 CFR 1.16. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

LzLabs GmbH. Ex. 1004-129

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
11/668,623	01/30/2007	Reid T. Copeland	POU920060152US1	4180

46369 7590 06/04/2012
HESLIN ROTHENBERG FARLEY & MESITI P.C.
5 COLUMBIA CIRCLE
ALBANY, NY 12203

EXAMINER	
HUISMAN, DAVID J	
ART UNIT	PAPER NUMBER
2183	
MAIL DATE	DELIVERY MODE
06/04/2012	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

LzLabs GmbH. Ex. 1004-130

Applicant-Initiated Interview Summary	Application No.	Applicant(s)	
	11/668,623	COPELAND ET AL.	
	Examiner	Art Unit	
	DAVID HUISMAN	2183	

All participants (applicant, applicant's representative, PTO personnel):

(1) *DAVID HUISMAN*.

(2) *Blanche E. Schiller, Reg. No. 35,670*.

(3) *Matthew M. Hulihan, Reg. No. 68,483*.

(4)_____.

Date of Interview: *31 May 2012*.

Type: ☒ Telephonic ☐ Video Conference
☐ Personal [copy given to: ☐ applicant ☐ applicant's representative]

Exhibit shown or demonstration conducted: ☐ Yes ☒ No.
If Yes, brief description: _____.

Issues Discussed ☐101 ☐112 ☒102 ☐103 ☐Others
(For each of the checked box(es) above, please describe below the issue and detailed description of the discussion)

Claim(s) discussed: *21, 26, and newly proposed claim 30*.

Identification of prior art discussed: *Liu*.

Substance of Interview
(For each issue discussed, provide a detailed description and indicate if agreement was reached. Some topics may include: identification or clarification of a reference or a portion thereof, claim interpretation, proposed amendments, arguments of any applied references etc...)

Applicant's representative (hereafter "applicant") and the examiner discussed proposed amendments to claims 21 and 26, and, in particular, the claimed distinguishing information. The examiner stated that the amendment to claim 21 appeared to overcome the current rejection, because the examiner's interpretation of Liu is that the condition codes are used to provide distinguishing information, and applicant has proposed that the condition code is calculated based on the distinguishing information (i.e., the reverse). The examiner did state, however, that the distinguishing information of claim 21 was still very broad and not defined, and that Liu may still read on the claim when applied in a different manner. No agreement was reached.

Applicant recordation instructions: The formal written reply to the last Office action must include the substance of the interview. (See MPEP section 713.04). If a reply to the last Office action has already been filed, applicant is given a non-extendable period of the longer of one month or thirty days from this interview date, or the mailing date of this interview summary form, whichever is later, to file a statement of the substance of the interview

Examiner recordation instructions: Examiners must summarize the substance of any interview of record. A complete and proper recordation of the substance of an interview should include the items listed in MPEP 713.04 for complete and proper recordation including the identification of the general thrust of each argument or issue discussed, a general indication of any other pertinent matters discussed regarding patentability and the general results or outcome of the interview, to include an indication as to whether or not agreement was reached on the issues raised.

☐ Attachment

/David J. Huisman/ Primary Examiner, Art Unit 2183	

LzLabs GmbH. Ex. 1004-131

Summary of Record of Interview Requirements

Manual of Patent Examining Procedure (MPEP), Section 713.04, Substance of Interview Must be Made of Record
A complete written statement as to the substance of any face-to-face, video conference, or telephone interview with regard to an application must be made of record in the application whether or not an agreement with the examiner was reached at the interview.

Title 37 Code of Federal Regulations (CFR) § 1.133 Interviews
Paragraph (b)

In every instance where reconsideration is requested in view of an interview with an examiner, a complete written statement of the reasons presented at the interview as warranting favorable action must be filed by the applicant. An interview does not remove the necessity for reply to Office action as specified in §§ 1.111, 1.135. (35 U.S.C. 132)

37 CFR §1.2 Business to be transacted in writing.
All business with the Patent or Trademark Office should be transacted in writing. The personal attendance of applicants or their attorneys or agents at the Patent and Trademark Office is unnecessary. The action of the Patent and Trademark Office will be based exclusively on the written record in the Office. No attention will be paid to any alleged oral promise, stipulation, or understanding in relation to which there is disagreement or doubt.

———

The action of the Patent and Trademark Office cannot be based exclusively on the written record in the Office if that record is itself incomplete through the failure to record the substance of interviews.

It is the responsibility of the applicant or the attorney or agent to make the substance of an interview of record in the application file, unless the examiner indicates he or she will do so. It is the examiner's responsibility to see that such a record is made and to correct material inaccuracies which bear directly on the question of patentability.

Examiners must complete an Interview Summary Form for each interview held where a matter of substance has been discussed during the interview by checking the appropriate boxes and filling in the blanks. Discussions regarding only procedural matters, directed solely to restriction requirements for which interview recordation is otherwise provided for in Section 812.01 of the Manual of Patent Examining Procedure, or pointing out typographical errors or unreadable script in Office actions or the like, are excluded from the interview recordation procedures below. Where the substance of an interview is completely recorded in an Examiners Amendment, no separate Interview Summary Record is required.

The Interview Summary Form shall be given an appropriate Paper No., placed in the right hand portion of the file, and listed on the "Contents" section of the file wrapper. In a personal interview, a duplicate of the Form is given to the applicant (or attorney or agent) at the conclusion of the interview. In the case of a telephone or video-conference interview, the copy is mailed to the applicant's correspondence address either with or prior to the next official communication. If additional correspondence from the examiner is not likely before an allowance or if other circumstances dictate, the Form should be mailed promptly after the interview rather than with the next official communication.

The Form provides for recordation of the following information:
- Application Number (Series Code and Serial Number)
- Name of applicant
- Name of examiner
- Date of interview
- Type of interview (telephonic, video-conference, or personal)
- Name of participant(s) (applicant, attorney or agent, examiner, other PTO personnel, etc.)
- An indication whether or not an exhibit was shown or a demonstration conducted
- An identification of the specific prior art discussed
- An indication whether an agreement was reached and if so, a description of the general nature of the agreement (may be by attachment of a copy of amendments or claims agreed as being allowable). Note: Agreement as to allowability is tentative and does not restrict further action by the examiner to the contrary.
- The signature of the examiner who conducted the interview (if Form is not an attachment to a signed Office action)

It is desirable that the examiner orally remind the applicant of his or her obligation to record the substance of the interview of each case. It should be noted, however, that the Interview Summary Form will not normally be considered a complete and proper recordation of the interview unless it includes, or is supplemented by the applicant or the examiner to include, all of the applicable items required below concerning the substance of the interview.

A complete and proper recordation of the substance of any interview should include at least the following applicable items:
1) A brief description of the nature of any exhibit shown or any demonstration conducted,
2) an identification of the claims discussed,
3) an identification of the specific prior art discussed,
4) an identification of the principal proposed amendments of a substantive nature discussed, unless these are already described on the Interview Summary Form completed by the Examiner,
5) a brief identification of the general thrust of the principal arguments presented to the examiner,
   (The identification of arguments need not be lengthy or elaborate. A verbatim or highly detailed description of the arguments is not required. The identification of the arguments is sufficient if the general nature or thrust of the principal arguments made to the examiner can be understood in the context of the application file. Of course, the applicant may desire to emphasize and fully describe those arguments which he or she feels were or might be persuasive to the examiner.)
6) a general indication of any other pertinent matters discussed, and
7) if appropriate, the general results or outcome of the interview unless already described in the Interview Summary Form completed by the examiner.

Examiners are expected to carefully review the applicant's record of the substance of an interview. If the record is not complete and accurate, the examiner will give the applicant an extendable one month time period to correct the record.

Examiner to Check for Accuracy

If the claims are allowable for other reasons of record, the examiner should send a letter setting forth the examiner's version of the statement attributed to him or her. If the record is complete and accurate, the examiner should place the indication, "Interview Record OK" on the paper recording the substance of the interview along with the date and the examiner's initials.

LzLabs GmbH. Ex. 1004-132

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
11/668,623	01/30/2007	Reid T. Copeland	POU920060152US1	4180

46369 7590 03/14/2012

HESLIN ROTHENBERG FARLEY & MESITI P.C.
5 COLUMBIA CIRCLE
ALBANY, NY 12203

EXAMINER	
HUISMAN, DAVID J	
ART UNIT	PAPER NUMBER
2183	

MAIL DATE	DELIVERY MODE
03/14/2012	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

LzLabs GmbH. Ex. 1004-133

Office Action Summary	Application No.	Applicant(s)	
	11/668,623	COPELAND ET AL.	
	Examiner	Art Unit	
	DAVID HUISMAN	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *21 December 2009*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ An election was made by the applicant in response to a restriction requirement set forth during the interview on _____; the restriction requirement and election have been incorporated into this action.

4)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

5)☒ Claim(s) *9-22 and 24-29* is/are pending in the application.

5a) Of the above claim(s) _____ is/are withdrawn from consideration.

6)☐ Claim(s) _____ is/are allowed.

7)☒ Claim(s) *9-22 and 24-29* is/are rejected.

8)☐ Claim(s) _____ is/are objected to.

9)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

10)☒ The specification is objected to by the Examiner.

11)☒ The drawing(s) filed on *16 June 2009* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

12)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

13)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

LzLabs GmbH. Ex. 1004-134

DETAILED ACTION

1. Claims 9-22 and 24-29 have been examined.

Specification

2. The disclosure is objected to because of the following informalities:

- In amended paragraph [0036], line 2, replace "accepts" with --accept--.

Appropriate correction is required.

Claim Objections

3. Claim 15 is objected to because of the following informalities: In line 5, replace "communications" with --communication--. Appropriate correction is required.

4. Claims 10, 16, and 22 are objected to because of the following informalities: In the last line of each claim, please insert a comma after "sign". Appropriate correction is required.

Claim Rejections - 35 USC § 112

5. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

6. Claims 14, 20, and 26 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Specifically, it is not clear what is meant by "using said result to provide to a shorter bit configuration". The scope of this is unclear. What does it mean for a result to provide a shorter bit configuration? Clarification is requested.

Claim Rejections - 35 USC § 101

7. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

8. Claims 9-14 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. Specifically, since the specification does not exclude a transitory medium (e.g., signals per se) as being a type of computer readable storage medium, or does not distinguish between storage media and transitory media, the broadest reasonable interpretation of "computer readable storage medium" includes interpreting it as a transitory medium, which is not statutory. Although applicant deleted the non-statutory language from the specification, deletion alone does not overcome the 101 rejection. Applicant must further explicitly disavow all embodiments encompassing non-statutory transitory media. An alternative fix to the 101 rejection would be to simply claim a "non-transitory computer readable storage medium".

Claim Rejections - 35 USC § 102

9. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

10. Claims 9-12, 14-18, 20-22, 24, and 26-28 are rejected under 35 U.S.C. 102(b) as being anticipated by Liu et al., U.S. Patent No. 6,049,864 (herein referred to as Liu).

11. Referring to claim 21, Liu has taught a method for emulating computer instructions from a source machine to produce sequences of instructions on a target machine, said method comprising:

a) obtaining by a target machine a computer instruction from a source machine, said source machine having a different architecture from said target machine. See Fig.3 and claims 1 and 3 of Liu, and note that instruction ADDF32 X1 = X2 + X3 is a CISC instruction to be executed on a RISC machine.

b) generating a sequence of target machine instructions which together operate to derive an encoding for a target machine condition code for the computer instruction. See Fig.3 and column 3, lines 4-6, and note that a sequence of addf32 and prodf32 instructions are generated to emulate the CISC instruction and also set at least some of a series of condition codes shown in Figs.1A, 1B, and 3.

c) wherein the sequence of target machine instructions directly calculates the condition code and provides distinguishing information to distinguish between a plurality of possible outcomes for the condition code without using branch instructions, at least part of said distinguishing information being separate from the condition code. See Fig.3, and note that the emulation of the ADDF32 source instruction results in the execution of multiple non-branch instructions to set carry, overflow, and sign codes. Also, distinguishing information may be produced. See column 4, lines 53-63, and note that distinguishing information representing a combination of flags may be stored to make subsequent branching more efficient. This stored information can be representative of a value being less than, less than or equal, or complements of those (greater than or greater than or equal).

12. Referring to claim 22, Liu has taught the method of claim 21 wherein said calculation of the target machine condition code comprises using at least one of carry, sign and overflow codes in the calculation. See Fig.3 and note, in the bottom left box, that carry codes (e.g. X1{32}, X1{33}, ...) are used in calculation the condition code shown in the bottom right box of Fig.3.

13. Referring to claim 24, Liu has taught the method of claim 21 wherein said method further includes executing said generated sequence of target machine instructions. See Fig.3.

14. Referring to claim 26, Liu has taught the method of claim 21 in which said generating comprises the steps of:

a) mimicking an instruction present in a source instruction stream in a manner which sets one or more target CPU flag bits and which places a result in a first storage location. See Fig.3 and note that a source CISC instruction is mimicked with at least two RISC instructions that set target flag bits and which also place a result in a first storage location. For purposes of examination, the first storage location is interpreted as bits X1{37:0}. The result of the addition is placed in these bits.

b) executing an instruction which uses said result to set at least one bit in second storage location so as to distinguish one or more case results. See Fig.3 and note that in response to executing the addf32 instruction, the result of the addition is used to set bit X1{38}, which is herein interpreted as the second storage location. This bit distinguishes between carry and no-carry case results.

c) using said result to provide a shorter bit configuration which serves to distinguish at least one of said case results. See Fig.3. Though the actual addition result is 32 bits (in X1{31:0}), bit X1{31} is used to indicate positive or negative, and consequently, the result provides a shorter configuration of 31 bits (X1{30:0}).

d) executing an instruction which employs said first storage location to produce a result which distinguishes a different set of case results. See Fig.3 and note that bits X1{31:30} are used is setting codes which distinguish a different set of case results.

e) executing an instruction which uses the results in said first and second storage locations to provide an indication in which at least three cases are distinguished, said cases representing condition codes of said source machine. See Fig.3, and note that prodf32 uses the results in the first storage location (specifically bit X1{37}) and the second storage location (specifically bit X1{38}) to provide carry and overflow indications that distinguish between results of the type: (1) no carry, no overflow, (2) no carry, overflow, (3) carry, no overflow, and (4) carry, overflow.

15. Referring to claims 9-10, 12, and 14, claims 9-10, 12, and 14 are rejected for the same reasons set forth in the rejections of claims 21-22, 24, and 26, respectively.

16. Referring to claim 11, Liu has taught the computer program product of claim 10, wherein said calculation employs temporary locations for storing intermediate results. See Fig.3, and note that in response to the addf32 instruction, not only is the result produced and stored in X1{31:0}, but other intermediate values are stored in X1{32:38}. These intermediate results are then used by the prodf32 instruction to set flags.

17. Referring to claims 15-16, 18, and 20, the systems of claims 15-16, 18, and 20 perform the methods of claims 21-22, 24 and 26, respectively. Consequently, claims 15-16, 18, and 20 are rejected for the same reasons set forth in the rejections of claims 21-22, 24, and 26, respectively. Also, it should be noted that Liu has taught a memory, both inherently and explicitly. That is, in Fig.3, Liu shows a memory X1[38:0] for storing data. Also, program

instructions are inherently stored on some medium so that the processor may fetch and process them.

18. Referring to claim 17, Liu has taught the computer system of claim 16. Furthermore, claim 17 is rejected for the same reasons set forth in the rejection of claim 11.

19. Referring to claim 27, Liu has taught the computer program product of claim 9, wherein the sequence of target machine instructions together operate to directly calculate the target machine condition code using at least one of addition, subtraction and multiplication. See Fig.3, and note the addition operation. Also, one of ordinary skill in the art would recognize that a subtraction and multiplication would be handled in a similar fashion.

20. Referring to claim 28, Liu has taught the computer program product of claim 9, wherein the providing distinguishing information comprises setting one or more bits in one or more registers. See column 4, lines 53-63.

Claim Rejections - 35 USC § 103

21. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

22. Claims 13, 19, 25, and 29 are rejected under 35 U.S.C. 103(a) as being unpatentable over Liu in view of the examiner's taking of Official Notice.

23. Referring to claim 25, Liu has taught the method of claim 24. Liu has not explicitly taught that said executing occurs at substantially the same time as said generating. However, the

examiner asserts that interpretive environments, in which instructions are translated at run-time, are well known and accepted in the art. Such systems reduce the amount of work done by the compiler and therefore, reduce compilation time. As a result, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Liu such that the generating of the sequence of instructions occurs at run-time when the source instruction is encountered, thereby resulting in execution occurring at substantially the same time as said generating.

24. Referring to claim 13, claim 13 is rejected for the same reasons set forth in the rejection of claim 25.

25. Referring to claim 19, the system of claim 19 performs the method of claim 25. Consequently, claim 19 is rejected for the same reasons set forth in the rejection of claim 25.

26. Referring to claim 29, Liu has taught the computer program product of claim 9, wherein the distinguishing information distinguishes between outcomes for the following cases: the first operand is greater than the second operand; and the first operand is less than the second operand. See column 4, lines 53-63. Note that a “less" bit may be stored. If the bit is set, then the first operand is less than a second operand. If the bit is not set, then the first operand is greater than the second operand. Or, alternatively, a separate greater bit may be stored, as Liu has stated that the complements of the listed cases may be implements. Liu has further taught information indicating that a first operand and a second operand of a target machine instruction of the sequence of target machine instructions are equal (see the zero flag of Fig.3), but has not explicitly taught that this information is produced without executing a branch instruction. However, it should be realized that one of ordinary skill in the art would have recognized that the zero flag could be set in Liu without performing a branch instruction. For instance, one could

merely OR all of the X1{31:0} bits together and invert the result and assign it to the zero flag. There are number of alternatives to performing branching. As a result, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Liu such that distinguishing information which indicates that a first operand and a second operand of a target machine instruction of the sequence of target machine instructions are equal is produced without executing a branch instruction. Instead, the information would be produced by hardware logic (e.g., OR gate), which as is known, is faster than a group of software instructions, including a branch, to perform the same function.

Examiner Comments

27. In claims 13, 19, and 25, applicant claims "wherein said executing occurs at substantially the same time as said generating". The question of whether "substantially" renders the claim indefinite is a valid one. However, paragraphs [0035]-[0036] of the specification state that execution of a sequence occurs immediately upon the sequence being generated, as one might find in an interpretive environment, or execution may occur at a later time, like in a compilation-like environment. Hence, in order for claims 13, 19, and 25 to be definite, claims 13, 19, and 25 are hereby limited such that generation of a sequence and execution of that sequence occur no further apart in time than it would take an interpreter to generate the sequence at run-time, which is less than the amount of time between generating the sequence at compile-time using a compiler, and then executing that sequence.

Response to Arguments

28. Applicant's arguments filed on December 21, 2009, have been fully considered, but are non-persuasive for the reasons set forth in the rejections above. Specifically, the examiner asserts that Liu has taught that which applicant argues Liu has not taught (i.e., the distinguishing information).

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to DAVID J. HUISMAN whose telephone number is (571)272-4168. The examiner can normally be reached on Monday-Friday (8:00-4:30).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/David J. Huisman/
Primary Examiner, Art Unit 2183

EAST Search History

EAST Search History (Prior Art)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
S42	1	"20030093649".pn.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S43	86	emulat$3 with "condition code"	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S44	102	emulat$3 with ("condition code" "flag bit" "status bit") and (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S45	29	emulat$3 with ("condition code" "flag bit" "status bit") same branch and (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S46	68	("20020026633" | "4514803" | "4812975" | "5313614" | "5329471" | "5408637" | "5440702" | "5551015" | "5574873" | "5574927" | "5577233" | "5598560" | "5606696" | "5649203" | "5724590" | "5737631" | "5751982" | "5768593" | "5805792" | "5805895" | "5870575" | "5890000" | "5901308" | "5903760" | "5925124" | "5930509" | "5946491" | "5956495" | "5966539" | "5983335" | "6000028" |	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-145

		"6009261" | "6031994" | "6076155" | "6077314" | "6091897" | "6105124" | "6126329" | "6170083" | "6185675" | "6199095" | "6205545" | "6223339" | "6301705" | "6308318" | "6321379" | "6330691" | "6360194" | "6367067" | "6427234" | "6434741" | "6463582" | "6477641" | "6502237" | "6505296" | "6522985" | "6523173" | "6526572" | "6530078" | "6530079" | "6535903" | "6539471" | "6539541" | "6549987" | "6564297" | "6594824" | "6631514" | "6643630").PN.				
S47	6	(emulat$3 translat$3) same ("condition code" "flag bit" "status bit") same (without avoid $4) near2 branch$3 and (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S48	144	(emulat$3 translat$3) with (without avoid$4) near2 branch$3 and (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S49	12	(emulat$3 translat$3) with (without avoid$4) near2 branch$3 and (overflow carry sign) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-146

S50	5	("4951195" | "5241678" | "5301325" | "5313614" | "5598560").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S51	28	("5903760").URPN.	USPAT	OR	ON	2009/03/16 16:40
S52	68	("20020026633" | "4514803" | "4812975" | "5313614" | "5329471" | "5408637" | "5440702" | "5551015" | "5574873" | "5574927" | "5577233" | "5598560" | "5606696" | "5649203" | "5724590" | "5737631" | "5751982" | "5768593" | "5805792" | "5805895" | "5870575" | "5890000" | "5901308" | "5903760" | "5925124" | "5930509" | "5946491" | "5956495" | "5966539" | "5983335" | "6000028" | "6009261" | "6031994" | "6076155" | "6077314" | "6091897" | "6105124" | "6126329" | "6170083" | "6185675" | "6199095" | "6205545" | "6223339" | "6301705" | "6308318" | "6321379" |	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-147

		"6330691" | "6360194" | "6367067" | "6427234" | "6434741" | "6463582" | "6477641" | "6502237" | "6505296" | "6522985" | "6523173" | "6526572" | "6530078" | "6530079" | "6535903" | "6539471" | "6539541" | "6549987" | "6564297" | "6594824" | "6631514" | "6643630").PN.				
S53	68	("20020026633" | "4514803" | "4812975" | "5313614" | "5329471" | "5408637" | "5440702" | "5551015" | "5574873" | "5574927" | "5577233" | "5598560" | "5606696" | "5649203" | "5724590" | "5737631" | "5751982" | "5768593" | "5805792" | "5805895" | "5870575" | "5890000" | "5901308" | "5903760" | "5925124" | "5930509" | "5946491" | "5956495" | "5966539" | "5983335" | "6000028" | "6009261" |	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-148

		"6031994" | "6076155" | "6077314" | "6091897" | "6105124" | "6126329" | "6170083" | "6185675" | "6199095" | "6205545" | "6223339" | "6301705" | "6308318" | "6321379" | "6330691" | "6360194" | "6367067" | "6427234" | "6434741" | "6463582" | "6477641" | "6502237" | "6505296" | "6522985" | "6523173" | "6526572" | "6530078" | "6530079" | "6535903" | "6539471" | "6539541" | "6549987" | "6564297" | "6594824" | "6631514" | "6643630").PN.				
S54	723	(emulat$3 translat$3) with compar$4 with instruction	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S55	21	(emulat$3 translat$3) with compar$4 with instruction same (code flag) near2 (condition status bit)	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S56	412	712/209.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S57	487	712/221.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S58	867	712/227.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S59	897	717/136.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-149

S60	265	717/137.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S61	234	717/138.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S62	1	emulat$3 with ("condition code" "flag bit" "status bit") and (overflow carry sign)	USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2009/03/16 16:40
S63	1362	717/124.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S64	140	717/134.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S65	1560	(emulat$3 translat$3) same (arithmetic add $5 subtract$3) same (code flag) near2 (condition status bit)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S66	101	(emulat$3 translat$3) same (arithmetic add $5 subtract$3) with set $4 with (code flag) near2 (condition status bit)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S67	37	(emulat$3 translat$3) same (arithmetic add $5 subtract$3) with set $4 with (code flag) near2 (condition status bit) same (Z O S C sign zero overflow carry)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S68	29	(emulat$3 translat$3) same (arithmetic add $5 subtract$3) with set $4 with (code flag) near2 (condition status bit) same (Z O S C sign zero overflow carry) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S69	7	("20020046298" "20040176941" "20050015754" "20070011664" "5301302" "5678032" "5966515").pn.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S70	4	S69 and (code flag) near2 (condition status bit)	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-150

S71	119	simulat$3 near4 (code flag) near2 (condition status bit)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S72	67	simulat$3 near4 (code flag) near2 (condition status bit) and g06f$. ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S73	35	simulat$3 near4 (code flag) near2 (condition status bit) and (overflow carry) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S74	127	(emulat$3 translat$3) and (host target) with set$4 with (code flag) near2 (condition status bit) and ((overflow O) and (carry C) and (sign S negative N)) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S75	59	(emulat$3 translat$3) and (host target) with set$4 with (code flag) near2 (condition status bit) and ((overflow O) and (carry C) and (sign S negative N)) and ("712"/$.ccls. "717"/$.ccls.)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S76	32	emulat$3 and (host target) with set$4 with (code flag) near2 (condition status bit) and ((overflow O) and (carry C) and (sign S negative N)) and ("712"/$.ccls. "717"/$. ccls.)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S77	1	emulat$3 and (host target) with set$4 with (code flag) near2 (condition status bit) same ((overflow O) and (carry C) and (sign S negative N)) and ("712"/$.ccls. "717"/$.ccls.)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-151

S78	1	emulat$3 and (host target) with set$4 with (code flag) near2 (condition status bit) same (((overflow O) and (carry C) and (sign S negative N))) and ("712"/$.ccls. "717"/$.ccls.)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S79	6	("4415969" | "5327537" | "5440749" | "5493687" | "5590359" | "5689720").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S80	2	("5774694").URPN.	USPAT	OR	ON	2009/03/16 16:40
S81	8	("5313614" | "5481693" | "5619664" | "5774694" | "5859999" | "5860017" | "5884059" | "6049864").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S82	1	"20060173931".pn.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:44
S83	446	703/26.ccls.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:46
S84	20	emulat$3 with ("condition code" "flag bit" "status bit") same (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/07/14 17:52
S85	0	emulat$3 with ("condition code" "flag bit" "status bit") same (overflow carry sign)	USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2009/07/14 17:53
S86	47	huisman.xp.	US-PGPUB; USPAT	OR	ON	2010/03/04 09:27
S87	1913	subtract$3 with carry with (bit flag)	US-PGPUB; USPAT	OR	ON	2010/03/04 11:03
S88	186	subtract$3 with set$4 with carry adj (bit flag)	US-PGPUB; USPAT	OR	ON	2010/03/04 11:03
S89	112	subtract$3 with set$4 with carry adj (bit flag) and "712"/$.ccls.	US-PGPUB; USPAT	OR	ON	2010/03/04 11:03
S90	2	("5574931" "5999976").pn.	US-PGPUB; USPAT	OR	ON	2010/03/04 15:04

LzLabs GmbH. Ex. 1004-152

EAST Search History (Interference)

<This search history is empty>

3/5/10 11:50:08 AM
C:\Documents and Settings\DHuisman\My Documents\EAST\Workspaces\11-668623.wsp

LzLabs GmbH. Ex. 1004-153

Search Notes	Application/Control No.	Applicant(s)/Patent Under Reexamination
	11668623	COPELAND ET AL.
	Examiner	Art Unit
	DAVID J HUISMAN	2183

SEARCHED			
Class	Subclass	Date	Examiner
712	209, 221, 227	3-16-09	DJH
717	124, 134, 136, 137, 138	3-16-09	DJH
703	26	3-16-09	DJH

SEARCH NOTES		
Search Notes	Date	Examiner
EAST Search (all databases)	3/16/09	DJH
Inventor Name Search (PALM)	3/16/09	DJH
Updated EAST Search (all databases)	7-14-09	DJH
Updated EAST Search (all databases)	3-5-10	DJH

INTERFERENCE SEARCH			
Class	Subclass	Date	Examiner

LzLabs GmbH. Ex. 1004-154

Doc code: RCEX
Doc description: Request for Continued Examination (RCE)

PTO/SB/30EFS (06-09)
Approved for use through 06/30/2009. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

REQUEST FOR CONTINUED EXAMINATION(RCE)TRANSMITTAL
(Submitted Only via EFS-Web)

Application Number	11668623	Filing Date	2007-01-30	Docket Number (if applicable)	POU920060152US1	Art Unit	2183
First Named Inventor	Reid T. Copeland			Examiner Name	David J. Huisman		

This is a Request for Continued Examination (RCE) under 37 CFR 1.114 of the above-identified application.
Request for Continued Examination (RCE) practice under 37 CFR 1.114 does not apply to any utility or plant application filed prior to June 8, 1995, or to any design application. The Instruction Sheet for this form is located at WWW.USPTO.GOV

SUBMISSION REQUIRED UNDER 37 CFR 1.114

Note: If the RCE is proper, any previously filed unentered amendments and amendments enclosed with the RCE will be entered in the order in which they were filed unless applicant instructs otherwise. If applicant does not wish to have any previously filed unentered amendment(s) entered, applicant must request non-entry of such amendment(s).

☐ Previously submitted. If a final Office action is outstanding, any amendments filed after the final Office action may be considered as a submission even if this box is not checked.

- ☐ Consider the arguments in the Appeal Brief or Reply Brief previously filed on _____
- ☐ Other _____

☒ Enclosed

- ☒ Amendment/Reply
- ☐ Information Disclosure Statement (IDS)
- ☐ Affidavit(s)/ Declaration(s)
- ☐ Other _____

MISCELLANEOUS

☐ Suspension of action on the above-identified application is requested under 37 CFR 1.103(c) for a period of months _____
(Period of suspension shall not exceed 3 months; Fee under 37 CFR 1.17(i) required)

☐ Other _____

FEES

The RCE fee under 37 CFR 1.17(e) is required by 37 CFR 1.114 when the RCE is filed.

☒ The Director is hereby authorized to charge any underpayment of fees, or credit any overpayments, to Deposit Account No 081935

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT REQUIRED

☒ Patent Practitioner Signature

☐ Applicant Signature

LzLabs GmbH. Ex. 1004-155

PTO/SB/30EFS (06-09)
Approved for use through 06/30/2009. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Signature of Registered U.S. Patent Practitioner			
Signature	/Blanche E. Schiller/	Date (YYYY-MM-DD)	2009-12-21
Name	Blanche E. Schiller, Esq.	Registration Number	35670

This collection of information is required by 37 CFR 1.114. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450.

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

LzLabs GmbH. Ex. 1004-156

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these record s.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

LzLabs GmbH. Ex. 1004-157

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants:	Copeland et al.	Confirmation No.: 4180
Serial No.:	11/668,623	Group Art Unit: 2183
Filed:	01/30/2007	Examiner: David J. Huisman
Title:	EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS WITHOUT EXECUTING BRANCH INSTRUCTIONS	

CERTIFICATE OF ELECTRONIC TRANSMISSION

I hereby certify that this correspondence is being transmitted electronically to: United States Patent and Trademark Office, Alexandria, VA 22313-1450, on December 21, 2009.

Jill K. Becker

Date of Signature: December 21, 2009

To: Mail Stop RCE
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Amendment Accompanying Filing of Request for Continued Examination (RCE)

Dear Sir:

This RCE and Amendment are being submitted in reply to the Office Action mailed August 20, 2009 for the above-referenced patent application. A response to the Final Office Action was initially due on or before November 20, 2009, and thus, a Request For Extension Of Time and the requisite fee are enclosed herewith. Therefore, this Response is being timely filed on December 21, 2009, as December 20, 2009 was a Sunday.

Amendments to the Specification begin on page 2 of this paper.

Amendments to the Claims begin on page 4 of this paper.

Remarks begin on page 10 of this paper.

LzLabs GmbH. Ex. 1004-158

Amendments to the Specification

Please amend the above-identified application, as follows:

[0032] In general, this process has the following steps, none of which includes the use or execution of any branch instructions. First, an instruction (step 100 in Figure 2) in the target machine's architecture is executed which mimics the instruction present in the source instruction stream in a manner which sets one or more target CPU flag bits and which places a result in a storage location (first location, such as rC above) accessible to the emulator. Next, an instruction (step 105) is executed which uses that result to set a bit or bits in another emulator controlled storage location (second location, such as rX above) to distinguish one or more case results. Next, the aforementioned result is used to ~~reset itself (step 110) to~~provide a shorter bit configuration (one bit in the above example) which also serves to distinguish one or more case results (step 110). Next, an instruction is executed (step 115) which employs the first storage location to produce a result which distinguishes a different set of case results. Lastly, a target machine instruction is executed (step 120) which uses the results in the first and second locations to provide an indication in one of the two emulator accessible instructions in which at least three cases are distinguished.

[0035] In any event the environment in which the present invention operates is shown in Figure 3. The present invention operates in a data processing environment which effectively includes one or more of the computer elements shown in Figure 3. In particular, computer 500 includes central processing unit (CPU) 520 which accesses programs and data stored within random access memory 510. Memory 510 is typically volatile in nature and accordingly such systems are provided with nonvolatile memory typically in the form of rotatable magnetic memory 540. While memory 540 is preferably a nonvolatile magnetic device, other media may be employed. CPU ~~530~~ 520 communicates with users at consoles such as terminal 550 through Input/Output unit 530. Terminal 550 is typically one of many, if not thousands, of consoles in communication with computer 500 through one or more I/O unit 530. In particular, console unit 550 is shown as having included therein a device for reading medium of one or more types such as CD-ROM 600 shown in Figure ~~5~~4. Media 600, an example of which is shown in Figure 4, comprises any convenient device including,

LzLabs GmbH. Ex. 1004-159

but not limited to, magnetic media, optical storage devices and chips such as flash memory devices or so-called thumb drives. ~~Disk 600 also represents a more generic distribution medium in the form of electrical signals used to transmit data bits which represent codes for the instructions discussed herein. While such transmitted signals may be ephemeral in nature they still, nonetheless constitute a physical medium carrying the coded instruction bits and are intended for permanent capture at the signal's destination or destinations.~~

[0036] The typical emulation environment in which the present invention is employed is illustrated in Figure 5. Emulators such as 320 ~~except~~ accepts, as input, instruction streams ~~305,~~ 305 representing machine or assembly language instructions which are designed to operate on source machine 300. Emulator 320 employees memory 315 in target machine 310 to produce a stream of instructions which are capable of executing on target machine 310. While Figure 5 particularly shows operation within an emulation environment, it is also noted that the present invention contemplates a situation in which emulator 320 operates essentially as an interpreter in which the instructions are not only translated to the new architecture but in which they are also executed at essentially the same time.

LzLabs GmbH. Ex. 1004-160

Amendments to the Claims

Kindly cancel claim 23, without prejudice, and amend claims 9-12, 14-18, 20-22, 24 and 26, and add new claims 27-29, as set forth below. The changes in the amended claims are shown by strikethrough for deleted matter and underlining for added matter.

1-8. (Canceled)

9. (Currently Amended) A computer program product for emulating computer instructions from a source machine to produce sequences of instructions on a target machine, said computer program product comprising:

a computer readable storage medium readable by a processor and storing instructions for execution by the processor for performing a method comprising:

~~generating a sequence of target machine instructions which together operate to directly calculate target machine condition codes from carry, sign and overflow codes without including branch instructions from the target machine~~obtaining by a target machine a computer instruction from a source machine, said source machine having a different architecture from said target machine; and

generating a sequence of target machine instructions which together operate to derive an encoding for a target machine condition code for the computer instruction, wherein the sequence of target machine instructions directly calculates the condition code and provides distinguishing information to distinguish between a plurality of possible outcomes for the condition code without using branch instructions, at least part of said distinguishing information being separate from the condition code.

10. (Currently Amended) The computer program product of claim 9, wherein said calculation of the target machine condition ~~codes is carried out in a sequence of non-branching instructions that manipulate condition code settings produced within the target~~

LzLabs GmbH. Ex. 1004-161

~~machine itself to produce the same condition code in a target machine location~~code comprises using at least one of carry, sign and overflow codes in the calculation.

11. (Currently Amended) The computer program product of claim 10, wherein said ~~manipulation~~ calculation employs temporary locations for storing intermediate results.

12. (Currently Amended) The computer program product of claim 9, wherein said method further ~~including~~includes executing said generated sequence of target machine instructions.

13. (Previously Presented) The computer program product of claim 12, wherein said executing occurs at substantially the same time as said generating.

14. (Currently Amended) The computer program product of claim 9, wherein said generating comprises:

mimicking an instruction present in a source instruction stream in a manner which sets one or more target CPU flag bits and which places a result in a first storage location;

executing an instruction which uses said result to set at least one bit in a second storage location so as to distinguish one or more case results;

using said result to ~~reset itself to~~provide a shorter bit configuration which serves to distinguish at least one of said case results;

executing an instruction which employs said first storage location to produce a result which distinguishes a different set of case results; and

executing an instruction which uses the results in said first and second storage locations to provide an indication in which at least three cases are distinguished, said cases representing condition codes of said source machine.

LzLabs GmbH. Ex. 1004-162

15. (Currently Amended) A computer system for emulating computer instructions from a source machine to produce sequences of instructions on a target machine, the computer system comprising:

a memory; and

a processor in communications with the memory, wherein the computer system is capable of performing a method, the method comprising:

~~generating a sequence of target machine instructions which together operate to directly calculate target machine condition codes from carry, sign and overflow codes without including branch instructions from the target machine~~obtaining by a target machine a computer instruction from a source machine, said source machine having a different architecture from said target machine; and

generating a sequence of target machine instructions which together operate to derive an encoding for a target machine condition code for the computer instruction, wherein the sequence of target machine instructions directly calculates the condition code and provides distinguishing information to distinguish between a plurality of possible outcomes for the condition code without using branch instructions, at least part of said distinguishing information being separate from the condition code.

16. (Currently Amended) The computer system of claim 15, wherein said calculation of the target machine condition ~~codes is carried out in a sequence of non-branching instructions that manipulate condition code settings produced within the target machine itself to produce the same condition code in a target machine location~~code comprises using at least one of carry, sign and overflow codes in the calculation.

17. (Currently Amended) The computer system of claim 16, wherein said ~~manipulation~~ calculation employs temporary locations for storing intermediate results.

LzLabs GmbH. Ex. 1004-163

18. (Currently Amended) The computer system of claim 15, wherein said method further ~~including~~ includes executing said generated sequence of target machine instructions.

19. (Previously Presented) The computer system of claim 18, wherein said executing occurs at substantially the same time as said generating.

20. (Currently Amended) The computer system of claim 15, wherein said generating comprises:

mimicking an instruction present in a source instruction stream in a manner which sets one or more target CPU flag bits and which places a result in a first storage location;

executing an instruction which uses said result to set at least one bit in a second storage location so as to distinguish one or more case results;

using said result to ~~reset itself to~~provide a shorter bit configuration which serves to distinguish at least one of said case results;

executing an instruction which employs said first storage location to produce a result which distinguishes a different set of case results; and

executing an instruction which uses the results in said first and second storage locations to provide an indication in which at least three cases are distinguished, said cases representing condition codes of said source machine.

21. (Currently Amended) A method for emulating computer instructions from a source machine to produce sequences of instructions on a target machine, said method comprising:

~~generating a sequence of target machine instructions which together operate to directly calculate target machine condition codes from carry, sign and overflow codes without including branch instructions from the target machine~~obtaining by a target machine a computer instruction from a source machine, said source machine having a different architecture from said target machine; and

LzLabs GmbH. Ex. 1004-164

generating a sequence of target machine instructions which together operate to derive an encoding for a target machine condition code for the computer instruction, wherein the sequence of target machine instructions directly calculates the condition code and provides distinguishing information to distinguish between a plurality of possible outcomes for the condition code without using branch instructions, at least part of said distinguishing information being separate from the condition code.

22. (Currently Amended) The method of claim 21, wherein said calculation of the target machine condition ~~codes is carried out in a sequence of non-branching instructions that manipulate condition code settings produced within the target machine itself to produce the same condition code in a target machine location~~ code comprises using at least one of carry, sign and overflow codes in the calculation.

23. (Canceled)

24. (Currently Amended) The method of claim 21, wherein said method further ~~including~~ includes executing said generated sequence of target machine instructions.

25. (Previously Presented) The method of claim 24, wherein said executing occurs at substantially the same time as said generating.

26. (Currently Amended) The method of claim 21, wherein said generating comprises:

mimicking an instruction present in a source instruction stream in a manner which sets one or more target CPU flag bits and which places a result in a first storage location;

executing an instruction which uses said result to set at least one bit in a second storage location so as to distinguish one or more case results;

using said result to ~~reset the result to~~provide a shorter bit configuration which serves to distinguish at least one of said case results;

LzLabs GmbH. Ex. 1004-165

executing an instruction which employs said first storage location to produce a result which distinguishes a different set of case results; and

executing an instruction which uses the results in said first and second storage locations to provide an indication in which at least three cases are distinguished, said cases representing condition codes of said source machine.

27. (New) The computer program product of claim 9, wherein the sequence of target machine instructions together operate to directly calculate the target machine condition code using at least one of addition, subtraction and multiplication.

28. (New) The computer program product of claim 9, wherein the providing distinguishing information comprises setting one or more bits in one or more registers.

29. (New) The computer program product of claim 9, wherein the distinguishing information distinguishes between outcomes for the following cases: a first operand and a second operand of a target machine instruction of the sequence of target machine instructions are equal; the first operand is greater than the second operand; and the first operand is less than the second operand.

LzLabs GmbH. Ex. 1004-166

Remarks

Entry of the amendments, reconsideration of the application, as amended, and allowance of all pending claims are respectfully requested. Upon entry of the amendments, claims 9-22 and 24-29 are pending.

With the above amendments, applicants are clarifying one or more aspects of the present invention. These amendments are being made in a bona fide attempt to further prosecution of this application and not in acquiescence of the rejections. Support for the amendments may be found throughout applicants' specification, including, for instance, paragraphs 12, 21, 27 and 29. Therefore, no new matter has been added.

In the Final Office Action, dated August 20, 2009, claims 14, 20 and 26 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Without acquiescing to this rejection, applicants have amended those claims to remove the objected to language. Therefore, applicants respectfully request withdrawal of this rejection.

Additionally, claims 9-14 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. Without acquiescing to this rejection, applicants have amended independent claim 9 to specifically indicate that the storage medium is a computer readable storage medium, and has removed the objected to language from the specification. Thus, applicants respectfully request withdrawal of the §101 rejection.

Yet further, claims 9-12, 14-18, 20-24 and 26 are rejected under 35 U.S.C. 102(b) as being anticipated by Liu et al. (U.S. Patent No. 6,049,864); and claims 13, 19 and 25 are rejected under 35 U.S.C. 103(a) as being unpatentable over Liu et al. in view of the Examiner's taking of Official Notice. Applicants respectfully, but most strenuously, traverse these rejections to any extent deemed applicable to the amended claims.

Applicants' invention, in one aspect, is directed to the emulation of source machine instructions on a target machine, in which the source machine and the target machine have different architectures. The emulation includes encoding condition codes for the instructions.

LzLabs GmbH. Ex. 1004-167

For instance, a sequence of target machine instructions is generated, which together operate to directly calculate target machine condition codes without the use of branch instructions. As one example, the instruction sequences set a bit or bits in a result or temporary register to distinguish each possible outcome of a condition code setting. Further, the instruction sequences typically manipulate the carry, sign and/or overflow bits.

In particular, in one aspect, applicants claim a computer program product emulating computer instructions from a source machine to produce sequences of instructions on a target machine. The computer program product includes a computer readable storage medium readable by a processor and storing instructions for execution by the processor for performing a method. The method includes, for instance, obtaining by a target machine a computer instruction from a source machine, the source machine having a different architecture from the target machine; and generating a sequence of target machine instructions which together operate to derive an encoding for a target machine condition code for the computer instruction, wherein the sequence of target machine instructions directly calculates the condition code and provides distinguishing information to distinguish between a plurality of possible outcomes for the condition code without using branch instructions, at least part of the distinguishing information being separate from the condition code.

Thus, in this aspect of applicants' claimed invention, a sequence of target machine instructions is generated which together operate to derive an encoding for a target machine condition code, wherein the sequence of target machine instructions directly calculates the condition code and provides distinguishing information to distinguish a plurality of possible outcomes for the condition code. At least this aspect of applicants' claimed invention is not described, taught or suggested in Liu.

Liu describes a technique for scheduling a flag generating instruction and a subsequent instruction. In Liu, flags are set by using intermediate flag data produced during execution of an instruction. While Liu describes setting flags, Liu fails to describe, teach or suggest one or more aspects of applicants' claimed invention. As one example, Liu fails to describe, teach or suggest generating a sequence of target machine instructions which together operate to derive an encoding for a condition code and provide distinguishing

LzLabs GmbH. Ex. 1004-168

information to distinguish a plurality of possible outcomes for the condition code. Further, Liu fails to describe, teach or suggest that at least part of the distinguishing information is separate from the condition code.

Applicants respectfully submit that there is no description, teaching or suggestion in Liu of, for instance, providing distinguishing information that distinguishes between a plurality of possible outcomes for the condition code. Liu sets a bit, but does not provide the distinguishing information, as claimed by applicants. Further, there is no description, teaching or suggestion of providing this distinguishing information, in which at least part of the distinguishing information is separate from the condition code. Since at least these aspects are missing from Liu, applicants respectfully submit that Liu fails to anticipate or render obvious applicants' claimed invention. Therefore, applicants respectfully request an indication of allowability for independent claim 9.

Further, applicants respectfully submit that independent claim 15 is also patentable over Liu. As described above, Liu fails to describe, teach or suggest at least applicants' claimed element of generating a sequence of target machine instructions that directly calculates the condition code and provides distinguishing information to distinguish between a plurality of possible outcomes for the condition code, wherein at least part of the distinguishing information is separate from the condition code. Since this is not described, taught or suggested in Liu, Liu does not anticipate nor render obvious applicants' claimed invention, as recited in independent claim 15.

Yet further, independent claim 21 is also patentable over Liu. As described above, Liu fails to describe, teach or suggest at least applicants' claimed element of generating a sequence of target machine instructions that directly calculates the condition code and provides distinguishing information to distinguish between a plurality of possible outcomes for the condition code, wherein at least part of the distinguishing information is separate from that condition code. Since this is not described, taught or suggested in Liu, Liu does not anticipate nor render obvious applicants' claimed invention, as recited in independent claim 21.

LzLabs GmbH. Ex. 1004-169

Moreover, applicants respectfully submit that the dependent claims are patentable for the same reasons as the independent claims, as well as for their own additional features. For instance, applicants specifically recite in dependent claim 29 that the distinguishing information distinguishes between outcomes for the following cases: a first operand and a second operand of a target machine instruction of the sequence of target machine instructions are equal; the first operand is greater than the second operand; and the first operand is less than the second operand. There is no description, teaching or suggestion in Liu of generating a sequence of target machine instructions, which derive an encoding for a target machine condition code and provides distinguishing information to distinguish between outcomes for the following cases: equal operands, the first operand being greater than a second operand, and the first operand being less than the second operand. Thus, applicants respectfully request an indication of allowability for dependent claim 29.

Further, dependent claim 27 specifically recites that the target machine instructions directly calculate the target machine condition code using at least one of addition, subtraction and multiplication. This is not described, taught or suggested in Liu, and therefore, applicants respectfully submit that Liu does not anticipate or render obvious this claim.

For at least the above reasons, applicants respectfully request an indication of allowability for all pending claims.

Should the Examiner wish to discuss this case with applicants' attorney, he is invited to contact the undersigned.

Respectfully submitted,

Blanche E. Schiller

Blanche E. Schiller
Attorney for Applicants
Registration No.: 35,670

Dated: December 21, 2009

HESLIN ROTHENBERG FARLEY & MESITI P.C.
5 Columbia Circle
Albany, New York 12203-5160
Telephone: (518) 452-5600
Facsimile: (518) 452-5579

LzLabs GmbH. Ex. 1004-170

Electronic Patent Application Fee Transmittal

Application Number:	11668623
Filing Date:	30-Jan-2007
Title of Invention:	METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS
First Named Inventor/Applicant Name:	Reid T. Copeland
Filer:	Blanche E. Schiller
Attorney Docket Number:	POU920060152US1

Filed as Large Entity

Utility under 35 USC 111(a) Filing Fees

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Basic Filing:				
Pages:				
Claims:				
Miscellaneous-Filing:				
Petition:				
Patent-Appeals-and-Interference:				
Post-Allowance-and-Post-Issuance:				
Extension-of-Time:				
Extension - 1 month with $0 paid	1251	1	130	130

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Miscellaneous:				
Request for continued examination	1801	1	810	810
		Total in USD ($)		**940**

Electronic Acknowledgement Receipt

EFS ID:	6681712
Application Number:	11668623
International Application Number:	
Confirmation Number:	4180
Title of Invention:	METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS
First Named Inventor/Applicant Name:	Reid T. Copeland
Customer Number:	46369
Filer:	Blanche E. Schiller
Filer Authorized By:	
Attorney Docket Number:	POU920060152US1
Receipt Date:	21-DEC-2009
Filing Date:	30-JAN-2007
Time Stamp:	16:57:40
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	yes
Payment Type	Deposit Account
Payment was successfully received in RAM	$940
RAM confirmation Number	3544
Deposit Account	081935
Authorized User	
The Director of the USPTO is hereby authorized to charge indicated fees and credit any overpayment as follows: Charge any Additional Fees required under 37 C.F.R. Section 1.16 (National application filing, search, and examination fees) Charge any Additional Fees required under 37 C.F.R. Section 1.17 (Patent application and reexamination processing fees)	

Charge any Additional Fees required under 37 C.F.R. Section 1.21 (Miscellaneous fees and charges)					
File Listing:					
Document Number	**Document Description**	**File Name**	**File Size(Bytes)/ Message Digest**	**Multi Part /.zip**	**Pages (if appl.)**
1	Extension of Time	POU920060152US1_EOT.pdf	64803 f4e80debbb2b0f4036a76e7a4b09b0261b0d835a	no	1
Warnings:					
Information:					
2	Request for Continued Examination (RCE)	POU920060152US1_RCE.pdf	697742 526789a86ec765be79941cbe6501a644dfaa250a	no	3
Warnings:					
Information:					
3		POU920060152US1_Amendment.pdf	675759 b26b782cae69bc81144830e17085eee352e82d53	yes	13

	Multipart Description/PDF files in .zip description		
	Document Description	**Start**	**End**
	Amendment Submitted/Entered with Filing of CPA/RCE	1	1
	Specification	2	3
	Claims	4	9
	Applicant Arguments/Remarks Made in an Amendment	10	13

Warnings:					
Information:					
4	Fee Worksheet (PTO-875)	fee-info.pdf	32473 71592bc032c9f29a6e8c214cd94a8777ad0c8d59	no	2
Warnings:					
Information:					
			Total Files Size (in bytes):	1470777	

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office
If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

PETITION FOR EXTENSION OF TIME UNDER 37 CFR 1.136(a) (Large Entity)	Docket No. POU920060152US1

In Re Application Of: **Reid T. Copeland**

Application No.	Filing Date	Examiner	Customer No.	Group Art Unit	Confirmation No.
11/668,623	2007-01-30	David J. Huisman	46369	2183	4180

Invention: **EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS WITHOUT EXECUTING BRANCH INSTRUCTIONS**

COMMISSIONER FOR PATENTS:

This is a request under the provisions of 37 CFR 1.136(a) to extend the period for filing a response to the Office Action of **August 20, 2009** (*Date*) in the above-identified application.

The requested extension is as follows (check time period desired):

☒ One month ☐ Two months ☐ Three months ☐ Four months ☐ Five months

from: **November 20, 2009** (*Date*) until: **December 20, 2009 (Sunday)** (*Date*)

The fee for the extension of time is **$130** and is to be paid as follows:

☐ A check in the amount of the fee is enclosed.

☒ The Director is hereby authorized to charge any fees which may be required, or credit any overpayment, to Deposit Account No. **08-1935**

☒ If an additional extension of time is required, please consider this a petition therefor and charge any additional fees which may be required to Deposit Account No. **08-1935**

☐ Payment by credit card. Form PTO-2038 is attached.

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

Blanche E. Schiller
Signature

Dated: **December 21, 2009**

Blanche E. Schiller, Esq.
Reg. No. 35,670
Heslin Rothenberg Farley & Mesiti P.C.
5 Columbia Circle
Albany, NY 12203
Telephone: (518) 452-5600
Facsimile: (518) 452-5579

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to "Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450" [37 CFR 1.8(a)] on _____.
(Date)

Signature of Person Mailing Correspondence

Typed or Printed Name of Person Mailing Correspondence

cc:

P12LARGE/REV11

PTO/SB/06 (07-06)
Approved for use through 1/31/2007. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

PATENT APPLICATION FEE DETERMINATION RECORD Substitute for Form PTO-875	Application or Docket Number 11/668,623	Filing Date 01/30/2007	☐ To be Mailed

APPLICATION AS FILED – PART I

	(Column 1)	(Column 2)	SMALL ENTITY ☐		OR	OTHER THAN SMALL ENTITY	
FOR	NUMBER FILED	NUMBER EXTRA	RATE ($)	FEE ($)		RATE ($)	FEE ($)
☐ BASIC FEE (37 CFR 1.16(a), (b), or (c))	N/A	N/A	N/A			N/A	
☐ SEARCH FEE (37 CFR 1.16(k), (i), or (m))	N/A	N/A	N/A			N/A	
☐ EXAMINATION FEE (37 CFR 1.16(o), (p), or (q))	N/A	N/A	N/A			N/A	
TOTAL CLAIMS (37 CFR 1.16(i))	minus 20 =	*	X $ =		OR	X $ =	
INDEPENDENT CLAIMS (37 CFR 1.16(h))	minus 3 =	*	X $ =			X $ =	
☐ APPLICATION SIZE FEE (37 CFR 1.16(s))	If the specification and drawings exceed 100 sheets of paper, the application size fee due is $250 ($125 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).						
☐ MULTIPLE DEPENDENT CLAIM PRESENT (37 CFR 1.16(j))							
* If the difference in column 1 is less than zero, enter "0" in column 2.			TOTAL			TOTAL	

APPLICATION AS AMENDED – PART II

		(Column 1)		(Column 2)	(Column 3)	SMALL ENTITY		OR	OTHER THAN SMALL ENTITY	
	12/21/2009	CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA	RATE ($)	ADDITIONAL FEE ($)		RATE ($)	ADDITIONAL FEE ($)
AMENDMENT	Total (37 CFR 1.16(i))	* 20	Minus	** 20	= 0	X $ =		OR	X $52=	0
	Independent (37 CFR 1.16(h))	* 3	Minus	***3	= 0	X $ =		OR	X $220=	0
	☐ Application Size Fee (37 CFR 1.16(s))									
	☐ FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							OR		
						TOTAL ADD'L FEE		OR	TOTAL ADD'L FEE	0

		(Column 1)		(Column 2)	(Column 3)					
		CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA	RATE ($)	ADDITIONAL FEE ($)		RATE ($)	ADDITIONAL FEE ($)
AMENDMENT	Total (37 CFR 1.16(i))	*	Minus	**	=	X $ =		OR	X $ =	
	Independent (37 CFR 1.16(h))	*	Minus	***	=	X $ =		OR	X $ =	
	☐ Application Size Fee (37 CFR 1.16(s))									
	☐ FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							OR		
						TOTAL ADD'L FEE		OR	TOTAL ADD'L FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.
** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20".
*** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3".
The "Highest Number Previously Paid For" (Total or Independent) is the highest number found in the appropriate box in column 1.

Legal Instrument Examiner:
/FREDERICK E. BRISCOE/

This collection of information is required by 37 CFR 1.16. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

LzLabs GmbH. Ex. 1004-177

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
11/668,623	01/30/2007	Reid T. Copeland	POU920060152US1	4180

46369 7590 08/20/2009

HESLIN ROTHENBERG FARLEY & MESITI P.C.
5 COLUMBIA CIRCLE
ALBANY, NY 12203

EXAMINER	
HUISMAN, DAVID J	
ART UNIT	PAPER NUMBER
2183	
MAIL DATE	DELIVERY MODE
08/20/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

LzLabs GmbH. Ex. 1004-178

Office Action Summary	Application No.	Applicant(s)	
	11/668,623	COPELAND ET AL.	
	Examiner	Art Unit	
	DAVID J. HUISMAN	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *16 June 2009*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *9-26* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *9-26* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *16 June 2009* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

LzLabs GmbH. Ex. 1004-179

DETAILED ACTION

1. Claims 9-26 have been examined.

Papers Submitted

2. It is hereby acknowledged that the following papers have been received and placed of record in the file: Amendment, Authorization for Extension of Time for all Replies, and Interview Request as received on 6/16/2009.

Interview Request

3. Applicant filed an interview request form on June 16, 2009, requesting that every claim rejection and any remaining issues be discussed. The examiner has reviewed all of applicant's issues with the rejections and asserts that there are no additional issues to be discussed at this point other than those set forth in the official response by applicant. MPEP 713.01 states that if it appears that the interview would result in expediting the case to a final action, then it should be granted. The examiner asserts that since the examiner needs no clarification on any issues set forth in applicant's response, any interview would only delay advancement to final status. Therefore, this submitted interview request is denied. Normally, one interview after final may be granted. See MPEP 713.01 and 713.09. Therefore, if applicant wishes to discuss any further issues after final, he/she is requested to contact the examiner.

Specification

4. The disclosure is objected to because of the following informalities:

- In paragraph [0035], line 8, replace "530" with --520--.
- On page 12 of the original specification, 2nd line from the bottom (in paragraph [0035]), "Figure 5" appears to have been replaced with --Figure 4-- but such an amendment was not indicated.
- In paragraph [0036], line 2, it appears that "except" has been replaced with --accept, but such an amendment was not indicated.
- The grammar of the second sentence on paragraph [0036] appears to be incorrect. Applicant should insert commas around "as input" and delete the comma after "305" for increased clarity.

Appropriate correction is required.

Claim Rejections - 35 USC § 112

5. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

6. Claims 14, 20, and 26 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Specifically, it is not clear what is meant by "using said result to reset itself/the result to a shorter bit configuration". How does a result reset itself? And, how are bits reduced? Also, applicant should not use the word "itself" (in claims 14 and 20), as it is not completely clear what applicant is referring to.

Claim Rejections - 35 USC § 101

7. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

8. Claims 9-14 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. Specifically, applicant states in paragraph [0035] of the specification that one type of medium is a distribution medium in the form of electrical signals used to transmit data bits. Though applicant claims a storage medium, it is not clear which of the exemplary media set forth in the specification applicant intends to be a storage medium, especially when applicant states that "While such transmitted signals may be ephemeral in nature they still, nonetheless constitute a physical medium carrying the coded instruction bits and are intended for permanent capture at the signal's destination or destinations". That is, it is not clear if applicant also intends for the distribution medium to be a type of storage medium. Claims directed to electrical signals do not fall into one of the four statutory categories of invention. Therefore, until applicant clarifies, these claims are deemed non-statutory.

Claim Rejections - 35 USC § 102

9. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

10. Claims 9-12, 14-18, 20-24 and 26 are rejected under 35 U.S.C. 102(b) as being anticipated by Liu et al., U.S. Patent No. 6,049,864 (herein referred to as Liu).

11. Referring to claim 21, Liu has taught a method for emulating computer instructions from a source machine (see Fig.3 and claims 1 and 3 of Liu, and note that instruction ADDF32 X1 = X2 + X3 is from a CISC machine) to produce sequences of instructions on a target machine (see Fig.3 and column 3, lines 4-6, and note that a sequence of addf32 and prodf32 instructions are produced to emulate the CISC instruction), said method comprising: generating a sequence of target machine instructions which together operate to directly calculate target machine condition codes from carry, sign and overflow codes without including branch instructions from the target machine. See Fig.3, and note that the emulation of the ADDF32 source instruction results in the execution of multiple instructions, none of which are branches, and at least one of which sets carry, overflow, and sign codes.

12. Referring to claim 22, Liu has taught the method of claim 21 in which said calculation of target machine condition codes is carried out in a sequence of non-branching instructions that manipulate condition code settings produced within the target machine itself to produce the same condition code in a target machine location. See Fig.3.

13. Referring to claim 23, Liu has taught the method of claim 22 in which said manipulation employs temporary locations for storing intermediate results. See Fig.3, and note that in response to the addf32 instruction, not only is the result produced and stored in X1{31:0}, but other intermediate values are stored in X1{32:38}. These intermediate results are then used by the prodf32 instruction to set flags.

14. Referring to claim 24, Liu has taught the method of claim 21 further including the step of executing said generated sequence of target machine instructions. See Fig.3.

15. Referring to claim 26, Liu has taught the method of claim 21 in which said generating comprises the steps of:

a) mimicking an instruction present in a source instruction stream in a manner which sets one or more target CPU flag bits and which places a result in a first storage location. See Fig.3 and note that a source CISC instruction is mimicked with at least two RISC instructions that set target flag bits and which also place a result in a first storage location. For purposes of examination, the first storage location is interpreted as bits X1{37:0}. The result of the addition is placed in these bits.

b) executing an instruction which uses said result to set at least one bit in second storage location so as to distinguish one or more case results. See Fig.3 and note that in response to executing the addf32 instruction, the result of the addition is used to set bit X1{38}, which is herein interpreted as the second storage location. This bit distinguishes between carry and no-carry case results.

c) using said result to reset itself to a shorter bit configuration which serves to distinguish at least one of said case results. See Fig.3. Though the actual addition result is 32 bits (in X1{31:0}), bit X1{31} is used to indicate positive or negative, and consequently, the addition result is reset to 31 bits (X1{30:0}).

d) executing an instruction which employs said first storage location to produce a result which distinguishes a different set of case results. See Fig.3 and note that bits X1{31:30} are used is setting codes which distinguish a different set of case results.

e) executing an instruction which uses the results in said first and second storage locations to provide an indication in which at least three cases are distinguished, said cases representing condition codes of said source machine. See Fig.3, and note that prodf32 uses the results in the

first storage location (specifically bit X1{37}) and the second storage location (specifically bit X1{38}) to provide carry and overflow indications that distinguish between results of the type: (1) no carry, no overflow, (2) no carry, overflow, (3) carry, no overflow, and (4) carry, overflow.

16. Referring to claims 9-12 and 14, claims 9-12 and 14 are rejected for the same reasons set forth in the rejections of claims 21-24 and 26, respectively.

17. Referring to claims 15-18 and 20, the systems of claims 15-18 and 20 perform the methods of claims 21-24 and 26, respectively. Consequently, claims 15-18 and 20 are rejected for the same reasons set forth in the rejections of claims 21-24 and 26, respectively. Also, it should be noted that Liu has taught a memory, both inherently and explicitly. That is, in Fig.3, Liu shows a memory X1[38:0] for storing data. Also, program instructions are inherently stored on some medium so that the processor may fetch and process them.

Claim Rejections - 35 USC § 103

18. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

19. Claims 13, 19, and 25 are rejected under 35 U.S.C. 103(a) as being unpatentable over Liu in view of the examiner's taking of Official Notice.

20. Referring to claim 25, Liu has taught the method of claim 24. Liu has not explicitly taught that said executing occurs at substantially the same time as said generating. However, the examiner asserts that interpretive environments, in which instructions are translated at run-time,

are well known and accepted in the art. Such systems reduce the amount of work done by the compiler and therefore, reduce compilation time. As a result, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Liu such that the generating of the sequence of instructions occurs at run-time when the source instruction is encountered, thereby resulting in execution occurring at substantially the same time as said generating.

21. Referring to claim 13, claim 13 is rejected for the same reasons set forth in the rejection of claim 25.

22. Referring to claim 19, the system of claim 19 performs the method of claim 25. Consequently, claim 19 is rejected for the same reasons set forth in the rejection of claim 25.

Response to Arguments

23. Applicant's arguments filed on June 16, 2009, have been fully considered but they are not persuasive.

24. Applicant argues the novelty/rejection of claim 1 on pages 12-13 of the remarks, in substance that:

> "The cited patent to Liu et al. is totally devoid of any reference whatsoever to anything that even remotely resembles emulation. The cited process in the patent to Liu neither generates, mimics or emulates sequences of computer instructions. Likewise, and to no surprise, the cited patent nowhere refers either to a source machine nor to a target machine."

25. These arguments are not found persuasive for the following reasons:

a) The examiner asserts that one merely needs to read the background section and/or claims 1 and 3, of Liu, to realize that Liu is directed to emulation. Specifically, the ADDF32 instruction set forth in Fig.3 is a CISC instruction. CISC instructions execute, natively, on more complex CISC machines. However, RISC machines are faster and simpler. Therefore, the idea behind

Liu is to emulate a CISC instruction by breaking it down into a sequence of simple RISC instructions (Fig.3). This way, the RISC machines are made compatible with older CISC software. In summation, even though the exact terms "source machine" and "target machine" are absent from Liu's disclosure, Liu is, in fact, directed to emulation.

26. Applicant argues the novelty/rejection of claim 1 on pages 13-14 of the remarks, in substance that:

> "We lay aside for the moment that Liu et al. teach absolutely nothing with respect to target machines (or to source machines), but rather focus on the fact that Liu et al. do in fact include branch instructions.
>
> Contrary to what the Examiner asserts to be the case, the fact is that Liu et al. not only fail to appreciate the lack of desirability for branching instructions (and specifically fail to appreciate it in an emulation environment) they do in fact teach their use, contrarily to the present specification and claims. To understand this requires a knowledge of the intricacies of the C programming language structure as used in the cited patent to Liu et al. In particular, the Examiner refers to Figure 3 from the cited patent as supporting the position that branch instructions are avoided in the teachings of Liu et al. However, the Examiner ignores the following language from Figure 4 from Liu et al.:
>
> AFLAGS.zero = (X1 {top_bit_pos:0} = = 0);
>
> In fact, this language structure is really a shorthand notation for the following pseudo-code (note the use of the "==" symbol, one which has a specific meaning in the description found in Liu et al.):
>
> ```
> if(X1 {31:0} ==0)
>     AFLAGS.zero = 1
> else
>     ALFAGS.zero = 0
> ```
>
> Support for Applicants' position with respect to this C language construction is found in the following excerpt from the Wikipedia article titled "C syntax," which states: "One of the operations implicitly forbidden is comparison: structures and unions cannot be compared using C's standard comparison facilities (==, >, <, etc.)." So, it is seen that the language that the Examiner points to as being indicative of "no branch" conditions, actually does in fact describe a comparison operation and its concomitant branching operation. In short, Liu et al. do not teach how to set condition codes without a branching operation. Nowhere do they teach, disclose or suggest that branching is to be avoided. In fact, the language found in the paragraph beginning at column 4, line 54 clearly recites the use of branch instructions. This appears to be the only place in the cited document where branching is even mentioned, and that is in the context of its being used as opposed to being avoided."

27. These arguments are not found persuasive for the following reasons:

a) While applicant is correct that "= =" indicates presence of a branch instruction, applicant's argument is directed to language not in the claims. That is, the claims are concerned with calculating carry, sign, and overflow codes. In Fig.3 of Liu, only the zero flag is calculated based on a branch instruction (i.e., only the zero flag is associated with "= ="). All of the claimed codes (carry, sign, and overflow), are calculated without a branch instruction. If applicant were to claim calculating a zero flag without using a branch instruction, Liu would be overcome. However, it should be realized that one of ordinary skill in the art would have recognized that the zero flag could be set in Liu without performing a branch. For instance, one could merely OR all of the X1 bits together and invert the result and assign it to the zero flag. There are number of alternatives to performing branching.

28. Applicant argues the novelty/rejection of claim 1 on pages 14-15 of the remarks, in substance that:

> "The Examiner is also ignoring the clear indication in Figure 5 from Liu et al. that their process always involves the use of a branching instruction. In this regard, particular attention is directed to the decision box in Figure 5 which asks the question: "Does the third instruction have a data dependency on the second instruction?" Applicants request that the Examiner pay particular attention to the overall thrust of the patent to Liu et al. In this regard, it is to be noted that documents cited against an applicant's claims are required to be considered in their entirety for all that they teach. The gist of this patent is the scheduling of the execution of instructions which may or may not have outcomes (results) that are dependent on condition codes set by earlier instructions. Thus, at its core, the subject patent to Liu et al. must make the inquiry as to this dependency. As such, they require a branch instruction to make this determination. This they show in their Figure 5."

29. These arguments are not found persuasive for the following reasons:

a) The examiner asserts that he is not ignoring any part of the Liu. Applicant's argument is first invalid because the scheduling of operations in Fig.5 has nothing to do with the actual calculation of condition codes. Fig.3 sets forth how the claimed codes are calculated and none of

them require branch instruction execution. Furthermore, even if the "branch" of Fig.5 was required to determine the codes, which is not the case, there is nothing in Liu that explicitly states that a branch instruction is executed to carry out the method of Fig.5. Fig.5 appears to set forth achievement of out-of-order scheduling based on dependency analysis. Such analysis is not uncommonly performed by a scoreboard-type structure, which comprises bits that indicate dependency/readiness of operations. That is, a particular bit may be detected by the hardware as having a particular value, and the corresponding instruction would be or would not be scheduled because on the value of the bit. A branch does not inherently need to be performed, and without explicit disclosure of a branch, it is not clear how applicant can come to the conclusion that Liu does not anticipate the claimed invention. Again though, this argument is moot because Fig.5 has nothing to do with the actual calculation of codes, which is set forth in Fig.3.

30. Applicant argues the novelty/rejection of claim 1 on page 15 of the remarks, in substance that:

> "Furthermore, it is seen from the patent to Liu et al. that no further explanation is provided in their section 4.1.2 to describe how the "has_even_parity" operation can be evaluated across different target platforms, without branches. Such teachings simply are not present. For this reason, as well as those stated above, it is seen that the rejection of the subject claims under 35 USC §102 cannot be sustained. It is therefore very respectfully requested that it be withdrawn."

31. These arguments are not found persuasive for the following reasons:

a) The even parity flag has nothing to do with the claimed flags (overflow, carry, sign). Hence, even if the even parity flag requires branches to calculate the flag, which the examiner does not concede, this is immaterial to the current claim1 and the rejection thereof.

Conclusion

32. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to DAVID J. HUISMAN whose telephone number is (571)272-4168. The examiner can normally be reached on Monday-Friday (8:00-4:30).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/David J. Huisman/
Primary Examiner, Art Unit 2183

Search Notes	Application/Control No.	Applicant(s)/Patent Under Reexamination
	11668623	COPELAND ET AL.
	Examiner	Art Unit
	DAVID J HUISMAN	2183

SEARCHED			
Class	Subclass	Date	Examiner
712	209, 221, 227	3-16-09	DJH
717	124, 134, 136, 137, 138	3-16-09	DJH
703	26	3-16-09	DJH

SEARCH NOTES		
Search Notes	Date	Examiner
EAST Search (all databases)	3/16/09	DJH
Inventor Name Search (PALM)	3/16/09	DJH
Updated EAST Search (all databases)	7-14-09	DJH

INTERFERENCE SEARCH			
Class	Subclass	Date	Examiner

	/DAVID J HUISMAN/ Primary Examiner.Art Unit 2183

EAST Search History

EAST Search History (Prior Art)

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	20	emulat$3 with ("condition code" "flag bit" "status bit") same (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/07/14 17:52
L2	0	emulat$3 with ("condition code" "flag bit" "status bit") same (overflow carry sign)	USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2009/07/14 17:53
S42	1	"20030093649".pn.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S43	86	emulat$3 with "condition code"	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S44	102	emulat$3 with ("condition code" "flag bit" "status bit") and (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S45	29	emulat$3 with ("condition code" "flag bit" "status bit") same branch and (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S46	68	("20020026633" | "4514803" | "4812975" | "5313614" | "5329471" | "5408637" | "5440702" | "5551015" | "5574873" | "5574927" | "5577233" | "5598560" | "5606696" | "5649203" | "5724590" | "5737631" | "5751982" | "5768593" | "5805792" | "5805895" | "5870575" | "5890000" |	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-193

		"5901308" | "5903760" | "5925124" | "5930509" | "5946491" | "5956495" | "5966539" | "5983335" | "6000028" | "6009261" | "6031994" | "6076155" | "6077314" | "6091897" | "6105124" | "6126329" | "6170083" | "6185675" | "6199095" | "6205545" | "6223339" | "6301705" | "6308318" | "6321379" | "6330691" | "6360194" | "6367067" | "6427234" | "6434741" | "6463582" | "6477641" | "6502237" | "6505296" | "6522985" | "6523173" | "6526572" | "6530078" | "6530079" | "6535903" | "6539471" | "6539541" | "6549987" | "6564297" | "6594824" | "6631514" | "6643630").PN.				
S47	6	(emulat$3 translat$3) same ("condition code" "flag bit" "status bit") same (without avoid $4) near2 branch$3 and (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-194

S48	144	(emulat$3 translat$3) with (without avoid$4) near2 branch$3 and (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S49	12	(emulat$3 translat$3) with (without avoid$4) near2 branch$3 and (overflow carry sign) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S50	5	("4951195" | "5241678" | "5301325" | "5313614" | "5598560").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S51	28	("5903760").URPN.	USPAT	OR	ON	2009/03/16 16:40
S52	68	("20020026633" | "4514803" | "4812975" | "5313614" | "5329471" | "5408637" | "5440702" | "5551015" | "5574873" | "5574927" | "5577233" | "5598560" | "5606696" | "5649203" | "5724590" | "5737631" | "5751982" | "5768593" | "5805792" | "5805895" | "5870575" | "5890000" | "5901308" | "5903760" | "5925124" | "5930509" | "5946491" | "5956495" | "5966539" | "5983335" | "6000028" | "6009261" | "6031994" | "6076155" | "6077314" | "6091897" |	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-195

		"6105124" | "6126329" | "6170083" | "6185675" | "6199095" | "6205545" | "6223339" | "6301705" | "6308318" | "6321379" | "6330691" | "6360194" | "6367067" | "6427234" | "6434741" | "6463582" | "6477641" | "6502237" | "6505296" | "6522985" | "6523173" | "6526572" | "6530078" | "6530079" | "6535903" | "6539471" | "6539541" | "6549987" | "6564297" | "6594824" | "6631514" | "6643630").PN.				
S53	68	("20020026633" | "4514803" | "4812975" | "5313614" | "5329471" | "5408637" | "5440702" | "5551015" | "5574873" | "5574927" | "5577233" | "5598560" | "5606696" | "5649203" | "5724590" | "5737631" | "5751982" | "5768593" | "5805792" | "5805895" | "5870575" | "5890000" |	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-196

		"5901308" | "5903760" | "5925124" | "5930509" | "5946491" | "5956495" | "5966539" | "5983335" | "6000028" | "6009261" | "6031994" | "6076155" | "6077314" | "6091897" | "6105124" | "6126329" | "6170083" | "6185675" | "6199095" | "6205545" | "6223339" | "6301705" | "6308318" | "6321379" | "6330691" | "6360194" | "6367067" | "6427234" | "6434741" | "6463582" | "6477641" | "6502237" | "6505296" | "6522985" | "6523173" | "6526572" | "6530078" | "6530079" | "6535903" | "6539471" | "6539541" | "6549987" | "6564297" | "6594824" | "6631514" | "6643630").PN.				
S54	723	(emulat$3 translat$3) with compar$4 with instruction	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-197

S55	21	(emulat$3 translat$3) with compar$4 with instruction same (code flag) near2 (condition status bit)	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S56	412	712/209.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S57	487	712/221.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S58	867	712/227.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S59	897	717/136.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S60	265	717/137.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S61	234	717/138.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S62	1	emulat$3 with ("condition code" "flag bit" "status bit") and (overflow carry sign)	USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2009/03/16 16:40
S63	1362	717/124.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S64	140	717/134.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S65	1560	(emulat$3 translat$3) same (arithmetic add $5 subtract$3) same (code flag) near2 (condition status bit)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S66	101	(emulat$3 translat$3) same (arithmetic add $5 subtract$3) with set $4 with (code flag) near2 (condition status bit)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S67	37	(emulat$3 translat$3) same (arithmetic add $5 subtract$3) with set $4 with (code flag) near2 (condition status bit) same (Z O S C sign zero overflow carry)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-198

S68	29	(emulat$3 translat$3) same (arithmetic add $5 subtract$3) with set $4 with (code flag) near2 (condition status bit) same (Z O S C sign zero overflow carry) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S69	7	("20020046298" "20040176941" "20050015754" "20070011664" "5301302" "5678032" "5966515").pn.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S70	4	S69 and (code flag) near2 (condition status bit)	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S71	119	simulat$3 near4 (code flag) near2 (condition status bit)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S72	67	simulat$3 near4 (code flag) near2 (condition status bit) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S73	35	simulat$3 near4 (code flag) near2 (condition status bit) and (overflow carry) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S74	127	(emulat$3 translat$3) and (host target) with set$4 with (code flag) near2 (condition status bit) and ((overflow O) and (carry C) and (sign S negative N)) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S75	59	(emulat$3 translat$3) and (host target) with set$4 with (code flag) near2 (condition status bit) and ((overflow O) and (carry C) and (sign S negative N)) and ("712"/$.ccls. "717"/$.ccls.)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-199

S76	32	emulat$3 and (host target) with set$4 with (code flag) near2 (condition status bit) and ((overflow O) and (carry C) and (sign S negative N)) and ("712"/$.ccls. "717"/$.ccls.)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S77	1	emulat$3 and (host target) with set$4 with (code flag) near2 (condition status bit) same ((overflow O) and (carry C) and (sign S negative N)) and ("712"/$.ccls. "717"/$.ccls.)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S78	1	emulat$3 and (host target) with set$4 with (code flag) near2 (condition status bit) same (((overflow O) and (carry C) and (sign S negative N))) and ("712"/$.ccls. "717"/$.ccls.)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
S79	6	("4415969" | "5327537" | "5440749" | "5493687" | "5590359" | "5689720").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S80	2	("5774694").URPN.	USPAT	OR	ON	2009/03/16 16:40
S81	8	("5313614" | "5481693" | "5619664" | "5774694" | "5859999" | "5860017" | "5884059" | "6049864").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
S82	1	"20060173931".pn.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:44
S83	446	703/26.ccls.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:46

7/14/09 5:57:53 PM
C:\Documents and Settings\DHuisman\My Documents\EAST\Workspaces\11-668623.wsp

LzLabs GmbH. Ex. 1004-200

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants:	Copeland et al.	Confirmation No.: 4180
Serial No.:	11/668,623	Group Art Unit: 2183
Filed:	01/30/2007	Examiner: David J. Huisman
Title:	EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS WITHOUT EXECUTING BRANCH INSTRUCTIONS (as amended)	

CERTIFICATE OF ELECTRONIC TRANSMISSION

I hereby certify that this correspondence is being transmitted electronically to: United States Patent and Trademark Office, Alexandria, VA 22313-1450, on June 16, 2009.

Jill K. Becker

Date of Signature: June 16, 2009

To: Mail Stop AMENDMENTS
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Response To Office Action

Dear Sir:

This Response to Office Action is being submitted in reply to the Office Action mailed March 24, 2009 for the above-referenced patent application. A response to the Office Action is initially due on or before June 24, 2009, and thus, this Response is timely filed.

Amendments to the Specification begin on page 2 of this paper.

Amendments to the Drawings begin on page 4 of this paper.

Amendments to the Claims begin on page 5 of this paper.

Remarks begin on page 10 of this paper.

LzLabs GmbH. Ex. 1004-201

Amendments to the Specification

Please amend the above-identified application, as follows:

Kindly amended the Title on pages 1 and 18, as follows:

EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS WITHOUT EXECUTING BRANCH INSTRUCTIONS

Kindly amend the Background of the Invention, as follows:

[0002] In virtually all modern data processing systems, the execution of various operations such as arithmetic operations, logical operations and even data transfer operations, may result in the generation of several bits of data to indicate the outcome status of instruction execution. These bits are typically referred to as condition codes. As a simple example, a special condition code setting may be set after an arithmetic addition which results in an overflow due to the addends being too large for the number of bits available for the result. The use of condition codes permeates the execution of almost every instruction.

Kindly amend the Detailed Description, as follows:

[0035] In any event the environment in which the present invention operates is shown in Figure 3. The present invention operates in a data processing environment which effectively includes one or more of the computer elements shown in Figure 3. In particular, computer 500 includes central processing unit (CPU) 520 which accesses programs and data stored within random access memory 510. Memory 510 is typically volatile in nature and accordingly such systems are provided with nonvolatile memory typically in the form of rotatable magnetic memory 540. While memory 540 is preferably a nonvolatile magnetic device, other media may be employed. CPU 530 communicates with users at consoles such as terminal 550 through Input/Output unit 520. Terminal 550 is typically one of many, if not thousands, of consoles in communication with computer 500 through one or more I/O unit 530. In particular, console unit 550 is shown as having included therein a device 560 for reading medium of one or more types such as CD-ROM 600 shown in Figure 4. Media 600,

LzLabs GmbH. Ex. 1004-202

an example of which is shown in Figure 4, comprises any convenient device including, but not limited to, magnetic media, optical storage devices and chips such as flash memory devices or so-called thumb drives. Disk 600 also represents a more generic distribution medium in the form of electrical signals used to transmit data bits which represent codes for the instructions discussed herein. While such transmitted signals may be ephemeral in nature they still, nonetheless constitute a physical medium carrying the coded instruction bits and are intended for permanent capture at the signal's destination or destinations.

[0036] The typical emulation environment in which the present invention is employed is illustrated in Figure 5. Emulators such as 320 accept as input instruction streams 305, representing machine or assembly language instructions which are designed to operate on source machine 300. Emulator 320 employees memory 315 in target machine 310 to produce [[a]] stream of instructions 325 which are capable of executing on target machine 310. While Figure 5 particularly shows operation within an emulation environment, it is also noted that the present invention contemplates a situation in which emulator 320 operates essentially as an interpreter in which the instructions are not only translated to the new architecture but in which they are also executed at essentially the same time.

LzLabs GmbH. Ex. 1004-203

Amendments to the Drawings

The drawings have been amended to include the "PRIOR ART" legend on Figure 1. Replacement sheets 1-4 are attached hereto. Substitute sheets for all of the figures are also included, not only to correct the spelling of "mimicking" but also to more closely comply with the regulations concerning the use of the caption "Fig. X" as opposed to "Figure X."

LzLabs GmbH. Ex. 1004-204

Amendments to the Claims

Kindly amend claims 9-26 as set forth below. The changes in the amended claims are shown by strikethrough for deleted matter and underlining for added matter.

1-8. (Canceled)

9. (Previously Presented) A computer program product for emulating computer instructions from a source machine to produce sequences of instructions on a target machine, said computer program product comprising:

a storage medium readable by a processor and storing instructions for execution by the processor for performing a method comprising:

generating a sequence of target machine instructions which together operate to directly calculate target machine condition codes from carry, sign and overflow codes without including branch instructions from the target machine.

10. (Previously Presented) The computer program product of claim 9, wherein said calculation of target machine condition codes is carried out in a sequence of non-branching instructions that manipulate condition code settings produced within the target machine itself to produce the same condition code in a target machine location.

11. (Previously Presented) The computer program product of claim 10, wherein said manipulation employs temporary locations for storing intermediate results.

12. (Previously Presented) The computer program product of claim 9, further including executing said generated sequence of target machine instructions.

13. (Previously Presented) The computer program product of claim 12, wherein said executing occurs at substantially the same time as said generating.

LzLabs GmbH. Ex. 1004-205

14. (Previously Presented) The computer program product of claim 9, wherein said generating comprises:

mimicking an instruction present in a source instruction stream in a manner which sets one or more target CPU flag bits and which places a result in a first storage location;

executing an instruction which uses said result to set at least one bit in a second storage location so as to distinguish one or more case results;

using said result to reset itself to a shorter bit configuration which serves to distinguish at least one of said case results;

executing an instruction which employs said first storage location to produce a result which distinguishes a different set of case results; and

executing an instruction which uses the results in said first and second storage locations to provide an indication in which at least three cases are distinguished, said cases representing condition codes of said source machine.

15. (Previously Presented) A computer system for emulating computer instructions from a source machine to produce sequences of instructions on a target machine, the computer system comprising:

a memory; and

a processor in communications with the memory, wherein the computer system is capable of performing a method, the method comprising:

generating a sequence of target machine instructions which together operate to directly calculate target machine condition codes from carry, sign and overflow codes without including branch instructions from the target machine.

LzLabs GmbH. Ex. 1004-206

16. (Previously Presented) The computer system of claim 15, wherein said calculation of target machine condition codes is carried out in a sequence of non-branching instructions that manipulate condition code settings produced within the target machine itself to produce the same condition code in a target machine location.

17. (Previously Presented) The computer system of claim 16, wherein said manipulation employs temporary locations for storing intermediate results.

18. (Previously Presented) The computer system of claim 15, further including executing said generated sequence of target machine instructions.

19. (Previously Presented) The computer system of claim 18, wherein said executing occurs at substantially the same time as said generating.

20. (Previously Presented) The computer system of claim 15, wherein said generating comprises:

mimicking an instruction present in a source instruction stream in a manner which sets one or more target CPU flag bits and which places a result in a first storage location;

executing an instruction which uses said result to set at least one bit in a second storage location so as to distinguish one or more case results;

using said result to reset itself to a shorter bit configuration which serves to distinguish at least one of said case results;

executing an instruction which employs said first storage location to produce a result which distinguishes a different set of case results; and

executing an instruction which uses the results in said first and second storage locations to provide an indication in which at least three cases are distinguished, said cases representing condition codes of said source machine.

LzLabs GmbH. Ex. 1004-207

21. (Previously Presented) A method for emulating computer instructions from a source machine to produce sequences of instructions on a target machine, said method comprising:

generating a sequence of target machine instructions which together operate to directly calculate target machine condition codes from carry, sign and overflow codes without including branch instructions from the target machine.

22. (Previously Presented) The method of claim 21, wherein said calculation of target machine condition codes is carried out in a sequence of non-branching instructions that manipulate condition code settings produced within the target machine itself to produce the same condition code in a target machine location.

23. (Previously Presented) The method of claim 22, wherein said manipulation employs temporary locations for storing intermediate results.

24. (Previously Presented) The method of claim 21, further including executing said generated sequence of target machine instructions.

25. (Previously Presented) The method of claim 24, wherein said executing occurs at substantially the same time as said generating.

26. (Currently amended) The method of claim 21, wherein said generating comprises:

mimicking an instruction present in a source instruction stream in a manner which sets one or more target CPU flag bits and which places a result in a first storage location;

executing an instruction which uses said result to set at least one bit in a second storage location so as to distinguish one or more case results;

using said result to reset ~~itself~~ the result to a shorter bit configuration which serves to distinguish at least one of said case results;

LzLabs GmbH. Ex. 1004-208

executing an instruction which employs said first storage location to produce a result which distinguishes a different set of case results; and

executing an instruction which uses the results in said first and second storage locations to provide an indication in which at least three cases are distinguished, said cases representing condition codes of said source machine.

LzLabs GmbH. Ex. 1004-209

Remarks

At present, claims 14, 20, and 26 stand as being rejected under 35 USC §112, second paragraph. Claims 9-14 are rejected under 35 USC §101. Claims 9-12, 14-18, 20-24 and 26 stand as being rejected under 35 USC §102(b) as being based upon the patent to Liu et al. (US Patent No. 6,049,864, issued April 11, 2000). Claims 13, 19, and 25 stand as being rejected under 35 USC §103(a) as being unpatentable over the aforementioned patent to Liu et al. in view of the Examiner's taking of Official Notice and asserting that "interpretive environments, in which instructions are translated at run-time, are well known and accepted in the art."

Entry of the amendments, reconsideration of the application, as amended, and allowance of all pending claims are respectfully requested. In light of the comments herein and the modifications made above, Applicants respectfully, but most strenuously, traverse all of the rejections for the reasons set forth below. Accordingly, claims 9-26 remain pending in the instant application.

With the above amendments, Applicants are distinctly claiming and particularly pointing out one or more features of their invention. These claim amendments are being provided to clarify Applicants' invention and are not in any way an acquiescence to the art based rejections. Support for the amendments is found throughout Applicants' specification.

Preliminary Matters

The Examiner has asserted that the title of the invention is not sufficiently descriptive. In order to address the Examiner's concerns in this regard, Applicants have herein amended the title in the manner suggested by the Examiner. However, it is understood that this modification to the title of the patent application is not in any way intended to affect the scope of the claims. In particular, the added language is not intended in any way to be a claim limitation.

The Examiner has also objected to Applicant's specification and asserts that "Applicant fails to provide proper antecedent basis for 'computer program product ... comprising a storage medium readable by a processor.' " In this regard, it is noted that the

LzLabs GmbH. Ex. 1004-210

specification refers to several possible storage media: memory 510 in Figure 3, media 600 shown in Figure 4 (reference corrected herein) or rotatable magnetic memory 540. Any one of these media would be capable and usable as supporting Applicant's claim for a computer program product even though the most typical form in which a computer program product would be embodied would be the CD_ROM 600. However, it is noted that reference numeral 600 refers to a variety of possible storage media including, but not limited to, magnetic media, optical storage devices and chips such as flash memory devices or so-called thumb drives. Accordingly, it is seen that the antecedent basis for this language from Applicants' "program product claims" is amply described in the written description and in the drawings as well. Accordingly, it is requested that either (1) this objection be more clearly defined, or (2) that it be withdrawn in light of the discussion above.

The Examiner has also objected to the specification based on a number of obvious grammatical or typographical errors none of which rises to any substantive level. Nonetheless, Applicants are appreciative of the Examiner's attention to detail and appreciate the opportunity to have corrected these informalities. The modifications made above should remove any pending concerns with respect to these informalities. However, having accepted (no pun intended) the change from "except" to "accept" in Applicants' paragraph **[0035]**, Applicants' discern no other problem with the sentence in question and agree completely with the Examiner's interpretation with the change in place.

The Examiner has also objected to Applicants' Figure 1 in asserting that it should be labeled as "PRIOR ART." That change has been made with the drawings submitted herewith.

The Examiner has also objected to two spelling errors in Figure 2. These are also corrected via the drawing sheets submitted herewith. Applicants have also amended the other drawings as well so as to change "Figure 1" to "Fig. 1" so as to more closely comply with the regulations in this regard. Accordingly, the drawing rejections should now be withdrawn.

LzLabs GmbH. Ex. 1004-211

The Examiner has also objected to the drawings in that reference numerals 560 and 325 were not found in Applicants' specification. This problem has been corrected via the modifications made above.

Claim 26 is also objected to because of a typographical error which has been corrected herein.

Rejection under 35 USC § 102

It is first noted that a rejection under 35 USC §102 is a narrow ground of rejection. It requires each and every claim element to be found within the four corners of a single cited document. Any exceptions to this rule are not germane to the present discussion.

Some preliminary, overall comparisons between the cited art and the subject invention are useful to consider. Notably, the present invention is directed to a method and system used to derive an encoding of bits (as required by target machines being emulated). In contrast, Liu et al. only consider the production of a set of discrete bits that are fairly easy to derive, notably when the target machine provides a very similar set of result bits as shown in their examples. The examples provided in the specification for the present invention describe how to derive an encoding of 0, 1 or 2 (with binary bit patterns of 00, 01, 10, respectively) given only the relative ordering of two operands in the emulated instruction. This is to be contrasted with the patent to Liu et al., where individual condition code bits are simply transferred or trivially derived from similar incoming intermediate flags. In this regard, Applicant's paragraph **[0031]** is particularly worthy of consideration and comparison.

It is also noted that the claims in question (9-12, 14-18, 20-24 and 26) are directed to emulation. That is to say, one or more aspects of the present invention are directed to a process in which computer instructions on a source machine are processed so as to generate one or more computer instructions that are contained within the architecture of a target computer. In addition to language that refers to "generation," the language of the subject claims refers also to "mimicking" an instruction. Applicants' claimed invention is therefore one that is directed to emulation. The cited patent to Liu et al. is totally devoid of any reference whatsoever to anything that even remotely resembles emulation. The cited process

LzLabs GmbH. Ex. 1004-212

in the patent to Liu neither generates, mimics or emulates sequences of computer instructions. Likewise, and to no surprise, the cited patent nowhere refers either to a source machine nor to a target machine.

Accordingly, since rejections under 35 USC §102 are narrow grounds of rejection and since all of the rejected claims refer to emulation and since the cited patent to Liu et al. is devoid of any reference, suggestion or hint of emulation, it is seen that the art fails to teach the core of what is being claimed. Since the art fails in this respect, it is therefore very respectfully requested that the rejection under 35 USC §102 be withdrawn.

This, however, is not the only failure with respect to the teachings of Liu et al. The Examiner is reminded that Applicants' claim 9 specifically refers to "without including branch instructions from the target machine." We lay aside for the moment that Liu et al. teach absolutely nothing with respect to target machines (or to source machines), but rather focus on the fact that Liu et al. do in fact include branch instructions.

Contrary to what the Examiner asserts to be the case, the fact is that Liu et al. not only fail to appreciate the lack of desirability for branching instructions (and specifically fail to appreciate it in an emulation environment) they do in fact teach their use, contrarily to the present specification and claims. To understand this requires a knowledge of the intricacies of the C programming language structure as used in the cited patent to Liu et al. In particular, the Examiner refers to Figure 3 from the cited patent as supporting the position that branch instructions are avoided in the teachings of Liu et al. However, the Examiner ignores the following language from Figure 4 from Liu et al.:

AFLAGS.zero = (X1 {top_bit_pos:0} = = 0);

In fact, this language structure is really a shorthand notation for the following pseudo-code (note the use of the "==" symbol, one which has a specific meaning in the description found in Liu et al.):

```
if (X1{31:0} == 0)
        AFLAGS.zero = 1
else
        ALFAGS.zero = 0
```


LzLabs GmbH. Ex. 1004-213

Support for Applicants' position with respect to this C language construction is found in the following excerpt from the Wikipedia article titled "C syntax," which states: "One of the operations implicitly forbidden is comparison: structures and unions cannot be compared using C's standard comparison facilities (==, >, <, etc.)." So, it is seen that the language that the Examiner points to as being indicative of "no branch" conditions, actually does in fact describe a comparison operation and its concomitant branching operation. In short, Liu et al. do not teach how to set condition codes without a branching operation. Nowhere do they teach, disclose or suggest that branching is to be avoided. In fact, the language found in the paragraph beginning at column 4, line 54 clearly recites the use of branch instructions. This appears to be the only place in the cited document where branching is even mentioned, and that is in the context of its being used as opposed to being avoided.

In further support of Applicants' position in this regard, the Examiner's attention is also directed to the following excerpt from the Wikipedia article titled "Relational operator":

> "In languages that use "==" to **test for equality** (usually C-like languages), the "==" operator is distinct from the "=" operator, the latter being used for assignment. Languages that use this style include all C-style languages, such as C, Java, and PHP." [Emphasis added herein.]

Elsewhere in the same article, the definition of a relational operator is set forth:

> "In computer science a relational operator is a programming language construct or operator that **tests** some kind of relation between two entities." [Emphasis added herein.]

It is thus unequivocally the case that the use of "==" by Liu et al. teaches the use of a test. The use of a test result inevitably infers the use of branching instructions that arise out of the use of the test result, for otherwise, what would be their purpose?

The Examiner is also ignoring the clear indication in Figure 5 from Liu et al. that their process **always** involves the use of a branching instruction. In this regard, particular attention is directed to the decision box in Figure 5 which asks the question: "Does the third instruction have a data dependency on the second instruction?" Applicants request that the Examiner pay particular attention to the overall thrust of the patent to Liu et al. In this regard, it is to be noted that documents cited against an applicant's claims are required to be

LzLabs GmbH. Ex. 1004-214

considered in their entirety for all that they teach. The gist of this patent is the scheduling of the execution of instructions which may or may not have outcomes (results) that are dependent on condition codes set by earlier instructions. Thus, at its core, the subject patent to Liu et al. must make the inquiry as to this dependency. As such, they require a branch instruction to make this determination. This they show in their Figure 5. It is also seen in the following excerpt from their patent:

> **Liu et al. (Column 2, lines 43-47)**: The second instruction is scheduled to execute before the third instruction **if** the third as instruction has a data dependency on the second instruction, otherwise the third instruction may be scheduled to execute before the second instruction. [Emphasis added herein.]

For Liu et al., it is imperative that this determination be made and it can only be made with the use of a branch instruction. Accordingly, it is seen that a closer reading of the teachings found in the cited patent actually teach away from an aspect of the claimed invention.

Furthermore, it is seen from the patent to Liu et al. that no further explanation is provided in their section 4.1.2 to describe how the "has_even_parity" operation can be evaluated across different target platforms, without branches. Such teachings simply are not present. For this reason, as well as those stated above, it is seen that the rejection of the subject claims under 35 USC §102 cannot be sustained. It is therefore very respectfully requested that it be withdrawn.

Rejection under 35 USC §103

The Examiner has also rejected Applicants' claims 13, 19, and 25 based on the above cited patent to Liu et al. together with his having taken official notice and has asserted that "interpretive environments, in which instructions are translated at run-time, are well known and accepted in the art."

As discussed above, it is seen that Liu et al. do not in fact teach away from the use of branching instructions. However, this aspect of the present invention is either a stated or implied feature present in Applicants' claims. In fact, a close examination of the patent to Liu et al. reveals that they actually do require a branching operation as indicated by their use of the double equal sign (==), a symbol well known in the programming and computer

LzLabs GmbH. Ex. 1004-215

description arts to represent a conditional instruction, which necessarily involves a branching operation. Since the base document upon which the Examiner relies fails to include any discussion whatsoever concerning the avoidance of branch instructions and since it actually teaches away from the claimed invention, it is seen that it cannot support a ground for rejection under 35 USC §103, with or without the Examiner's Official Notice.

For all of the reasons set forth above, Applicants respectfully submit that their invention is patentable over the cited document and over the Examiner's official notice, and respectfully request an indication of allowability for all of the pending claims.

Applicant's attorney wishes to point out to the Examiner that should the Examiner find it either necessary or desirable to discuss the present response, that he would be willing to discuss any matter which would assist the Examiner in furthering the prosecution of the present application with all due and proper regard for the appropriate scope of the invention. Accordingly, should the Examiner wish to discuss this case with Applicant's attorney, the Examiner may contact Applicant's attorney via the number listed below.

If the Examiner has not had the opportunity to discuss this application telephonically with the inventors' attorney prior to receiving this response, Applicants respectfully request a telephonic interview with the Examiner at the time the Examiner considers this Response, if the Examiner continues to have concerns over the claimed invention. An Applicant Initiated Interview Request Form is included herewith.

Respectfully submitted,

Blanche E. Schiller

Blanche E. Schiller
Attorney for Applicants
Registration No.: 35,650

Dated: June 16, 2009

HESLIN ROTHENBERG FARLEY & MESITI P.C
5 Columbia Circle
Albany, New York 12203-5160
Telephone: (518) 452-5600
Facsimile: (518) 452-5579

LzLabs GmbH. Ex. 1004-216

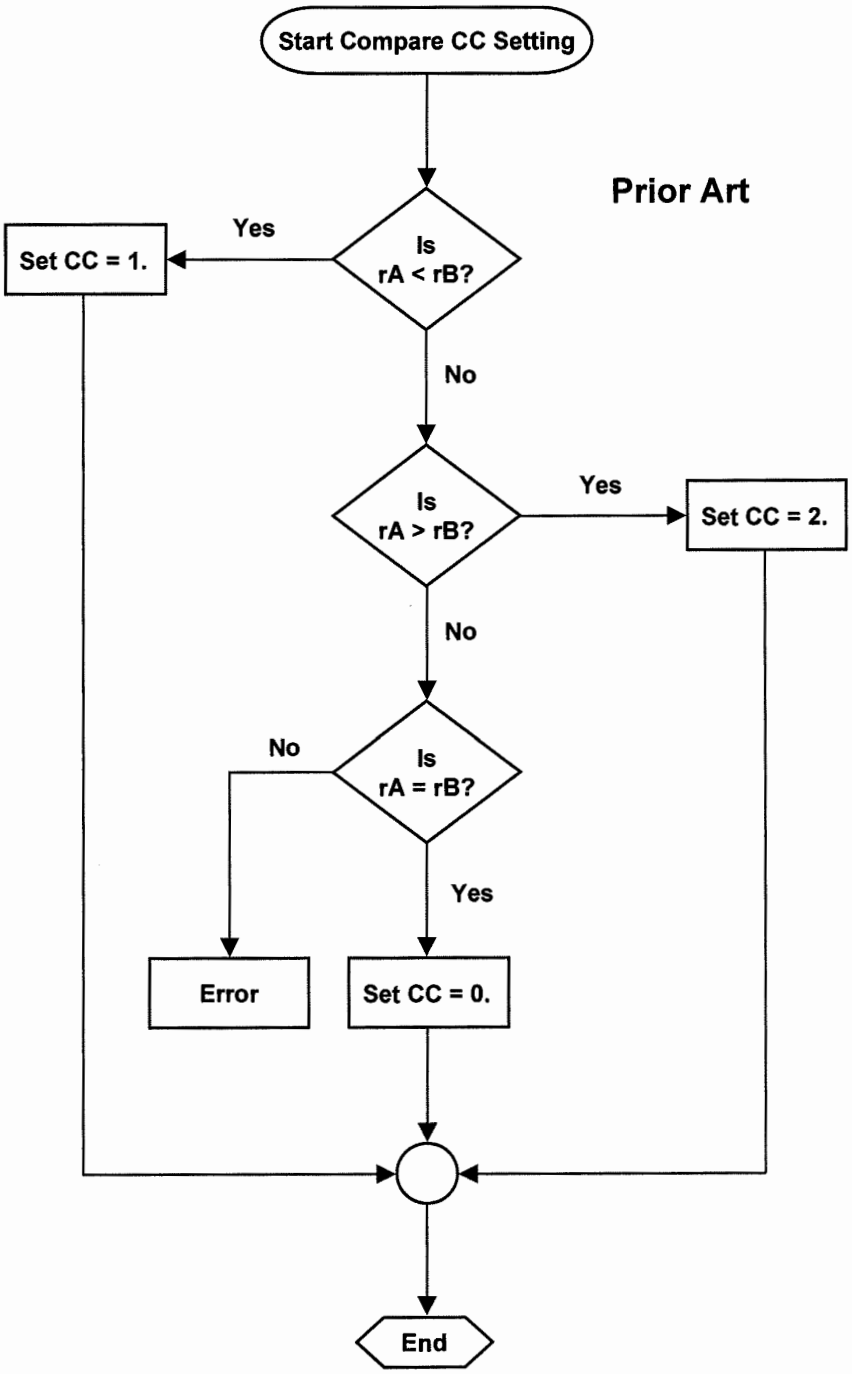

Start Compare CC Setting
Prior Art
Is rA < rB?
Yes
Set CC = 1.
No
Is rA > rB?
Yes
Set CC = 2.
No
Is rA = rB?
No
Yes
Error
Set CC = 0.
End

Fig. 1

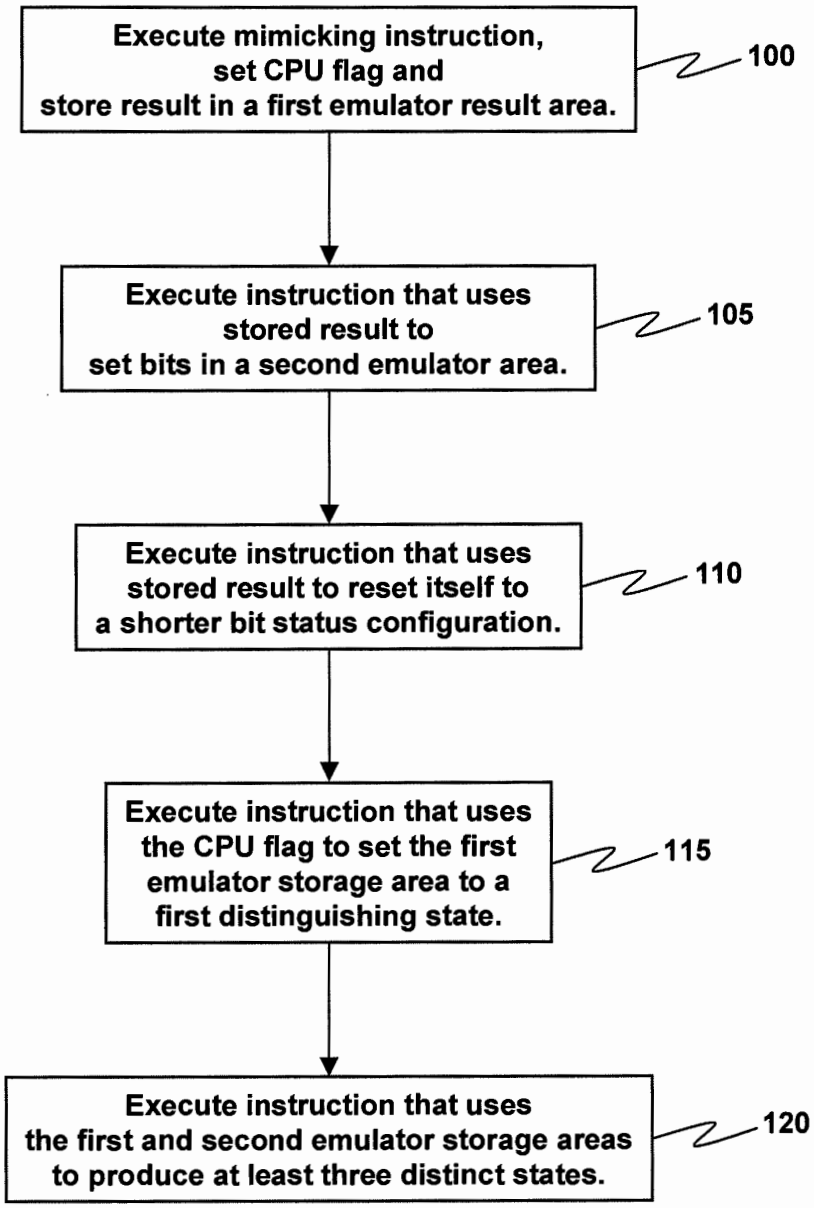

Execute mimicking instruction, set CPU flag and store result in a first emulator result area.
100
Execute instruction that uses stored result to set bits in a second emulator area.
105
Execute instruction that uses stored result to reset itself to a shorter bit status configuration.
110
Execute instruction that uses the CPU flag to set the first emulator storage area to a first distinguishing state.
115
Execute instruction that uses the first and second emulator storage areas to produce at least three distinct states.
120

Fig. 2

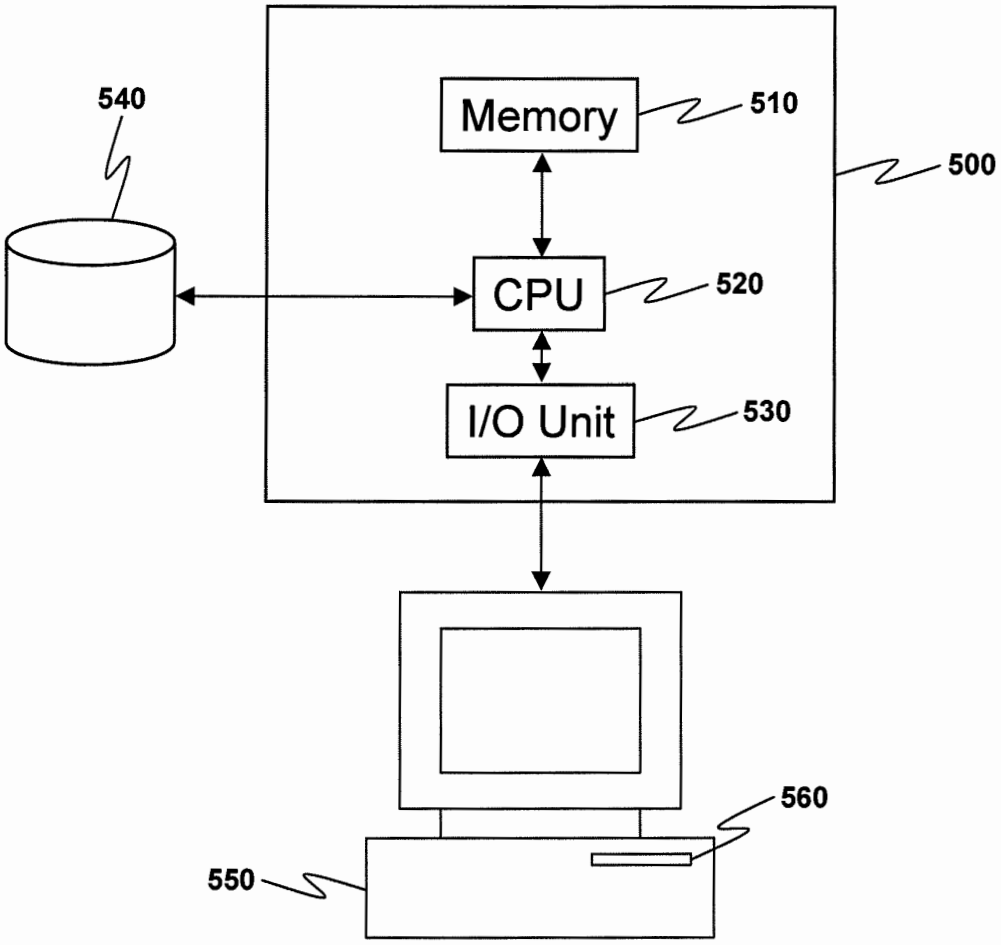


Fig. 3

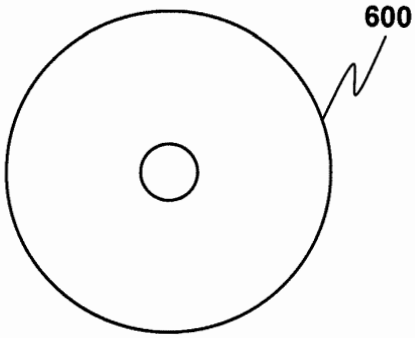


Fig. 4

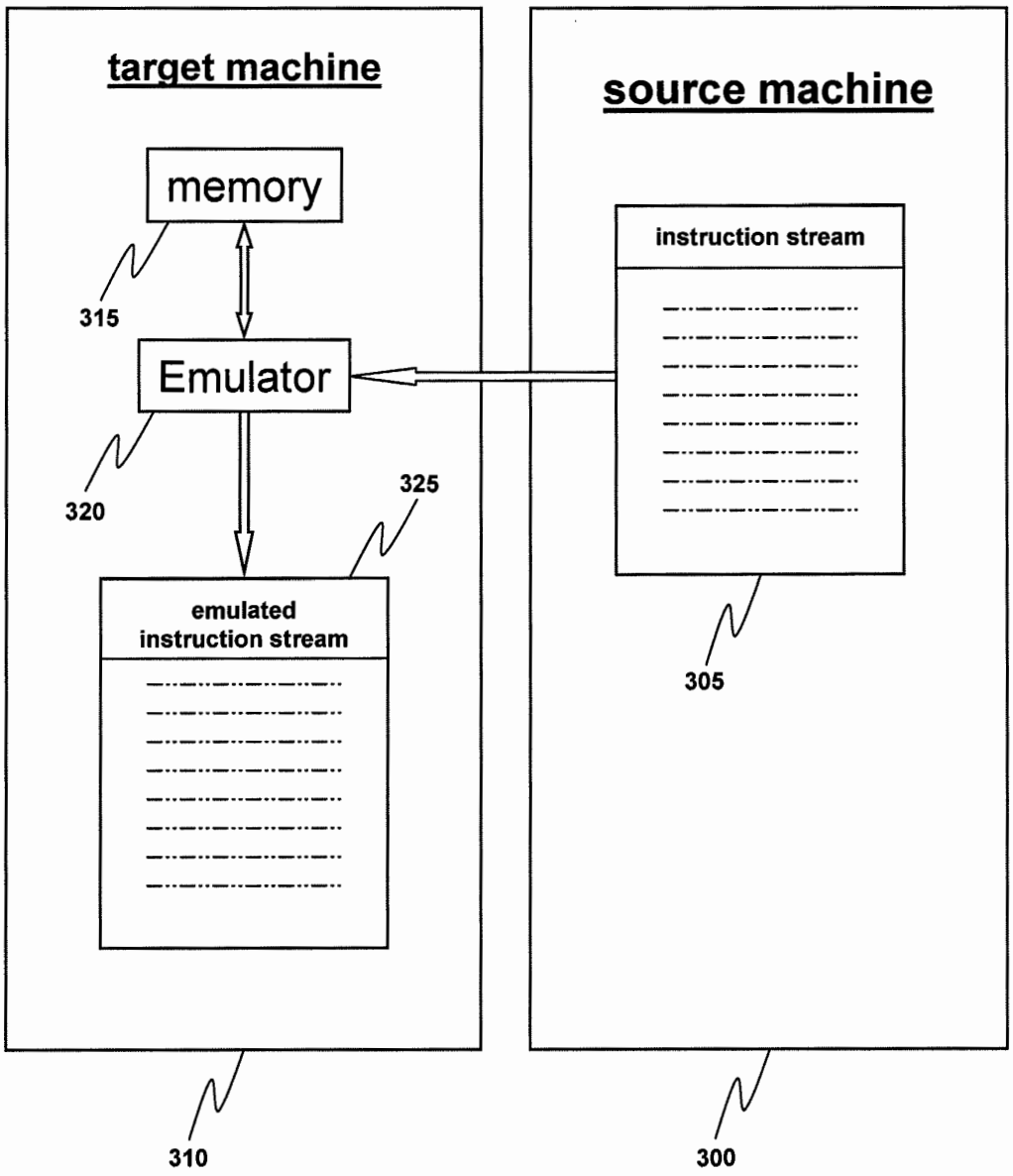

target machine
source machine
memory
315
Emulator
320
325
emulated
instruction stream
instruction stream
305
310
300

Fig. 5

PTOL-413A (12-08)
Approved for use through 01/31/2009. OMB 0651-0031
U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

Applicant Initiated Interview Request Form

Application No.: 11/668,623 First Named Applicant: Reid T. Copeland
Examiner: David J. Huisman Art Unit: 2183 Status of Application: Pending

Tentative Participants:
(1) Examiner David J. Huisman (2) Blanche E. Schiller, Esq.
(3) (4)

Proposed Date of Interview: At Examiner's convenience **Proposed Time:** (AM/PM)

Type of Interview Requested:
(1) [×] Telephonic (2) [ ] Personal (3) [ ] Video Conference

Exhibit To Be Shown or Demonstrated: [ ] YES [×] NO
If yes, provide brief description:

Issues To Be Discussed

Issues (Rej., Obj., etc.)	Claims / Fig. #s	Prior Art	Discussed	Agreed	Not Agreed
(1) 112 Rej.	14, 20 & 26		[ ]	[ ]	[ ]
(2) 101 Rej.	9-14		[ ]	[ ]	[ ]
(3) 102(b) Rej.	9-12, 14-18, 20-24 & 26	Liu et al.	[ ]	[ ]	[ ]
(4) 103(a) Rej.	13, 29 & 25	Liu et al. & Official Notice	[ ]	[ ]	[ ]

[ ] Continuation Sheet Attached

Brief Description of Arguments to be Presented:
To discuss any remaining issues that the Examiner may have.

An interview was conducted on the above-identified application on .

NOTE: This form should be completed by applicant and submitted to the examiner in advance of the interview (see MPEP § 713.01).
This application will not be delayed from issue because of applicant's failure to submit a written record of this interview. Therefore, applicant is advised to file a statement of the substance of this interview (37 CFR 1.133(b)) as soon as possible.

Blanche E. Schiller
Applicant / Applicant's Representative Signature

Blanche E. Schiller, Esq.
Typed/Printed Name of Applicant or Representative

35,670
Registration Number, if applicable

Examiner / SPE Signature

This collection of information is required by 37 CFR 1.133. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 21 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. **DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

Electronic Acknowledgement Receipt

EFS ID:	5526918
Application Number:	11668623
International Application Number:	
Confirmation Number:	4180
Title of Invention:	METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS
First Named Inventor/Applicant Name:	Reid T. Copeland
Customer Number:	46369
Filer:	Blanche E. Schiller
Filer Authorized By:	
Attorney Docket Number:	POU920060152US1
Receipt Date:	16-JUN-2009
Filing Date:	30-JAN-2007
Time Stamp:	16:37:16
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1		POU920060152US1_PR01.pdf	939656 d6ee8ab21b6041870b1d6df690116eacb87b6f28	yes	22

	Multipart Description/PDF files in .zip description		
	Document Description	**Start**	**End**
	Miscellaneous Incoming Letter	1	1
	Amendment/Req. Reconsideration-After Non-Final Reject	2	2
	Specification	3	5
	Claims	6	10
	Applicant Arguments/Remarks Made in an Amendment	11	17
	Drawings-only black and white line drawings	18	21
	Letter Requesting Interview with Examiner	22	22
Warnings:			
Information:			
	Total Files Size (in bytes):	939656	

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

<u>New Applications Under 35 U.S.C. 111</u>
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

<u>National Stage of an International Application under 35 U.S.C. 371</u>
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

<u>New International Application Filed with the USPTO as a Receiving Office</u>
If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

AMENDMENT TRANSMITTAL LETTER (Large Entity) Applicant(s): **Copeland et al.**					Docket No. **POU920060152US1**
Application No. **11/668,623**	Filing Date **01/30/2007**	Examiner **David J. Huisman**	Customer No. **46369**	Group Art Unit **2183**	Confirmation No. **4180**

Invention: **EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS WITHOUT EXECUTING BRANCH INSTRUCTIONS (as amended)**

COMMISSIONER FOR PATENTS:

Transmitted herewith is an amendment in the above-identified application.

The fee has been calculated and is transmitted as shown below.

CLAIMS AS AMENDED

	CLAIMS REMAINING AFTER AMENDMENT		HIGHEST # PREV. PAID FOR		NUMBER EXTRA CLAIMS PRESENT	RATE	ADDITIONAL FEE
TOTAL CLAIMS	18	-	20	=	0	x $52.00	$0.00
INDEP. CLAIMS	3	-	3	=	0	x $220.00	$0.00
Multiple Dependent Claims (check if applicable) ☐							$0.00
TOTAL ADDITIONAL FEE FOR THIS AMENDMENT							$0.00

☒ No additional fee is required for amendment.

☐ Please charge Deposit Account No. in the amount of

☐ A check in the amount of to cover the filing fee is enclosed.

☒ The Director is hereby authorized to charge payment of the following fees associated with this communication or credit any overpayment to Deposit Account **09-0463 (IBM)**

- ☒ Any additional filing fees required under 37 C.F.R. 1.16.
- ☒ Any patent application processing fees under 37 CFR 1.17.

☐ Payment by credit card. Form PTO-2038.

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

Blanche E. Schiller
Signature

Dated: **June 16, 2009**

Blanche E. Schiller, Esq.
Reg. No. 35,670
Heslin Rothenberg Farley & Mesiti P.C.
5 Columbia Circle
Albany, NY 12203
Telephone: (518) 452-5600
Facsimile: (518) 452-5579

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to "Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450" [37 CFR 1.8(a)] on

_____ .
(Date)

Signature of Person Mailing Correspondence

Typed or Printed Name of Person Mailing Correspondence

cc:

P11LARGE/REV10

PTO/SB/06 (07-06)
Approved for use through 1/31/2007. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

PATENT APPLICATION FEE DETERMINATION RECORD Substitute for Form PTO-875	Application or Docket Number 11/668,623	Filing Date 01/30/2007	☐ To be Mailed

APPLICATION AS FILED – PART I

	(Column 1)	(Column 2)	SMALL ENTITY ☐		OR	OTHER THAN SMALL ENTITY	
FOR	NUMBER FILED	NUMBER EXTRA	RATE ($)	FEE ($)		RATE ($)	FEE ($)
☒ BASIC FEE (37 CFR 1.16(a), (b), or (c))	N/A	N/A	N/A			N/A	300
☐ SEARCH FEE (37 CFR 1.16(k), (i), or (m))	N/A	N/A	N/A			N/A	
☐ EXAMINATION FEE (37 CFR 1.16(o), (p), or (q))	N/A	N/A	N/A			N/A	
TOTAL CLAIMS (37 CFR 1.16(i))	8 minus 20 =	* 0	X $ =		OR	X $50 =	0
INDEPENDENT CLAIMS (37 CFR 1.16(h))	3 minus 3 =	* 0	X $ =			X $200 =	0
☐ APPLICATION SIZE FEE (37 CFR 1.16(s))	If the specification and drawings exceed 100 sheets of paper, the application size fee due is $250 ($125 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).						
☐ MULTIPLE DEPENDENT CLAIM PRESENT (37 CFR 1.16(j))							
* If the difference in column 1 is less than zero, enter "0" in column 2.			TOTAL			TOTAL	300

APPLICATION AS AMENDED – PART II

		(Column 1)		(Column 2)	(Column 3)	SMALL ENTITY		OR	OTHER THAN SMALL ENTITY	
AMENDMENT	06/16/2009	CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA	RATE ($)	ADDITIONAL FEE ($)		RATE ($)	ADDITIONAL FEE ($)
	Total (37 CFR 1.16(i))	* 18	Minus	** 20	= 0	X $ =		OR	X $52=	0
	Independent (37 CFR 1.16(h))	* 3	Minus	***3	= 0	X $ =		OR	X $220=	0
	☐ Application Size Fee (37 CFR 1.16(s))									
	☐ FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							OR		
						TOTAL ADD'L FEE		OR	TOTAL ADD'L FEE	0

		(Column 1)		(Column 2)	(Column 3)					
AMENDMENT		CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA	RATE ($)	ADDITIONAL FEE ($)		RATE ($)	ADDITIONAL FEE ($)
	Total (37 CFR 1.16(i))	*	Minus	**	=	X $ =		OR	X $ =	
	Independent (37 CFR 1.16(h))	*	Minus	***	=	X $ =		OR	X $ =	
	☐ Application Size Fee (37 CFR 1.16(s))									
	☐ FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							OR		
						TOTAL ADD'L FEE		OR	TOTAL ADD'L FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.

** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20".

*** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3".

The "Highest Number Previously Paid For" (Total or Independent) is the highest number found in the appropriate box in column 1.

Legal Instrument Examiner:
/JESSICA GAYNOR/

This collection of information is required by 37 CFR 1.16. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
11/668,623	01/30/2007	Reid T. Copeland	POU920060152US1	4180

46369 7590 03/24/2009

HESLIN ROTHENBERG FARLEY & MESITI P.C.
5 COLUMBIA CIRCLE
ALBANY, NY 12203

EXAMINER	
HUISMAN, DAVID J	
ART UNIT	PAPER NUMBER
2183	
MAIL DATE	DELIVERY MODE
03/24/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

LzLabs GmbH. Ex. 1004-226

Office Action Summary	Application No.	Applicant(s)	
	11/668,623	COPELAND ET AL.	
	Examiner	Art Unit	
	DAVID J. HUISMAN	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *20 February 2009*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *9-26* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *9-26* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *30 January 2007* is/are: a)☐ accepted or b)☒ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *1/30/2007 & 11/7/2008*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

LzLabs GmbH. Ex. 1004-227

DETAILED ACTION

1. Claims 9-26 have been examined.

Papers Submitted

2. It is hereby acknowledged that the following papers have been received and placed of record in the file: IDS as received on 1/30/2007 and 11/7/2008, and Preliminary Amendment as received on 2/20/2009.

Information Disclosure Statement

3. There are two copies of the same IDS filed on November 7, 2008, in the electronic file. Hence, only one copy has been considered and initialed. The duplicate copy has been marked "not considered".

Specification

4. The title of the invention is not descriptive. A new title is required that is clearly indicative of the invention to which the claims are directed. The examiner recommends amending the title to include the concept of producing emulated condition code settings without executing branch instructions.

5. The specification is objected to as failing to provide proper antecedent basis for the claimed subject matter. See 37 CFR 1.75(d)(1) and MPEP § 608.01(o). Correction of the following is required: Applicant fails to provide proper antecedent basis for "computer program

product…comprising a storage medium readable by a processor". Applicant should make it clear in the specification as to which media are considered storage media.

6. The disclosure is objected to because of the following informalities:

- At the end of paragraph [0002], please insert a period.
- In paragraph [0035], line 8, replace "530" with --520--.
- On page 12, 2nd line from the bottom (in paragraph [0035]), replace "Figure 5" with --Figure 4--.
- In paragraph [0036], line 2, replace "except" with --accept. Also, the grammar of the second sentence on paragraph [0036] appears to be incorrect. Should the sentence instead be "Emulators such as 320 accept, as input, instruction streams 305 representing machine or assembly language instructions which are designed to operate on source machine 300."

Appropriate correction is required.

Drawings

7. Figure 1 should be designated by a legend such as --Prior Art-- because only that which is old is illustrated. See MPEP § 608.02(g). Corrected drawings in compliance with 37 CFR 1.121(d) are required in reply to the Office action to avoid abandonment of the application. The replacement sheet(s) should be labeled "Replacement Sheet" in the page header (as per 37 CFR 1.84(c)) so as not to obstruct any portion of the drawing figures. If the changes are not accepted by the examiner, the applicant will be notified and informed of any required corrective action in the next Office action. The objection to the drawings will not be held in abeyance.

8. Fig.2 is objected to because:

- In step 100, "mimicing" should be replaced with --mimicking--.
- In step 105, "sets" should be replaced with --set--.

Corrected drawing sheets in compliance with 37 CFR 1.121(d) are required in reply to the Office action to avoid abandonment of the application. Any amended replacement drawing sheet should include all of the figures appearing on the immediate prior version of the sheet, even if only one figure is being amended. The figure or figure number of an amended drawing should not be labeled as "amended." If a drawing figure is to be canceled, the appropriate figure must be removed from the replacement sheet, and where necessary, the remaining figures must be renumbered and appropriate changes made to the brief description of the several views of the drawings for consistency. Additional replacement sheets may be necessary to show the renumbering of the remaining figures. Each drawing sheet submitted after the filing date of an application must be labeled in the top margin as either "Replacement Sheet" or "New Sheet" pursuant to 37 CFR 1.121(d). If the changes are not accepted by the examiner, the applicant will be notified and informed of any required corrective action in the next Office action. The objection to the drawings will not be held in abeyance.

9. The drawings are objected to as failing to comply with 37 CFR 1.84(p)(5) because they include the following reference character(s) not mentioned in the description:

- In Fig.3, number 560 was not found in the specification.
- In Fig.5, number 325 was not found in the specification.

Corrected drawing sheets in compliance with 37 CFR 1.121(d), or amendment to the specification to add the reference character(s) in the description in compliance with 37 CFR

1.121(b) are required in reply to the Office action to avoid abandonment of the application. Any amended replacement drawing sheet should include all of the figures appearing on the immediate prior version of the sheet, even if only one figure is being amended. Each drawing sheet submitted after the filing date of an application must be labeled in the top margin as either "Replacement Sheet" or "New Sheet" pursuant to 37 CFR 1.121(d). If the changes are not accepted by the examiner, the applicant will be notified and informed of any required corrective action in the next Office action. The objection to the drawings will not be held in abeyance.

Claim Objections

10. Claim 26 is objected to because of the following informalities: In line 4, please insert --a-- after "in". Appropriate correction is required.

Claim Rejections - 35 USC § 112

11. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

12. Claims 14, 20, and 26 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Specifically, it is not clear what is meant by "using said result to reset itself to a shorter bit configuration". How doe a result reset itself? And, how are bits reduced? Also, applicant should not use the word "itself", as it is not completely clear what applicant is referring to.

Claim Rejections - 35 USC § 101

13. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

14. Claims 9-14 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. Specifically, applicant states in paragraph [0035] of the specification that one type of medium is a distribution medium in the form of electrical signals used to transmit data bits. Though applicant claims a storage medium, it is not clear which of the exemplary media set forth in the specification applicant intends to be a storage medium, especially when applicant states that "While such transmitted signals may be ephemeral in nature they still, nonetheless constitute a physical medium carrying the coded instruction bits and are intended for permanent capture at the signal's destination or destinations". That is, it is not clear if applicant also intends for the distribution medium to be a type of storage medium. Claims directed to electrical signals do not fall into one of the four statutory categories of invention. Therefore, until applicant clarifies, these claims are deemed non-statutory.

Claim Rejections - 35 USC § 102

15. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

16. Claims 9-12, 14-18, 20-24 and 26 are rejected under 35 U.S.C. 102(b) as being anticipated by Liu et al., U.S. Patent No. 6,049,864 (herein referred to as Liu).

17. Referring to claim 21, Liu has taught a method for emulating computer instructions from a source machine (see Fig.3 and claims 1 and 3 of Liu, and note that instruction ADDF32 X1 = X2 + X3 is from a CISC machine) to produce sequences of instructions on a target machine (see Fig.3 and column 3, lines 4-6, and note that a sequence of addf32 and prodf32 instructions are produced to emulate the CISC instruction), said method comprising: generating a sequence of target machine instructions which together operate to directly calculate target machine condition codes from carry, sign and overflow codes without including branch instructions from the target machine. See Fig.3, and note that the emulation of the ADDF32 source instruction results in the execution of multiple instructions, none of which are branches, and at least one of which sets carry, overflow, and sign codes.

18. Referring to claim 22, Liu has taught the method of claim 21 in which said calculation of target machine condition codes is carried out in a sequence of non-branching instructions that manipulate condition code settings produced within the target machine itself to produce the same condition code in a target machine location. See Fig.3.

19. Referring to claim 23, Liu has taught the method of claim 22 in which said manipulation employs temporary locations for storing intermediate results. See Fig.3, and note that in response to the addf32 instruction, not only is the result produced and stored in X1{31:0}, but other intermediate values are stored in X1{32:38}. These intermediate results are then used by the prodf32 instruction to set flags.

20. Referring to claim 24, Liu has taught the method of claim 21 further including the step of executing said generated sequence of target machine instructions. See Fig.3.

21. Referring to claim 26, Liu has taught the method of claim 21 in which said generating comprises the steps of:

a) mimicking an instruction present in a source instruction stream in a manner which sets one or more target CPU flag bits and which places a result in a first storage location. See Fig.3 and note that a source CISC instruction is mimicked with at least two RISC instructions that set target flag bits and which also place a result in a first storage location. For purposes of examination, the first storage location is interpreted as bits X1{37:0}. The result of the addition is placed in these bits.

b) executing an instruction which uses said result to set at least one bit in second storage location so as to distinguish one or more case results. See Fig.3 and note that in response to executing the addf32 instruction, the result of the addition is used to set bit X1{38}, which is herein interpreted as the second storage location. This bit distinguishes between carry and no-carry case results.

c) using said result to reset itself to a shorter bit configuration which serves to distinguish at least one of said case results. See Fig.3. Though the actual addition result is 32 bits (in X1{31:0}), bit X1{31} is used to indicate positive or negative, and consequently, the addition result is reset to 31 bits (X1{30:0}).

d) executing an instruction which employs said first storage location to produce a result which distinguishes a different set of case results. See Fig.3 and note that bits X1{31:30} are used is setting codes which distinguish a different set of case results.

e) executing an instruction which uses the results in said first and second storage locations to provide an indication in which at least three cases are distinguished, said cases representing condition codes of said source machine. See Fig.3, and note that prodf32 uses the results in the

first storage location (specifically bit X1{37}) and the second storage location (specifically bit X1{38}) to provide carry and overflow indications that distinguish between results of the type: (1) no carry, no overflow, (2) no carry, overflow, (3) carry, no overflow, and (4) carry, overflow.

22. Referring to claims 9-12 and 14, claims 9-12 and 14 are rejected for the same reasons set forth in the rejections of claims 21-24 and 26, respectively.

23. Referring to claims 15-18 and 20, the systems of claims 15-18 and 20 perform the methods of claims 21-24 and 26, respectively. Consequently, claims 15-18 and 20 are rejected for the same reasons set forth in the rejections of claims 21-24 and 26, respectively. Also, it should be noted that Liu has taught a memory, both inherently and explicitly. That is, in Fig.3, Liu shows a memory X1[38:0] for storing data. Also, program instructions are inherently stored on some medium so that the processor may fetch and process them.

Claim Rejections - 35 USC § 103

24. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

25. Claims 13, 19, and 25 are rejected under 35 U.S.C. 103(a) as being unpatentable over Liu in view of the examiner's taking of Official Notice.

26. Referring to claim 25, Liu has taught the method of claim 24. Liu has not explicitly taught that said executing occurs at substantially the same time as said generating. However, the examiner asserts that interpretive environments, in which instructions are translated at run-time,

are well known and accepted in the art. Such systems reduce the amount of work done by the compiler and therefore, reduce compilation time. As a result, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Liu such that the generating of the sequence of instructions occurs at run-time when the source instruction is encountered, thereby resulting in execution occurring at substantially the same time as said generating.

27. Referring to claim 13, claim 13 is rejected for the same reasons set forth in the rejection of claim 25.

28. Referring to claim 19, the system of claim 19 performs the method of claim 25. Consequently, claim 19 is rejected for the same reasons set forth in the rejection of claim 25.

Conclusion

29. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. Applicant is reminded that in amending in response to a rejection of claims, the patentable novelty must be clearly shown in view of the state of the art disclosed by the references cited and the objections made. Applicant must also show how the amendments avoid such references and objections. See 37 CFR § 1.111(c).

Scantlin, U.S. Patent No. 5,574,927, has taught condition code hardware which calculates condition codes in response to an arithmetic instruction.

Goettelmann et al., U.S. Patent No. 5,313,614, has taught expanding source code into intermediate code and from intermediate code to translated code. The translated code (Fig.11) shows setting a flag without using a branch (and instead with a load).

Knebel et al., U.S. Patent No. 6,807,625, has taught efficiently generating, storing, and consuming arithmetic flags when emulating macroinstructions with microinstructions.

Sandham et al., U.S. Patent No. 7,331,040, has taught condition code flag emulation for program code conversion.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to DAVID J. HUISMAN whose telephone number is (571)272-4168. The examiner can normally be reached on Monday-Friday (8:00-4:30).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/David J. Huisman/
Primary Examiner, Art Unit 2183
March 16, 2009

Notice of References Cited	Application/Control No. 11/668,623	Applicant(s)/Patent Under Reexamination COPELAND ET AL.	
	Examiner DAVID J. HUISMAN	Art Unit 2183	Page 1 of 1

U.S. PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Name	Classification
*	A	US-6,049,864	04-2000	Liu et al.	712/214
*	B	US-5,574,927	11-1996	Scantlin, Henry L.	712/41
*	C	US-5,313,614	05-1994	Goettelmann et al.	717/138
*	D	US-6,807,625	10-2004	Knebel et al.	712/221
*	E	US-7,331,040	02-2008	Sandham et al.	717/136
	F	US-			
	G	US-			
	H	US-			
	I	US-			
	J	US-			
	K	US-			
	L	US-			
	M	US-			

FOREIGN PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	Classification
	N					
	O					
	P					
	Q					
	R					
	S					
	T					

NON-PATENT DOCUMENTS

*		Include as applicable: Author, Title Date, Publisher, Edition or Volume, Pertinent Pages)
	U	
	V	
	W	
	X	

*A copy of this reference is not being furnished with this Office action. (See MPEP § 707.05(a).)
Dates in MM-YYYY format are publication dates. Classifications may be US or foreign.

LzLabs GmbH. Ex. 1004-238

Search Notes	Application/Control No.	Applicant(s)/Patent Under Reexamination
	11668623	COPELAND ET AL.
	Examiner	Art Unit
	DAVID J HUISMAN	2183

SEARCHED			
Class	Subclass	Date	Examiner
712	209, 221, 227	3-16-09	DJH
717	124, 134, 136, 137, 138	3-16-09	DJH
703	26	3-16-09	DJH

SEARCH NOTES		
Search Notes	Date	Examiner
EAST Search (all databases)	3/16/09	DJH
Inventor Name Search (PALM)	3/16/09	DJH

INTERFERENCE SEARCH			
Class	Subclass	Date	Examiner

	/DAVID J HUISMAN/ Primary Examiner.Art Unit 2183

Receipt date: 01/30/2007

INFORMATION DISCLOSURE CITATION

(*USE SEVERAL SHEETS IF NECESSARY*)

ATTY DOCKET NO.	SERIAL NO.
POU920060152US1	11/698623 - GAU: 2183
APPLICANT(S)	
COPELAND ET AL.	
FILING DATE	GROUP
HEREWITH	UNASSIGNED

U.S. PATENT DOCUMENTS

*EXAMINER INITIAL		DOCUMENT NUMBER	DATE	NAME	CLASS	SUBCLASS	FILING DATE IF APPROPRIATE
/DJH/	AA	2003/0093649 A1	05/15/2003	Hilton	712	41	

FOREIGN PATENT DOCUMENTS

		DOCUMENT NUMBER	DATE	COUNTRY	CLASS	SUBCLASS	TRANSLATION		
							NO	YES	ABSTRACT

OTHER DOCUMENTS (*including Author, Title, Date, Pertinent Pages, Etc.*)

EXAMINER	/David Huisman/	DATE CONSIDERED	10/09/2008

EXAMINER: Initial here if reference considered, whether or not citation is in conformance with MPEP 609; Draw line through citation if not in conformance and not considered. **Include copy of this form with next communication to applicant.**

LzLabs GmbH. Ex. 1004-240

PTO/SB/08a (10-08)
Approved for use through 11/30/2008. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	11668623
Filing Date	2007-01-30
First Named Inventor	Reid T. Copeland
Art Unit	2183
Examiner Name	
Attorney Docket Number	POU920060152US1

U.S.PATENTS

Examiner Initial*	Cite No	Patent Number	Kind Code[1]	Issue Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	5774694		1998-06-30	Murari Srinivasan	

If you wish to add additional U.S. Patent citation information please click the Add button.

U.S.PATENT APPLICATION PUBLICATIONS

Examiner Initial*	Cite No	Publication Number	Kind Code[1]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	20060173931		2006-08-03	James Broadhurst	

If you wish to add additional U.S. Published Application citation information please click the Add button.

FOREIGN PATENT DOCUMENTS

Examiner Initial*	Cite No	Foreign Document Number[3]	Country Code[2] i	Kind Code[4]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear	T[5]
	1	WO 97/50031	EP		1997-12-31	Intel Corp		
	2	EP 1455273	EP		2004-09-08	Northrop Grumman Corp.		

If you wish to add additional Foreign Patent Document citation information please click the Add button

NON-PATENT LITERATURE DOCUMENTS

DUPLICATE COPY - NOT CONSIDERED - DJH 3-16-09

LzLabs GmbH. Ex. 1004-241

Receipt date: 11/07/2008	Application Number	11668623	11668623 - GAU: 2183
INFORMATION DISCLOSURE STATEMENT BY APPLICANT	Filing Date	2007-01-30	
(Not for submission under 37 CFR 1.99)	First Named Inventor	Reid T. Copeland	
	Art Unit	2183	
	Examiner Name		
	Attorney Docket Number	POU920060152US1	

Examiner Initials*	Cite No	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc), date, pages(s), volume-issue number(s), publisher, city and/or country where published.	T[5]
	1	~~International Search Report PCT/EP2008/050725 POU920060152 4/10/2008~~	

If you wish to add additional non-patent literature document citation information please click the Add button

EXAMINER SIGNATURE

Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through a citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [2] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [3] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [4] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [5] Applicant is to place a check mark here if English language translation is attached.

DUPLICATE COPY - NOT CONSIDERED - DJH 3-16-09

LzLabs GmbH. Ex. 1004-242

Doc code :IDS
Doc description: Information Disclosure Statement (IDS) Filed

PTO/SB/08a (10-08)
Approved for use through 11/30/2008. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT
(Not for submission under 37 CFR 1.99)

Application Number	11668623
Filing Date	2007-01-30
First Named Inventor	Reid T. Copeland
Art Unit	2183
Examiner Name	
Attorney Docket Number	POU920060152US1

U.S.PATENTS

Examiner Initial*	Cite No	Patent Number	Kind Code[1]	Issue Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
/DJH/	1	5774694		1998-06-30	Murari Srinivasan	

If you wish to add additional U.S. Patent citation information please click the Add button.

U.S.PATENT APPLICATION PUBLICATIONS

Examiner Initial*	Cite No	Publication Number	Kind Code[1]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
/DJH/	1	20060173931		2006-08-03	James Broadhurst	

If you wish to add additional U.S. Published Application citation information please click the Add button.

FOREIGN PATENT DOCUMENTS

Examiner Initial*	Cite No	Foreign Document Number[3]	Country Code[2] i	Kind Code[4]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear	T[5]
/DJH/	1	WO 97/50031	EP		1997-12-31	Intel Corp		☐
/DJH/	2	EP 1455273	EP		2004-09-08	Northrop Grumman Corp.		☐

If you wish to add additional Foreign Patent Document citation information please click the Add button

NON-PATENT LITERATURE DOCUMENTS

LzLabs GmbH. Ex. 1004-243

Receipt date: 11/07/2008	Application Number	11668623	11668623 - GAU: 2183
INFORMATION DISCLOSURE STATEMENT BY APPLICANT	Filing Date	2007-01-30	
(Not for submission under 37 CFR 1.99)	First Named Inventor	Reid T. Copeland	
	Art Unit	2183	
	Examiner Name		
	Attorney Docket Number	POU920060152US1	

Examiner Initials*	Cite No	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc), date, pages(s), volume-issue number(s), publisher, city and/or country where published.	T[5]
/DJH/	1	International Search Report PCT/EP2008/050725 POU060152 4/18/2008	☐

If you wish to add additional non-patent literature document citation information please click the Add button

EXAMINER SIGNATURE

Examiner Signature	/David Huisman/	Date Considered	03/16/2009

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through a citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [2] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [3] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [4] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [5] Applicant is to place a check mark here if English language translation is attached.

LzLabs GmbH. Ex. 1004-244

Receipt date: 11/07/2008	Application Number	11668623	11668623 - GAU: 2183
INFORMATION DISCLOSURE STATEMENT BY APPLICANT (**Not for submission under 37 CFR 1.99)**	Filing Date	2007-01-30	
	First Named Inventor	Reid T. Copeland	
	Art Unit	2183	
	Examiner Name		
	Attorney Docket Number	POU920060152US1	

CERTIFICATION STATEMENT

Please see 37 CFR 1.97 and 1.98 to make the appropriate selection(s):

☐ That each item of information contained in the information disclosure statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(1).

OR

☐ That no item of information contained in the information disclosure statement was cited in a communication from a foreign patent office in a counterpart foreign application, and, to the knowledge of the person signing the certification after making reasonable inquiry, no item of information contained in the information disclosure statement was known to any individual designated in 37 CFR 1.56(c) more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(2).

☐ See attached certification statement.

☐ Fee set forth in 37 CFR 1.17 (p) has been submitted herewith.

☒ None

SIGNATURE

A signature of the applicant or representative is required in accordance with CFR 1.33, 10.18. Please see CFR 1.4(d) for the form of the signature.

Signature	/John E. Campbell/	Date (YYYY-MM-DD)	2008-11-07
Name/Print	John E. Campbell	Registration Number	52,687

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 1 hour to complete, including gathering, preparing and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

LzLabs GmbH. Ex. 1004-245

EAST Search History

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L2	1	"20030093649".pn.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
L3	86	emulat$3 with "condition code"	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
L4	102	emulat$3 with ("condition code" "flag bit" "status bit") and (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
L5	29	emulat$3 with ("condition code" "flag bit" "status bit") same branch and (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
L6	68	("20020026633" | "4514803" | "4812975" | "5313614" | "5329471" | "5408637" | "5440702" | "5551015" | "5574873" | "5574927" | "5577233" | "5598560" | "5606696" | "5649203" | "5724590" | "5737631" | "5751982" | "5768593" | "5805792" | "5805895" | "5870575" | "5890000" | "5901308" | "5903760" | "5925124" | "5930509" | "5946491" | "5956495" | "5966539" | "5983335" | "6000028" | "6009261" | "6031994" |	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-246

		"6076155" | "6077314" | "6091897" | "6105124" | "6126329" | "6170083" | "6185675" | "6199095" | "6205545" | "6223339" | "6301705" | "6308318" | "6321379" | "6330691" | "6360194" | "6367067" | "6427234" | "6434741" | "6463582" | "6477641" | "6502237" | "6505296" | "6522985" | "6523173" | "6526572" | "6530078" | "6530079" | "6535903" | "6539471" | "6539541" | "6549987" | "6564297" | "6594824" | "6631514" | "6643630").PN.				
L7	6	(emulat$3 translat$3) same ("condition code" "flag bit" "status bit") same (without avoid $4) near2 branch$3 and (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
L8	144	(emulat$3 translat$3) with (without avoid$4) near2 branch$3 and (overflow carry sign)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
L9	12	(emulat$3 translat$3) with (without avoid$4) near2 branch$3 and (overflow carry sign) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-247

L10	5	("4951195" | "5241678" | "5301325" | "5313614" | "5598560").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
L11	28	("5903760").URPN.	USPAT	OR	ON	2009/03/16 16:40
L12	68	("20020026633" | "4514803" | "4812975" | "5313614" | "5329471" | "5408637" | "5440702" | "5551015" | "5574873" | "5574927" | "5577233" | "5598560" | "5606696" | "5649203" | "5724590" | "5737631" | "5751982" | "5768593" | "5805792" | "5805895" | "5870575" | "5890000" | "5901308" | "5903760" | "5925124" | "5930509" | "5946491" | "5956495" | "5966539" | "5983335" | "6000028" | "6009261" | "6031994" | "6076155" | "6077314" | "6091897" | "6105124" | "6126329" | "6170083" | "6185675" | "6199095" | "6205545" | "6223339" | "6301705" | "6308318" | "6321379" |	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-248

		"6330691" | "6360194" | "6367067" | "6427234" | "6434741" | "6463582" | "6477641" | "6502237" | "6505296" | "6522985" | "6523173" | "6526572" | "6530078" | "6530079" | "6535903" | "6539471" | "6539541" | "6549987" | "6564297" | "6594824" | "6631514" | "6643630").PN.				
L13	68	("20020026633" | "4514803" | "4812975" | "5313614" | "5329471" | "5408637" | "5440702" | "5551015" | "5574873" | "5574927" | "5577233" | "5598560" | "5606696" | "5649203" | "5724590" | "5737631" | "5751982" | "5768593" | "5805792" | "5805895" | "5870575" | "5890000" | "5901308" | "5903760" | "5925124" | "5930509" | "5946491" | "5956495" | "5966539" | "5983335" | "6000028" | "6009261" |	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-249

		"6031994" | "6076155" | "6077314" | "6091897" | "6105124" | "6126329" | "6170083" | "6185675" | "6199095" | "6205545" | "6223339" | "6301705" | "6308318" | "6321379" | "6330691" | "6360194" | "6367067" | "6427234" | "6434741" | "6463582" | "6477641" | "6502237" | "6505296" | "6522985" | "6523173" | "6526572" | "6530078" | "6530079" | "6535903" | "6539471" | "6539541" | "6549987" | "6564297" | "6594824" | "6631514" | "6643630").PN.				
L14	723	(emulat$3 translat$3) with compar$4 with instruction	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
L15	21	(emulat$3 translat$3) with compar$4 with instruction same (code flag) near2 (condition status bit)	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
L16	412	712/209.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
L17	487	712/221.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
L18	867	712/227.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-250

L19	897	717/136.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
L20	265	717/137.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
L21	234	717/138.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
L22	1	emulat$3 with ("condition code" "flag bit" "status bit") and (overflow carry sign)	USOCR; FPRS; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2009/03/16 16:40
L23	1362	717/124.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
L24	140	717/134.ccls.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
L25	1560	(emulat$3 translat$3) same (arithmetic add $5 subtract$3) same (code flag) near2 (condition status bit)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
L26	101	(emulat$3 translat$3) same (arithmetic add $5 subtract$3) with set $4 with (code flag) near2 (condition status bit)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
L27	37	(emulat$3 translat$3) same (arithmetic add $5 subtract$3) with set $4 with (code flag) near2 (condition status bit) same (Z O S C sign zero overflow carry)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
L28	29	(emulat$3 translat$3) same (arithmetic add $5 subtract$3) with set $4 with (code flag) near2 (condition status bit) same (Z O S C sign zero overflow carry) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
L29	7	("20020046298" "20040176941" "20050015754" "20070011664" "5301302" "5678032" "5966515").pn.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-251

L30	4	L29 and (code flag) near2 (condition status bit)	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
L31	119	simulat$3 near4 (code flag) near2 (condition status bit)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
L32	67	simulat$3 near4 (code flag) near2 (condition status bit) and g06f$. ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
L33	35	simulat$3 near4 (code flag) near2 (condition status bit) and (overflow carry) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
L34	127	(emulat$3 translat$3) and (host target) with set$4 with (code flag) near2 (condition status bit) and ((overflow O) and (carry C) and (sign S negative N)) and g06f$.ipc.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
L35	59	(emulat$3 translat$3) and (host target) with set$4 with (code flag) near2 (condition status bit) and ((overflow O) and (carry C) and (sign S negative N)) and ("712"/$.ccls. "717"/$.ccls.)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
L36	32	emulat$3 and (host target) with set$4 with (code flag) near2 (condition status bit) and ((overflow O) and (carry C) and (sign S negative N)) and ("712"/$.ccls. "717"/$. ccls.)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
L37	1	emulat$3 and (host target) with set$4 with (code flag) near2 (condition status bit) same ((overflow O) and (carry C) and (sign S negative N)) and ("712"/$.ccls. "717"/$.ccls.)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40

LzLabs GmbH. Ex. 1004-252

L38	1	emulat$3 and (host target) with set$4 with (code flag) near2 (condition status bit) same (((overflow O) and (carry C) and (sign S negative N))) and ("712"/$.ccls. "717"/$.ccls.)	US-PGPUB; USPAT	OR	ON	2009/03/16 16:40
L39	6	("4415969" | "5327537" | "5440749" | "5493687" | "5590359" | "5689720").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
L40	2	("5774694").URPN.	USPAT	OR	ON	2009/03/16 16:40
L41	8	("5313614" | "5481693" | "5619664" | "5774694" | "5859999" | "5860017" | "5884059" | "6049864").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2009/03/16 16:40
L42	1	"20060173931".pn.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:44
L43	446	703/26.ccls.	US-PGPUB; USPAT	OR	ON	2009/03/16 16:46

3/16/09 4:50:37 PM
C:\Documents and Settings\DHuisman\My Documents\EAST\Workspaces\11-668623.wsp

LzLabs GmbH. Ex. 1004-253

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants:	Copeland et al.	Confirmation No.:	4180
Serial No.:	11/668,623	Group Art Unit:	2183
Filed:	01/30/2007	Examiner:	Unassigned
Title:	EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS (as amended)		

CERTIFICATE OF ELECTRONIC TRANSMISSION

I hereby certify that this correspondence is being transmitted electronically to: United States Patent and Trademark Office, Alexandria, VA 22313-1450, on February 20, 2009.

Jill K. Becker

Date of Signature: February 20, 2009

To: Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Preliminary Amendment

Dear Sir:

Prior to examination of the above-referenced application, please amend the application as follows:

Amendments to the Specification begin on page 2 of this paper.

Amendments to the Claims begin on page 3 of this paper.

Remarks begin on page 7 of this paper.

LzLabs GmbH. Ex. 1004-254

Amendments to the Specification

Kindly amend the title on pages 1 and 19 of the specification, as follows:

~~METHOD FOR~~ EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS

LzLabs GmbH. Ex. 1004-255

Amendment to the Claims

Kindly cancel claims 1-8, without prejudice, and add new claims 9-26, as set forth below. The changes in the amended claims are shown by strikethrough (for deleted matter) and underlining (for added matter).

1-8. (Canceled)

9. (New) A computer program product for emulating computer instructions from a source machine to produce sequences of instructions on a target machine, said computer program product comprising:

a storage medium readable by a processor and storing instructions for execution by the processor for performing a method comprising:

generating a sequence of target machine instructions which together operate to directly calculate target machine condition codes from carry, sign and overflow codes without including branch instructions from the target machine.

10. (New) The computer program product of claim 9, wherein said calculation of target machine condition codes is carried out in a sequence of non-branching instructions that manipulate condition code settings produced within the target machine itself to produce the same condition code in a target machine location.

11. (New) The computer program product of claim 10, wherein said manipulation employs temporary locations for storing intermediate results.

12. (New) The computer program product of claim 9, further including executing said generated sequence of target machine instructions.

13. (New) The computer program product of claim 12, wherein said executing occurs at substantially the same time as said generating.

LzLabs GmbH. Ex. 1004-256

14. (New) The computer program product of claim 9, wherein said generating comprises:

mimicking an instruction present in a source instruction stream in a manner which sets one or more target CPU flag bits and which places a result in a first storage location;

executing an instruction which uses said result to set at least one bit in a second storage location so as to distinguish one or more case results;

using said result to reset itself to a shorter bit configuration which serves to distinguish at least one of said case results;

executing an instruction which employs said first storage location to produce a result which distinguishes a different set of case results; and

executing an instruction which uses the results in said first and second storage locations to provide an indication in which at least three cases are distinguished, said cases representing condition codes of said source machine.

15. (New) A computer system for emulating computer instructions from a source machine to produce sequences of instructions on a target machine, the computer system comprising:

a memory; and

a processor in communications with the memory, wherein the computer system is capable of performing a method, the method comprising:

generating a sequence of target machine instructions which together operate to directly calculate target machine condition codes from carry, sign and overflow codes without including branch instructions from the target machine.

16. (New) The computer system of claim 15, wherein said calculation of target machine condition codes is carried out in a sequence of non-branching instructions that

LzLabs GmbH. Ex. 1004-257

manipulate condition code settings produced within the target machine itself to produce the same condition code in a target machine location.

17. (New) The computer system of claim 16, wherein said manipulation employs temporary locations for storing intermediate results.

18. (New) The computer system of claim 15, further including executing said generated sequence of target machine instructions.

19. (New) The computer system of claim 18, wherein said executing occurs at substantially the same time as said generating.

20. (New) The computer system of claim 15, wherein said generating comprises:

mimicking an instruction present in a source instruction stream in a manner which sets one or more target CPU flag bits and which places a result in a first storage location;

executing an instruction which uses said result to set at least one bit in a second storage location so as to distinguish one or more case results;

using said result to reset itself to a shorter bit configuration which serves to distinguish at least one of said case results;

executing an instruction which employs said first storage location to produce a result which distinguishes a different set of case results; and

executing an instruction which uses the results in said first and second storage locations to provide an indication in which at least three cases are distinguished, said cases representing condition codes of said source machine.

21. (New) A method for emulating computer instructions from a source machine to produce sequences of instructions on a target machine, said method comprising:

generating a sequence of target machine instructions which together operate to directly calculate target machine condition codes from carry, sign and overflow codes without including branch instructions from the target machine.

LzLabs GmbH. Ex. 1004-258

22. (New) The method of claim 21, wherein said calculation of target machine condition codes is carried out in a sequence of non-branching instructions that manipulate condition code settings produced within the target machine itself to produce the same condition code in a target machine location.

23. (New) The method of claim 22, wherein said manipulation employs temporary locations for storing intermediate results.

24. (New) The method of claim 21, further including executing said generated sequence of target machine instructions.

25. (New) The method of claim 24, wherein said executing occurs at substantially the same time as said generating.

26. (New) The method of claim 21, wherein said generating comprises:

mimicking an instruction present in a source instruction stream in a manner which sets one or more target CPU flag bits and which places a result in a first storage location;

executing an instruction which uses said result to set at least one bit in second storage location so as to distinguish one or more case results;

using said result to reset itself to a shorter bit configuration which serves to distinguish at least one of said case results;

executing an instruction which employs said first storage location to produce a result which distinguishes a different set of case results; and

executing an instruction which uses the results in said first and second storage locations to provide an indication in which at least three cases are distinguished, said cases representing condition codes of said source machine.

LzLabs GmbH. Ex. 1004-259

Remarks

Entry of the amendments, consideration of the application, as amended, and allowance of all pending claims are respectfully requested. After entry of the amendments, claims 9-26 are pending.

Support for the amendments can be found in the claims as originally filed and throughout applicants' specification (see, e.g., paragraphs 11, 32 and 35). Thus, no new matter is added.

Should the Examiner have any questions regarding this application, please call applicants' attorney at the below listed number.

Respectfully submitted,

Blanche E. Schiller

Blanche E. Schiller
Attorney for Applicants
Registration No.: 35,670

Dated: February 20, 2009

HESLIN ROTHENBERG FARLEY & MESITI P.C.
5 Columbia Circle
Albany, New York 12203-5160
Telephone: (518) 452-5600
Facsimile: (518) 452-5579

LzLabs GmbH. Ex. 1004-260

Electronic Acknowledgement Receipt

EFS ID:	4832063
Application Number:	11668623
International Application Number:	
Confirmation Number:	4180
Title of Invention:	METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS
First Named Inventor/Applicant Name:	Reid T. Copeland
Customer Number:	46369
Filer:	Blanche E. Schiller
Filer Authorized By:	
Attorney Docket Number:	POU920060152US1
Receipt Date:	20-FEB-2009
Filing Date:	30-JAN-2007
Time Stamp:	16:40:05
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1		POU920060152US1_PreliminaryAmendment.pdf	298616 6fdb895ba48e8cd4c215e62fe3712160c82eed73	yes	8

	Multipart Description/PDF files in .zip description		
	Document Description	**Start**	**End**
	Miscellaneous Incoming Letter	1	1
	Preliminary Amendment	2	2
	Specification	3	3
	Claims	4	7
	Applicant Arguments/Remarks Made in an Amendment	8	8
Warnings:			
Information:			
	Total Files Size (in bytes):	298616	

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office
If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

AMENDMENT TRANSMITTAL LETTER (Large Entity) Applicant(s): **Copeland et al.**	Docket No. **POU920060152US1**

Application No.	Filing Date	Examiner	Customer No.	Group Art Unit	Confirmation No.
11/668,623	**01/30/2007**	**Unassigned**	**46369**	**2183**	**4180**

Invention: **EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS (as amended)**

COMMISSIONER FOR PATENTS:

Transmitted herewith is an amendment in the above-identified application.

The fee has been calculated and is transmitted as shown below.

CLAIMS AS AMENDED

	CLAIMS REMAINING AFTER AMENDMENT		HIGHEST # PREV. PAID FOR		NUMBER EXTRA CLAIMS PRESENT	RATE	ADDITIONAL FEE
TOTAL CLAIMS	18	-	20	=	0	x $52.00	$0.00
INDEP. CLAIMS	3	-	3	=	0	x $220.00	$0.00
Multiple Dependent Claims (check if applicable) ☐							$0.00
TOTAL ADDITIONAL FEE FOR THIS AMENDMENT							$0.00

- ☒ No additional fee is required for amendment.
- ☐ Please charge Deposit Account No. in the amount of
- ☐ A check in the amount of to cover the filing fee is enclosed.
- ☒ The Director is hereby authorized to charge payment of the following fees associated with this communication or credit any overpayment to Deposit Account **09-0463 (IBM)**
  - ☒ Any additional filing fees required under 37 C.F.R. 1.16.
  - ☒ Any patent application processing fees under 37 CFR 1.17.
- ☐ Payment by credit card. Form PTO-2038.

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

Blanche E. Schiller
Signature

Dated: **February 20, 2009**

Blanche E. Schiller, Esq.
Reg. No. 35,670
Heslin Rothenberg Farley & Mesiti P.C.
5 Columbia Circle
Albany, NY 12203
Telephone: (518) 452-5600
Facsimile: (518) 452-5579

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to "Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450" [37 CFR 1.8(a)] on

_____ .
(Date)

Signature of Person Mailing Correspondence

Typed or Printed Name of Person Mailing Correspondence

cc:

P11LARGE/REV10

TRANSMITTAL OF INFORMATION DISCLOSURE STATEMENT (Under 37 CFR 1.97(d))	Docket No. POU920060152US1

In Re Application Of: **Reid T. Copeland et. al.**

Application No.	Filing Date	Examiner	Customer No.	Group Art Unit	Confirmation No.
11/668,623	**1/30/2007**		**33558**	**2183**	**4180**

Title: **Method For Efficiently Emulating Computer Architecture Condition Code Settings**

Address to:
Commissioner for Patents

The Information Disclosure Statement submitted herewith is being filed after the period specified in 37 CFR 1.97(c), and on or before payment of the issue fee, and is accompanied by the Statement as specified in 37 CFR 1.97(e) and the fee set forth in 37 CFR 1.17(p).

☐ A check in the amount of is attached.

☒ The Director is hereby authorized to charge and credit Deposit Account No. **090463** as described below.

- ☒ Charge the amount of **$180.00**
- ☒ Credit any overpayment.
- ☒ Charge any additional fee required.

☐ Payment by credit card. Form PTO-2038 is attached.

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

Certificate of Transmission by Facsimile*

I certify that this document and authorization to charge deposit account is being facsimile transmitted to the United States Patent and Trademark Office (Fax no.) on

_____
(Date)

Signature

Typed or Printed Name of Person Signing Certificate

***This certificate may only be used if paying by deposit account.**

Certificate of Mailing by First Class Mail

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to "Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450" [37 CFR 1.8(a)] on

_____.
(Date)

Signature of Person Mailing Correspondence

Typed or Printed Name of Person Mailing Certificate

/John E. Campbell/
Signature

Dated: **November 7, 2008**

John E. Campbell, Reg. 52,687
IBM Corporation
IP Law Department
2455 South Road, MS P386
Poughkeepsie, NY 12601
(845) 433-1156

cc:

P10B/REV05

PTO/SB/08a (10-08)
Approved for use through 11/30/2008. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Not for submission under 37 CFR 1.99)

Application Number	11668623
Filing Date	2007-01-30
First Named Inventor	Reid T. Copeland
Art Unit	2183
Examiner Name	
Attorney Docket Number	POU920060152US1

U.S.PATENTS

Examiner Initial*	Cite No	Patent Number	Kind Code[1]	Issue Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	5774694		1998-06-30	Murari Srinivasan	

If you wish to add additional U.S. Patent citation information please click the Add button.

U.S.PATENT APPLICATION PUBLICATIONS

Examiner Initial*	Cite No	Publication Number	Kind Code[1]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	20060173931		2006-08-03	James Broadhurst	

If you wish to add additional U.S. Published Application citation information please click the Add button.

FOREIGN PATENT DOCUMENTS

Examiner Initial*	Cite No	Foreign Document Number[3]	Country Code[2] i	Kind Code[4]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear	T[5]
	1	WO 97/50031	EP		1997-12-31	Intel Corp		☐
	2	EP 1455273	EP		2004-09-08	Northrop Grumman Corp.		☐

If you wish to add additional Foreign Patent Document citation information please click the Add button

NON-PATENT LITERATURE DOCUMENTS

LzLabs GmbH. Ex. 1004-265

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	11668623
	Filing Date	2007-01-30
	First Named Inventor	Reid T. Copeland
	Art Unit	2183
	Examiner Name	
	Attorney Docket Number	POU920060152US1

Examiner Initials*	Cite No	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc), date, pages(s), volume-issue number(s), publisher, city and/or country where published.	T[5]
	1	International Search Report PCT/EP2008/050725 POU060152 4/18/2008	☐

If you wish to add additional non-patent literature document citation information please click the Add button

EXAMINER SIGNATURE

Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through a citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [2] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [3] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [4] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [5] Applicant is to place a check mark here if English language translation is attached.

LzLabs GmbH. Ex. 1004-266

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	11668623
	Filing Date	2007-01-30
	First Named Inventor	Reid T. Copeland
	Art Unit	2183
	Examiner Name	
	Attorney Docket Number	POU920060152US1

CERTIFICATION STATEMENT

Please see 37 CFR 1.97 and 1.98 to make the appropriate selection(s):

☐ That each item of information contained in the information disclosure statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(1).

OR

☐ That no item of information contained in the information disclosure statement was cited in a communication from a foreign patent office in a counterpart foreign application, and, to the knowledge of the person signing the certification after making reasonable inquiry, no item of information contained in the information disclosure statement was known to any individual designated in 37 CFR 1.56(c) more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(2).

☐ See attached certification statement.

☐ Fee set forth in 37 CFR 1.17 (p) has been submitted herewith.

☒ None

SIGNATURE

A signature of the applicant or representative is required in accordance with CFR 1.33, 10.18. Please see CFR 1.4(d) for the form of the signature.

Signature	/John E. Campbell/	Date (YYYY-MM-DD)	2008-11-07
Name/Print	John E. Campbell	Registration Number	52,687

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 1 hour to complete, including gathering, preparing and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

LzLabs GmbH. Ex. 1004-267

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 1 455 273 A2

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
08.09.2004 Bulletin 2004/37

(51) Int Cl.[7]: **G06F 9/455**

(21) Application number: **04002677.5**

(22) Date of filing: **01.03.2004**

(84) Designated Contracting States:
AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HU IE IT LI LU MC NL PL PT RO SE SI SK TR
Designated Extension States:
AL LT LV MK

(30) Priority: **06.03.2003 US 382598**

(71) Applicant: **Northrop Grumman Corporation**
Los Angeles, CA 90067-2199 (US)

(72) Inventors:
- **Cannon, William J.**
  Centerville, Ohio 45459 (US)
- **Zwirner, Eric W.**
  Dayton Ohio 45405 (US)
- **Hoerig, Timothy R.**
  Beavercreek Ohio 45431 (US)
- **Ward, Paul D.**
  Centerville Ohio 45458 (US)

(74) Representative: **Lieck, Hans-Peter, Dipl.-Ing.**
Boeters & Lieck
Widenmayerstrasse 36
80538 München (DE)

(54) Emulation of microprocessor instructions

(57) A method and system (20) for emulating instructions of legacy microprocessors which execute a compiled high-ordered language (28), such as C/C++, in which the compiled code is structured such that data and instructions segments (22; 24) are segregated. In order to improve the real-time performance of the system, legacy instructions (32) are directly mapped to equivalent instructions (34) of the host processor where possible. Additional techniques may optionally be employed to further increase the real-time performance of the system. By utilizing the direct mapping of the legacy instructions to host instructions, the emulation system (20) in accordance with the present invention provides increased real-time performance for relatively modern RISC microprocessors.

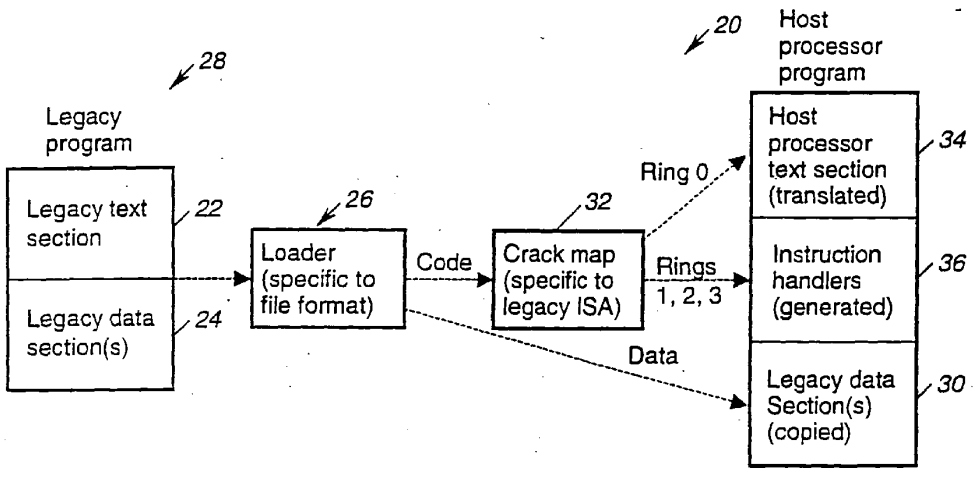


Figure 1

EP 1 455 273 A2

LzLabs GmbH. Ex. 1004-268

Description

[0001] The present invention relates to a method and apparatus for emulating instructions of a legacy microprocessor and, more particularly, to an emulation system for a legacy microprocessor, such as a 32-bit microprocessor, which executes a compiled higher-order language, such as C/C++, in which the compiled code is structured such that data and instruction segments are structured, for example, as in Reduced Instruction Set Computer (RISC) microprocessors, wherein the legacy instructions are directly mapped to an equivalent instruction of the host processor, where possible, to improve the real-time performance of the system.

[0002] It is known that microprocessors are configured with different instruction-set architectures (ISA). The ISA determines the instruction set for a particular microprocessor. Application programs are executed by a microprocessor, normally written in a relatively high-level language, which, in turn, are compiled into machine instructions compatible with the instruction set for a specific microprocessor.

[0003] Due to the age obsolescence of many existing microprocessors and corresponding slow speed, it is often desired to replace outdated existing microprocessors, hereinafter referred to as legacy microprocessors, with newer microprocessors, hereinafter referred to as host microprocessors. Unfortunately, depending on the particular upgrade, most times the instruction set of the host microprocessor is incompatible with the instruction set of the legacy microprocessor.

[0004] One way to solve this problem is to rewrite the existing application program taking into account the instruction set of the host microprocessor. Such a technique is relatively cumbersome and quite expensive. In order to work around instruction set incompatibilities, software systems are known for emulating instructions of legacy microprocessors. Such systems are often referred to as emulation systems. Such emulation systems are known to emulate the instruction set of the legacy microprocessor in order to enable the application programs written in terms of the instruction set for the legacy microprocessor to be executed by a newer, faster host microprocessor.

[0005] In many known emulation systems, legacy instructions are fetched by the host microprocessor, which, in turn, uses a look-up table to interpret the legacy instruction. With such emulation systems, since each legacy instruction must be interpreted, systems which incorporate cache memory are known to suffer from relatively high probability of cache misses, which decreases the overall throughput of the system.

[0006] In order to improve the throughput or speed of such emulation systems, U.S. Patent No. 6,041,402 discloses a direct vectored legacy instruction set emulation system. Rather than fetching legacy instructions and interpreting the instructions in software, the direct vectored legacy instruction set emulation system fetches direct vectors to software routines which emulate the legacy instructions. By fetching the direct vectors, the need for look-up tables for interpreting the legacy instructions is eliminated. Since look-up table accesses are eliminated with such a system, the probability of cache misses is greatly reduced, which, in turn, increases the throughput of the system. Unfortunately, the direct vectored legacy instruction set emulation requires too much performance overhead for relatively modern microprocessors, such as 32-bit RISC microprocessors. Thus, there is a need for an emulation system and method for emulating instructions of legacy microprocessors which provides improved real-time performance relative to known emulation systems.

[0007] Briefly, the present invention relates to a method and system for emulating instructions of legacy microprocessors which execute a compiled high-order language, such as C/C++, in which the compiled code is structured such that data and instructions segments are segregated, for example, as in reduced instruction set computer (RISC) microprocessors. In order to improve the real-time performance of the system, legacy instructions are directly mapped to equivalent instructions of the host processor, where possible. Additional techniques may optionally be employed to further increase the real-time performance of the system. When exact matches cannot be made, an unconditional branch instruction to an instruction handler is used. The instruction handler is made up of two or more host processor instructions that ultimately perform the same function as the replaced legacy instruction. For legacy instructions which cannot be emulated solely with host instructions, these instructions are emulated by instruction handlers which include several host instructions and which incorporate the use of scratch pad registers. For legacy instructions that involve architectural features of a legacy microprocessor that cannot be handled easily by the host processor's assembly language, an instruction handler is formed from a high level language, such as, C/C++ code. By utilizing the direct mapping of the legacy instructions to host instructions and one or more of the above-mentioned emulation techniques, the emulation system in accordance with the present invention provides increased real-time performance for relatively modern RISC microprocessors.

[0008] These and other advantages of the present invention will be readily understood with reference to the following specification and attached drawing wherein:

[0009] FIG. 1 is a high-level block diagram of the emulation system in accordance with the present invention.

[0010] FIG. 2 is a software flow diagram of an instruction cracking process used in accordance with the present invention.

[0011] FIG. 3 is a diagram of a portion of memory illustrating memory mapping in accordance with one aspect of the

LzLabs GmbH. Ex. 1004-269

invention.

[0012] FIG. 4 illustrates an exemplary example of direct mapping of the op code of an exemplary legacy ADDU (add unsigned) instruction to a host ADD instruction.

[0013] FIG. 5 is similar to FIG. 4 but illustrates the mapping of an exemplary legacy LHU (load half word unsigned) instruction to a host instruction.

[0014] FIG. 6 is similar to FIG. 4 and illustrates the mapping of an exemplary legacy SRA (shift right algebraic) instruction to a host instruction.

[0015] FIG. 7 is similar to FIG. 4 and illustrates the mapping of an exemplary legacy SYNC instruction to a host instruction.

[0016] FIG. 8 illustrates an exemplary host instruction handler for an exemplary legacy LB (load byte) instruction.

[0017] FIG. 9 is similar to FIG. 8 but for an exemplary legacy BEQ (branch if equal) instruction.

[0018] FIG. 10 illustrates an instruction stream which includes a number of direct mapped instructions, for example, as illustrated in FIGS. 4 through 7, illustrating unconditional branches to instruction handlers for legacy instructions which cannot be directly mapped to a host instruction.

[0019] FIG. 11 is an example of an emulation of an exemplary legacy instruction fully emulated by a plurality of host instructions which requires a scratch-pad register for a MIPS SLT (set if less than) instruction to a PowerPC handler.

[0020] FIG. 12 is similar to FIG. 11, except that it illustrates the emulation of an exemplary MTC1 (move to co-processor 1) instruction.

[0021] FIG. 13A is an example of an emulation of a legacy Ring 3 SYSCALL instruction that requires emulation by instruction handler, formed from higher-level instructions.

[0022] FIG. 13B is similar to FIG. 13A but for a Ring 3 CFCz legacy instruction.

[0023] FIG. 14 illustrates an instruction stream containing various legacy instructions and the corresponding handler memory space.

[0024] FIG. 15 illustrates execution of the emulation program in accordance with the present invention.

[0025] FIGS. 16-20 illustrate the various instruction formats for a MIPS type RISC microprocessor.

DETAILED DESCRIPTION

[0026] The present invention relates to a method and apparatus for emulating instructions of a particular category legacy microprocessors which execute a compiled higher-order language, such as C/C++, Fortran, or ADA. Such compiled code is typically structured such that data and instruction segments are segregated. The compiled code also conforms to other platform specific Application Binary Interfaces (ABI) which describe things like procedure calling and linkage conventions and register usage conventions.

[0027] The present invention relates to the emulation of a 32-bit RISC microprocessor using another 32-bit RISC microprocessor. As used herein, host processor denotes the processor that executes the emulation program. The term legacy processor is used to identify the processor being emulated.

[0028] The emulation technique in accordance with the present invention assumes similarities between the host and legacy processors to enable the legacy instructions to be categorized into one or more of the following four categories:

- Ring 0 - These legacy instructions can be replaced with a single host processor instruction that performs the same function. Modifying the op code of the instruction and rearranging the instruction field is generally all that is necessary for this category.
- Ring 1 - This type of legacy instruction can be replaced with two or more host processor instructions that perform the same function and do not require scratch-pad registers to perform the operation.
- Ring 2 - These legacy instructions can be replaced with two or more host processor instructions that perform the same function but require one or more scratch-pad registers to perform the operation.
- Ring 3 - These legacy instructions generally relate to advanced architectural features of the legacy microprocessor and further relate to instructions that are far too complex to be implemented utilizing only the host processor assembly language. These instructions are handled almost entirely in a high-level language, such as C/C++.

[0029] In order to enable the host processor to execute programs written for the legacy processor, the legacy processor's binary instructions must be rendered into the host processor's binary instructions in one of the four categories mentioned above. There are many operations performed by all microprocessors, such as simple arithmetic (i.e., add, subtract, multiply), logical operations (i.e., logical and, or, not), and data moving (i.e., load and store). Many of these instructions are very similar and identical between different brands and/or models of RISC microprocessors and tend to appear frequently throughout many application programs. Thus, by taking advantage of the similarities of the instruction sets between the host microprocessor and the legacy microprocessor and utilizing direct mapping as set forth in the Ring 0 category, wherever possible, the real-time performance of the system will be greatly improved. When

LzLabs GmbH. Ex. 1004-270

exact matches cannot be made, the legacy instruction is replaced with an unconditional branch instruction to a legacy instruction handler made up of two or more host processor instructions that ultimately perform the same function as the replaced legacy instruction, for example, for legacy instructions categorized as Ring 1 and Ring 2. The use of unconditional branches helps retain performance by taking advantage of the host microprocessor branch prediction capabilities. For advanced instructions that involve advanced features of the legacy microprocessor or cannot be easily handled with the host processor assembly language, Ring 3 category instructions are branched out to a higher-level language, such as C/C++, to handle the operation at a higher level.

[0030] In order to retain the performance advantage of Ring 0, Ring 1, and Ring 2 and because the register usage conventions differ between the legacy and host microprocessors, the legacy registers are re-mapped to the corresponding functional equivalent host register. For example, the stack pointer for a PowerPC host microprocessor is r1, while the stack pointer for a MIPS R4400 microprocessor is r29. Known emulator systems are known to keep legacy registers in memory, which require memory accesses and additional host instructions to manipulate a legacy register value, which substantially decreased the real-time performance. By re-mapping the legacy registers to the host processor registers, the real-time performance is enhanced relative to known emulation systems.

[0031] The invention is described in terms of a 32-bit R4400 RISC microprocessor by MIPS Technologies, Inc. (http://www.mips.com) as a legacy processor and a PowerPC host processor (http://www.motorola.com/powerpc). However, as mentioned above, the principles of the present invention are applicable to any compatible microprocessors in which some of the legacy processor instructions can be directly mapped to a host processor instruction. It is known that the number of 1:1 matches between instructions can be relatively high for compatible host/legacy processor combinations.

SYSTEM OVERVIEW

[0032] A high-level block diagram of the emulation system and method in accordance with the present invention is illustrated in FIG. 1 and generally identified with the reference numeral 20. As mentioned above, the emulation system and method in accordance with the present invention is amenable for use with legacy microprocessors in which the instructions and data are segregated, as indicated by the boxes 22 and 24. A file loader 26 locates the legacy program, generally identified with the reference numeral 28. File loaders are well known with the art. The file loader 26 reads the legacy file format and fetches the legacy binary files. As shown in FIG. 1, the file loader 26 fetches both the legacy instructions 22 as well as the legacy data files 24. These legacy instruction and data files 22 and 24 may be contained on a disk, a read-only memory, or fetched over a network. If the legacy data 24 is static, it is simply copied to a legacy data memory space 30 associated with the host microprocessor program. Alternatively, the legacy data may be created and initialized for data sections that are created dynamically as the program runs.

[0033] The legacy instructions 22 are provided by the file loader 26 to a crack map object 32. The crack map object 32 is specific to the legacy and host instruction set architectures, as will be discussed in more detail below. As will also be discussed in more detail below, the crack map object 32 analyzes each legacy instruction 22 and categorizes the instruction according to the four exemplary categories discussed above (i.e., Ring 0, Ring 1, Ring 2 and Ring 3). As shown in FIG. 1, Ring 1-3 instructions are placed in separate handler memory space from the Ring 0 memory space. The Ring 0 instructions are directly mapped by the crack map object 32 to the host processor memory space 34. All other instructions are mapped to the instruction handlers 36.

CRACK MAP OBJECT

[0034] The crack map object 32 converts each legacy binary instruction received from the file loader 26 into one or more host processor instructions. As mentioned above, the crack map object 32 described and illustrated herein is for emulating instructions of the legacy MIPS R4400 RISC microprocessor by way of a PowerPC G3/G4 processor.

[0035] Referring to FIG. 2, the crack map object 32 fetches and reads each legacy instruction 22 in steps 40 and 42. Each legacy instruction is then decoded based on its op-code in steps 44 and 46. More particularly, with reference to Tables 1 and 2 below, each legacy instruction is listed in a look-up table along with its legacy op-code along with other data fields, as described below. These look up tables, as will be discussed in more detail below, include the Ring category for each legacy instruction 22. Accordingly, the crack map object 32 checks in step 48 to determine whether the instruction falls into the Ring 0 category. If so, the crack map object 32 directly maps the legacy instruction 22 to a host instruction by translating the op-codes, as will be discussed in more detail below, in step 50. The translated host instruction is then written to the application program instruction stream in step 52.

[0036] If the legacy instruction 22 is not a Ring 0 category, an unconditional branch instruction to the next handler in the instruction stream is inserted in the instruction stream in step 54. The system then checks in steps 56, 58 and 60 to determine whether the legacy instruction falls into a Ring 1 category, Ring 2 category and Ring 3 category.

[0037] If the legacy instruction 22 falls into a Ring 1 category, as determined in 56, instructions to the instruction handler's 36 (FIG. 1) are written in step 58B and a return branch instruction is inserted at the end of the instruction

LzLabs GmbH. Ex. 1004-271

handler 36 in step 60A. Should the instruction be determined to be a Ring 2 category instruction, as determined in step 58A, the crack map object 32 inserts code to save data from the scratch pad registers and subsequently writes instructions to the instruction handler space 36 (FIG. 1) in step 64.

[0038] As mentioned above, Ring 2 category instructions require a plurality of host processor instructions along with scratch pad registers in order to provide an equivalent function to the legacy instruction. Once the instruction is emulated, the scratch pad registers associated with the Ring 2 instructions are restored in step 66. The system then returns to step 60A and inserts a branch at the end of the handler to return to the instruction stream.

[0039] If it is determined that the instruction is a Ring 3 category instruction, instruction code is inserted in the instruction handler 36 to save the machine state in step 60B. Subsequently in step 68, the instruction handler jumps to a higher order language program, for example C/C++, which provides an equivalent function to the legacy instruction. The higher order language by way of a compiler handles the return back to the instruction handler 36 which returns to step 70 where the machine state is restored and the system returns to the main instruction screen by inserting a return branch instruction at the end of the handler.

INSTRUCTION SPACING AND BRANCH FOLDING

[0040] As shown best in FIGS. 14 and 15, the host processor program is shown as an instruction stream, generally identified with the reference number 72, which includes Ring 0 type instructions and unconditional branches to instruction handlers 36 (FIG. 1). This configuration of placing Ring category 1-3 instructions in separate handlers space 36 from the Ring 0 space 34 and using the unconditional branches to branch to the Ring 1-3 instruction handlers space 36 (FIG. 1) is used to maintain distance between the instructions. More particularly, every program is anticipated to contain jump and/or branch instructions. If the spacing between instructions is not maintained, jumps and branches may end up at the wrong location, since many instructions operate in a relative fashion, jumping forward or backward a given distance from the current location rather than using an absolute destination address. Thus, when a 1:1 match cannot be made, a single unconditional branch is used in place of the legacy instruction as shown in FIGS. 14 and 15. The unconditional branch instruction vectors to the instruction handler space 36 (FIG. 1) to emulate the legacy instruction. When the handler completes its function, it returns to the instruction following the unconditional branch, as shown thus in FIGS. 14 and 15.

[0041] FIG. 3 illustrates an example of an conditional branch forward based upon relative addressing. More particularly, during the execution of the instruction at address 1004, the program counter points to the next instruction at address 1008. As shown in FIG. 3, the instruction BEQ is a conditional branch instruction. The conditional branch instruction, if taken, increments the current address by eight (8) to arrive at a destination address of 1010. If either of the instructions at 1008 or 100C cannot be translated 1:1 and if additional code is inserted in line, a branch would vector to an incorrect destination address. If instead an unconditional branch is used to branch to an instruction handler, for example, an unconditional branch inserted into the instruction stream 72 (FIGS. 14 and 15), the distance remains the same.

[0042] This technique relies on a feature of many RISC processors called "branch folding" which allows the unconditional branches to be automatically removed from the instruction stream by the processor with little or no loss of performance. When a branch is unconditional, it must be taken. The processor simply begins pre-fetching instructions down the new path, and little or no performance penalty is associated with the branch. On the other hand conditional branches can impose significant performance penalties because instruction pre-fetching can't occur until it is known which path to take.

REGISTER MAPPING

[0043] In accordance with one aspect of the present invention, register mapping is used to prevent conflicts between the host processor and the legacy processor. For example on a MIPS R4400 microprocessor, general register 29 is the stack pointer. However on a Power PC microprocessor, the stack pointer is register 1. In order to prevent conflicts, register fields in legacy instructions are remapped to the host processor registers. In an example where legacy instructions of a MIPS R4400 microprocessor instructions are mapped to a Power PC instructions, register 29 of the legacy instructions is remapped to register 1 of the host processor and register 1 of the legacy instructions is remapped to register 29 of the host processor. This may be done by treating the legacy register numbers as an index into an array which contains the host processor number to be used for the given legacy register number.

INSTRUCTION RENDERING

[0044] As mentioned above, legacy instructions are categorized into one of four categories: Ring 0, Ring 1, Ring 2 or Ring 3. As discussed above, the object code of the Ring 0 legacy instructions are directly mapped or converted to

LzLabs GmbH. Ex. 1004-272

a single host instruction which performs the same function. Examples of selected rendering of legacy instructions are described and illustrated herein. Mapping of Ring 0 instructions is illustrated in FIGS. 4-7. Examples of mapping of Ring 1 instructions are illustrated in FIG. 8 and 9. Examples of a Ring 2 instruction mapping are illustrated in FIGS. 11 and 12 while an example of a Ring 3 instruction mapping is illustrated in FIG. 13.

RING 0 INSTRUCTIONS

[0045] Ring 0 instructions are legacy processor instructions that can be replaced with a single host processor instruction. Binary reorganization (i.e. translating of the object code) of the legacy instructions is necessary most of the time even though the corresponding host processor instruction performs the same operation as the legacy processor instruction.

[0046] FIGS. 4-7 illustrate examples of conversions of several 32-bit MIPS R4400 RISC processor instructions to 32-bit Power PC RISC microprocessor equivalent instructions. In this example, the MIPS processor is the legacy processor while the Power PC is the host processor.

[0047] For Ring 0 instructions, the operation code (op code) is changed to match the appropriate instruction in the host processor. In some cases, a sub op code field must also be translated. Registers and immediate value fields are moved as necessary. In some cases, the host processor bit fields are set to an inactive state. Though not shown in the figures, legacy register mapping may also be done depending on the legacy/host processor combination.

[0048] FIG. 4 illustrates an example of a MIPS ADDU (add unsigned instruction) in its simplest binary representation. To convert the ADDU legacy instruction to an equivalent host ADD instruction, the op code and various sub op code fields as well as register fields are changed as illustrated in FIG. 4. The host instruction also has 2 additional single bit option fields (i.e., bits 0 and 10) that are not used for the legacy instruction. These fields are simply cleared.

[0049] Referring to FIG. 4, the ADDU legacy instruction (adds the values in the registers rs and rt and stores the results in a register rd). As shown, the legacy sub op code is 00000100001, which corresponds to bits 0-10 of the legacy instruction. The binary value 0000010001 corresponds to a decimal 33. The corresponding host instruction is an Add instruction.

[0050] In this case, the host instruction adds values in its registers rA and rB stores the results in a register rT. The format of the host processor instruction is slightly different from the format for the corresponding legacy instruction. In addition, the host instruction includes an RC (record condition) bit and an OE (overflow enable) bit as bits 0 and 10 of the 32 bit instruction. These options are not available for the legacy MIPS processor. As discussed above, these bits are simply zeroed out. After zeroing out bits 0 and 10, the equivalent sub op code of the host Add instruction is 100001010. Thus, in this case, the legacy op code 00000100001 is translated to 100001010 and, as mentioned above, and the registers are remapped. In particular, the legacy register rd is mapped to the host register rT while the legacy register rs is mapped to the host register rA and finally the legacy register rt is mapped to the host register rB. In addition, bits 26-31 of the legacy instruction are translated to the appropriate op code for the host instruction. As shown, the op code at bits 26-31, shown as 000000 is translated 011111.

[0051] FIGS. 5 and 6 illustrate other examples of direct mapping. For example, FIG. 5 illustrates direct mapping of a legacy LHU (load halfword unsigned) instruction and an SRA (shift right algebraic) instruction. In the case of the LHU instruction, immediate data is also transferred to the host processor instruction. The SRA instruction contains an immediate 5 bit shift count field. Finally FIG. 7 illustrates a legacy Sync instruction which has no argument. The corresponding Sync instruction similarly has no arguments. In this case the op code and sub op code are translated as illustrated in FIG. 7.

RING 1 INSTRUCTIONS

[0052] As mentioned above, Ring 1 category legacy instructions cannot be mapped directly to a single host processor instruction. Therefore, an unconditional branch instruction is inserted into the instruction stream 72 (FIGS. 14, 15) as discussed above. The branch vectors execution to an instruction handler that uses multiple host processor instructions to emulate the behavior of the emulated legacy instruction. Ring 1 instructions allow for a simple enough emulation that additional scratch registers are not required.

[0053] FIG. 8 illustrates an assembly language mnemonic for a legacy LB (load byte) instruction and a host instruction handler for the LB legacy instruction. The legacy LB instruction is categorized as a Ring 1 instruction because the legacy instruction also performs a sign extension which extends the sign bit in the destination register based on the sign of the byte loaded. An additional host processor instruction is required to perform the sign extension. More particularly, as illustrated in FIG. 8, five words are required to handle this instruction. The first word is the exact legacy binary instruction. The next word is a pointer to another handler for the same instruction with different operands. This is done to allow reusing a handler that has already been created. The singly linked list provides a mechanism for searching for a match. The third word starts the host processor handler of the legacy instruction. Any number of host

LzLabs GmbH. Ex. 1004-273

processor instructions may be present. Finally, a host processor sub routine return instruction is inserted to return to the main instruction stream.

[0054] FIG. 9 illustrates a legacy BEQ (branch if equal instruction) and a host instruction handler for emulating this instruction. In this case the BEQ legacy instruction is categorized as a Ring 1 instruction because the legacy BEQ instruction performs a comparison and branch all in one instruction. The host processor instructions do not allow for comparison branching in one instruction. Rather, the instructions for the host processor require that the compare must be done separately and the results stored in a condition code register. Then the conditional branch uses the stored condition code when deciding whether to branch.

[0055] In the case of the legacy target, the host handler may also incorporate the legacy instruction immediately following the branch of equal instruction since the legacy processor may execute it while calculating the destination address. The handler can accommodate this as well, although not shown in the figure. More particularly with reference to FIG. 10, a Ring 1 is shown in an instruction stream with several Ring 0 instructions. Whenever a Ring 1 instruction is executed, an unconditional branch and link instruction is used to vector execution to an instruction handler starting at the instruction following the two word header. As shown in FIG. 10, at the end of the instruction handler, execution returns to the instruction stream and specifically to the link address immediately following the branch.

RING 2 INSTRUCTION

[0056] Ring 2 instructions are similar to Ring 1 instructions except that one or more scratch registers are needed to perform the operation. This requires saving out one or more of the host processor registers and then restoring them before returning to the legacy instruction screen.

[0057] FIG. 11 is an example of a Ring 2 instruction. In particular, FIG. 11 illustrates a legacy SLT (set if less than instruction) and a host instruction handler for handling this instruction. The legacy instruction performs a compare and stores the zero for inequality and a zero for equality into a designated register. In this case the host processor, a Power PC processor, does not have a corresponding instruction. Thus, the host processor instead performs a compare and then extracts the results of the compare from a condition code register. That value must then be rotated into the correct position. These operations require a scratch register to work so one is allocated, saved out and restored when the handler is complete.

[0058] FIG. 12 illustrates another example of a legacy MTC1 instruction along with its corresponding host processor instruction handler. The MTC1 instruction is a move to co-processor 1 instruction which moves a value from a general purpose register into a floating-point register which is the legacy processor-coprocessor 1. In the case of the host microprocessor (i.e., PowerPC microprocessor), there are no instructions to move values directly from a general register to a floating-point register. Therefore, the instruction handler must use a scratch register to construct the address of a memory buffer. The general register value is stored in memory and then a floating-point load instruction is used to get it into the floating-point register as generally shown in FIG. 12.

[0059] It is anticipated that a Ring 2 instruction can be avoided if a host processor with adequate register resources is selected. In the case of a MIPS R4400 legacy microprocessor and a PowerPC host processor, each has a 32-bit general purpose register and 32-bit floating-point registers. This forces the requirement to save the registers through memory and restore them at one or more scratch registers are required.

RING 3 INSTRUCTIONS

[0060] Ring 3 instructions involve complex operations and advanced architectural features of the legacy processor that are not easily emulated using host processors or assembly language. Therefore, Ring 3 instructions are handled using a higher-order language, such as C/C++ instead. A host assembly language handler is set up for these instructions but its function is to call the higher-order language components as required.

[0061] FIG. 13A illustrates a MIPS SYSCALL (system call) instruction and its host handler. As shown, the machine's state is saved prior to making the call to the higher-level language and is returned and restored on return from the handler. The requirements of calling a function in a higher-order language are well documented by the application binary interface (ABI) document of the host processor. Such ABIs contain conventions that must be followed to ensure proper interaction of programs for any given architecture. The specific requirements. therefore, are dictated by the choice of the host processor. The handler for the Ring 3 instruction, then, is whatever code is necessary to call the higher-order language component.

[0062] FIG. 13B illustrates exemplary PowerPC assembly language and an exemplary handler for a Ring 3 CFCz instruction. The instruction handler is generated in PowerPC machine code. However, commented assembly is used for purposes of illustration.

[0063] The legacy instruction being emulated is a CFCz instruction, which moves a 32-bit value from a legacy co-processor register to a general-purpose register. The legacy microprocessor, in this case a MIPS RISC R4400 micro-

LzLabs GmbH. Ex. 1004-274

processor, allows for up to 4 coprocessors with coprocessor 1 being the floating-point unit. The instruction handler saves out the machine state, fetches the legacy instruction from the top of the handler and then extracts the three arguments consisting of the coprocessor register number, the general-purpose register number, and the coprocessor number.

```
// Save the machine state.  Registers get saved on the stack.
stwu     sp,-160(sp)     // adjust the stack pointer
mflr     r0              // get current link register
stw      r0,4(sp)        // save it on the stack
stw      r0,32(sp)       // save all PowerPC general registers to the stack
stw      r1,36(sp)       // starting at offset 32
.
stw      r30,152(sp)
stw      r31,156(sp)

// Setup to call the C/C++ handler code.  For CFCz, this involves fetching
// the legacy instruction from the top of the handler (second word, just
// before any executable code) and extracting the arguments from it.

// fetch the legacy instruction from the top of the handler so the arguments
// can be retrieved
bl       +4              // do a linking branch to get the program counter
nop                      //  into the link register (can't read it directly)
mflr     r13             // fetch the program counter from the link register
lwz      r13,-152(r13)   // load the legacy CFCz instruction

// get the first argument the contents of MIPS register rd
rlwinm   r14,r13,23,25,29 // extract the rd field and multiply by 2
addi     r14,r14,32      // form offset into stack for rd
lwzx     r3,r14,sp       // load the contents of rd from the stack

// get the second argument the z field
rlwinm   r4,r13,5,30,31  // extract the coprocessor number (0)

// Call the handler function.  The rest will be handled in C/C++ code.
bl    Execute_CFCz       // Execute_CFCz called as if it were a C function

// the return value comes back in r3 store it into the appropriate
// register on the stack in the next stage it will be placed into the
// appropriate register
rlwinm   r15,r13,18,25,29 // extract the rt field from the legacy instruction
addi     r15,r15,32      // form offset into stack for rd
stwx     r3,r15,sp       // store return value into appropriate register

// Restore the machine state.  Here all of the volatile registers are
// reloaded and the stack pointer is adjusted back.
lwz      r0,32(sp)       // restore non-volatile registers
lwz      r1,36(sp)
.
lwz      r30,152(sp)
lwz      r31,156(sp)
lwz      r0,4(sp)         // load the original link register
mtlr     r0               //  and restore it
addi     sp,sp,160        // restore the original stack pointer

blr                       // return to the instruction stream
```

[0064] In the set up section, the arguments are placed into the host, i.e. PowerPC, registers r3 and r4. This complies with the PowerPC application binary interface specification for calling a C function. The call to Execute_CFCz, is a designated C function, used for handling the CFCz instruction. An example for that function follows.

LzLabs GmbH. Ex. 1004-275

```
#include <CoprocessorControlManager.h>

// there is a pointer to the C++ class that manages the interaction with
// up to 4 MIPS coprocessors
extern CoprocessorControlManager *cp;


// we must give this function C style language naming so that it can be
// called from assembly/machine language code
extern ``C''
{
unsigned long Execute__CFC (unsigned long rd, unsigned long z);
};

// this C function calls the C++ object that manages coprocessor
// operations: for the CTCz instruction, only a register read or write
// is required but the consequences of a read or write are dependent on
// the nature of the coprocessor
unsigned long Execute__CFC (unsigned long rd, // passed in PowerPC register r3
                            unsigned long z)  // passed in PowerPC register r4
{
return cp->readReg (z, rd);
}
```

[0065] The following is an exemplary class specification for a C++ class that handles coprocessor operations.

```
// this is an example class specification for a coprocessor control
class CoprocessorControlManager
{
   public:

   // the public constructor initializes the object as necessary
   CoprocessorControlManager ();

   // the destructor performs cleanup when the class is destroyed
   ~CoprocessorControlManager ();

   // this function reads a coprocessor register and returns it value
   void readReg (unsigned long z, unsigned long rd);
};
```

[0066] The following is an exemplary implementation for the coprocessor read register function defined in the class above.

```
// this is an exemplary implementation of the method responsible for
// performing reads from coprocessor control registers
unsigned long CoprocessorControlManager ::readReg (unsigned long z,
                                                   unsigned long rd)
{
  // which coprocessor?
  switch (z)
  {
     // the floating point unit
     case 0:
     {
```


LzLabs GmbH. Ex. 1004-276

```
            // which register to read
            switch rd:
               // this is the coprocessor implementation and revision register which
               // contains information about the floating point unit (it is read only)
               case 0:
                  // return the version number
                  return 1;

               // this is the floating point control and status register
               case 1:
                  // rounding, trap enables, and flags are stored here
                  // CODE OMITTED
                  break;

               // all other regisers are reserved for this coprocessor so just
               // return 0;
              default:
                  return 0;
            } // end case

           // no other cases handled by default
           default:
             return 0;
         } // end switch

        // CODE OMITTED
    }
```

LOOK-UP TABLES

[0067] As mentioned above, Tables 1 and 2 represent the look-up tables used by the crack map object 32 for identifying legacy instructions and identifying the specific ring category for each of the legacy instructions for legacy instruction of a MIPS R4400 RISC legacy microprocessor.

TABLE 1
CLASSIFICATION OF MIPS 32 BIT INTEGER INSTRUCTIONS

Instruction	Opcode	Function	Type	Ring	RS	RT	RD	Immediate	Shamt
ADD	0	32	R	1					0
ADDI	8		I	0					
ADDIU	9		I	0					
ADDU	0	33	R	0					0
AND	0	36	R	0					0
ANDI	12		I	0					
BC0F	16		I	3	8	0			
BC0FL	16		I	3	8	2			
BC0T	16		I	3	8	1			
BC0TL	16		I	3	8	3			
BC1F	17		I	3	8	0			
BC1FL	17		I	3	8	2			
BC1T	17		I	3	8	1			

LzLabs GmbH. Ex. 1004-277

Instruction	Opcode	Function	Type	Ring	RS		RT	RD	Immediate	Shamt
BC1TL	17		I	3	8		3			0
BC2F	18		I	3	8		0			
BC2FL	18		I	3	8		2			
BC2T	18		I	3	8		1			0
BC2TL	18		I	3	8		3			
BEQ	8		I	2						
BEQL	20		I	2						
BGEZ	1		I	2			1			
BGEZAL	1		I	2			17			
BGEZALL	1		I	2			19			
BGEZL	1		I	2			3			
BGTZ	7		I	2			0			
BGTZL	23		I	2			0			
BLEZ	6		I	2			0			
BLEZL	22		I	2			0			
BLTZ	1		I	2			0			
BLTZAL	1		I	2			16			
BLTZALL	1		I	2			18			
BLTZL	1		I	2			2			
BNE	5		I	2						
BNEL	21		I	2						
BREAK	0	13	I	0						
CACHE	47		I	3						
CFC0	16	0	R	0		2				0
CFC1	17	0	R	3		2				0
CFC2	18	0	R	3		2				0
COP0	16		J	3					upper bit set	0
COP1	17		J	3					upper bit set	0
COP2	18		J	3					upper bit set	
CTC0	16	0	R	0		6				0
CTC1	17	0	R	3		6				
CTC2	18	0	R	3		6				0
CTC3	19	0	R	3		6				
DIV	0	26	R	2				0		0
DIVU	0	27	R	2				0		0
ERET	16	24	R	3		16	0	0		0
J	2		J	0						
JAL	3		J	2						
JALR	0	9	R	2			0			0
JR	0	8	R	1			0	0		0
LB	32		I	0	base				offset	
LBU	36		I	0	base				offset	
LH	33		I	0	base				offset	
LHU	37		I	0	base				offset	
LL	48		I	3	base				offset	
LUI	15		I	0		0				

LzLabs GmbH. Ex. 1004-278

Instruction	Opcode	Function	Type	Ring	RS	RT	RD	Immediate	Shamt
LW	35		I	0	base			offset	
LWC1	49		I	3	base			offset	
LWC2	50		I	3	base			offset	
LWL	34		I	0	base			offset	
LWR	38		I	0	base			offset	
LWU	39		I	0	base			offset	
MFC1	17	0	R	3	0				0
MFC2	18	0	R	3	0				
MFHI	0	16	R	1	0	0			0
MFLO	0	18	R	1	0	0			0
MTC1	17	0	R	3	4				
MTC2	18	0	R	3	4				
MTHI	0	17	R	2		0	0		0
MTLO	0	19	R	2		0	0		0
MULT	0	24	R	2			0		0
MULTU	0	25	R	3			0		0
NOR	0	39	R	0					0
OR	0	37	R	0					0
ORI	13		I	0					
SB	40		I	0					
SC	56		I	3	base			offset	
SH	41		I	0	base			offset	
SLL	0	0	R	0	0				
SLLV	0	4	R	0					0
SLT	0	42	R	2					0
SLTI	10		I	2					
SLTIU	11		I	2					
SLTU	0	43	R	2					0
SRA	0	3	R	0	0				
SRAV	0	7	R	0					0
SRL	0	2	R	0	0				
SRLV	0	6	R	0					0
SUB	0	34	R	1					0
SUBU	0	35	R	0					0
SW	43		I	0	base			offset	
SWC1	57		I	3	base			offset	
SWC2	58		I	3	base			offset	
SWL	42		I	0	base			offset	
SWR	46		I	0	base			offset	
SYNC	0	15	R	0	0	0	0		0
SYSCALL	0	12	R	3	code	code	code		code
TEQ	0	52	R	3			code		code
TEQI	1		I	3		12			
TGE	0	48	R	3					
TGEI	1		I	3		8			
TGEIU	1		I	3		9			

LzLabs GmbH. Ex. 1004-279

Instruction	Opcode	Function	Type	Ring	RS	RT	RD	Immediate	Shamt
TGEU	0	49	R	3			code		code
TLBP	16	8	R	3	16	0	0		0
TLBR	16	1	R	3	16	0	0		0
TLBWI	16	2	R	3	16	0	0		0
TLBWR	16	6	R	3	16	0	0		0
TLT	0	50	R	3			code		code
TLTI	1		I	3		10			
TLTIU	1		I	3		11			
TLTU	0	51	R	3			code		code
TNE	0	54	R	3			code		code
TNEI	1		I	3		14			
XOR	0	38	R	0					0
XORI	14		I	0					

TABLE 2
CLASSIFICATION OF MIPS 32 BIT FLOATING POINT INSTRUCTIONS

Instruction	Opcode	Function	Type	Ring	FT/Base	FS	FD	Offset	Format
ABS.fmt	17	5	R	0	0				
ADD.fmt	17	0	R	0					
C.cond.fmt	17	FC/cond	R	0			0		
CEIL.L.fmt	17	10	R	2	0				
CEIL.W.fmt	17	14	R	2	0				
CVT.D.fmt	17	33	R	2	0				
CVT.L.fmt	17	37	R	2	0				
CVT.S.fmt	17	32	R	2	0				
CVT.W.fmt	17	36	R	2	0				
DIV.fmt	17	3	R	0					
DMFC1	17	0	R	3			0		1
FLOOR.L.fmt	17	11	R	2	0				
FLOOR.W.fmt	17	15	R	2	0				
MOV.fmt	17	6	R	0	0				
MUL.fmt	17	2	R	0					
NEG.fmt	17	7	R	0	0				
ROUND.L.fmt	17	8	R	0	0				
ROUND.W.fmt	17	12	R	0	0				
SQRT.fmt	17	4	R	2	0				
TRUNC.L.fmt	17	9	R	0	0				
TRUNC.W.fmt	17	13	R	0	0				

[0068] As shown, Table 1 includes an exemplary ten fields. The first field is ider tified as an instruction field and identifies the assembly language "mnemonic" for each of the legacy instructions. The op-code field represents the op-code of the legacy instruction. The function field is used for register type instructions for both integer instruction format and a floating point instruction format instructions. For example, FIGS. 16-20 illustrate the integer and floating point binary instruction formats for a MIPS microprocessor. For register type instructions, the function field is used as a sub-op code and translates to a function field for the host processor as described, for example, in connection with FIG. 4.

LzLabs GmbH. Ex. 1004-280

example, bits 0 through 10 (See FIG. 4.). Thus, as shown, for the ADDU instruction, the decimal value of sub-op-code (i.e., bits 0 through 10) is 33. This value 33, in the case of the ADDU instruction is thus identified in the function field. The type field relates to the instruction type, for example, as illustrated in FIGS. 16-20. Tables 1 and 2 also include a ring field. The ring field corresponds to the Ring category as discussed above. The RS, RT and RD fields relate to the legacy registers and identify the register numbers associated with the registers rs, rt and rd. The immediate field is used for immediate type instructions, for example, as illustrated in FIG. 16. Finally, the "Shamt" field relates to the shift amount for example for legacy instructions as illustrated in FIG. 17.

[0069] Table 2 lists the 32 bit floating instructions for a MIPS R4400 RISC microprocessor. The "instruction," "op code," "function," "type" "ring" fields are the same as Table 1. The "FT/base" and "offset" fields are used for immediate type floating point instructions, for example, as shown in FIG. 19. The FT, FS and FD fields refer to register numbers for floating print instructions.

[0070] The format field indicates the size and format of the floating point value. Table 3 illustrates valid format field codes for designating single and double precision IEEE 754 floating point values as well as single and long word fixed point values. Other values of the format field are possible but used for decoding instructions only.

Format Code (fmt Field)	Mnemonic	Size	Format
16	S	Single	Binary floating-point
17	D	Double	Binary floating-point
18	-	-	-
19	-	-	-
20	W	Single	32-bit Binary Fixed-Point
21	L	Longword	64-bit Binary Fixed-Point
22-31	-	-	-

Claims

1. A method for emulating instructions of a legacy microprocessor with a host microprocessor comprising the steps of:

   (a) converting the object code of one or more legacy instructions to the object code of one or more host processor instructions with equivalent functions defining Ring 0 instructions;
   (b) replacing legacy instructions with said Ring 0 instructions.

2. The method as recited in claim 1,
   further including the step of re-mapping registers associated with said legacy instructions to corresponding host registers.

3. The method as recited in claim 1 or 2,
   further including the step of utilizing multiple host instructions which together perform the same function as a single legacy instruction defining Ring 1 instructions to replace one or more of said legacy instructions.

4. The method as recited in claim 3,
   further including the step of utilizing multiple host instructions which together perform the same function as a single scratch pad register but require the use of one or more scratch pad registers defining Ring 2 instructions to replace one or more of said legacy instructions.

5. The method as recited in claim 4,
   further including the step of emulating legacy instructions which can only be emulated with a high level language defining a Ring 3 instruction.

6. The method as recited in claim 3, 4 or 5,
   further including providing an instruction handler for said Ring 1 instructions.

LzLabs GmbH. Ex. 1004-281

7. The method as recited in claim 4, 5 or 6,
further including providing an instruction handler for said Ring 2 instructions.

8. The method as recited in claim 5, 6 or 7,
further including providing an instruction handler for said Ring 3 instructions.

9. The method as recited in claim 6, 7 or 8,
wherein said instruction handlers are mapped to separate address space from said Ring 0 instructions.

10. The method as recited in claim 9,
wherein said emulated instructions form an instruction stream containing Ring 0 instructions and unconditional branch instructions to instruction handlers for said Ring 1, Ring 2 and Ring 3 instructions.

LzLabs GmbH. Ex. 1004-282

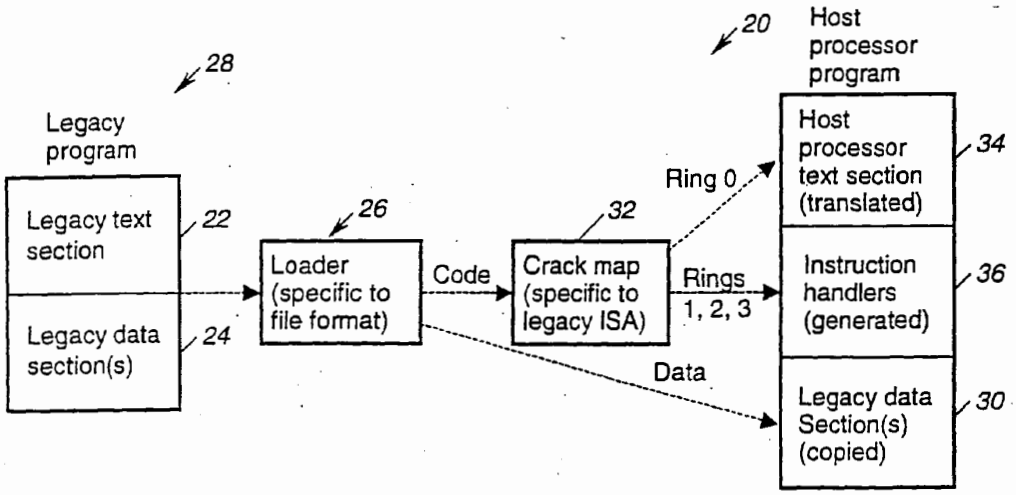


Figure 1

LzLabs GmbH. Ex. 1004-283

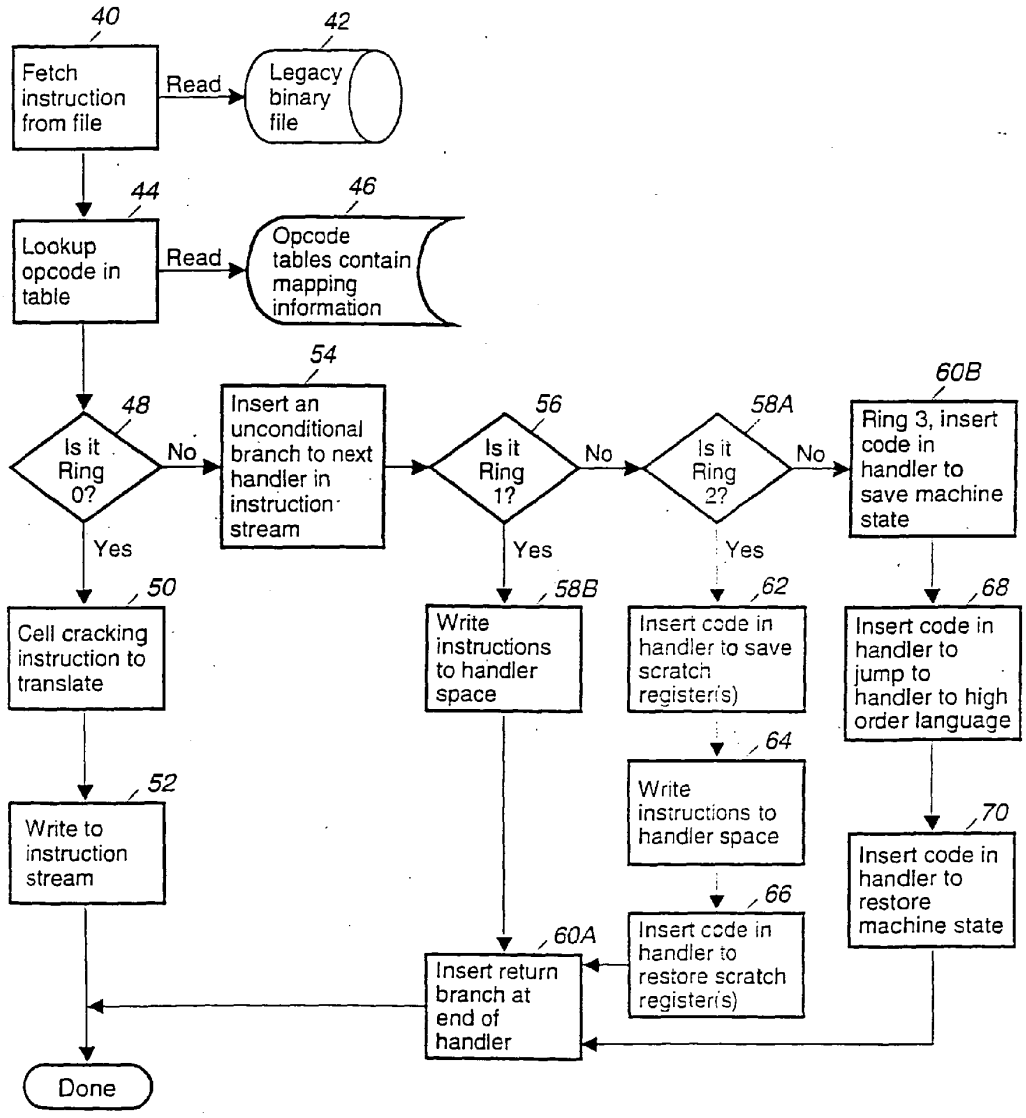


Figure 2

LzLabs GmbH. Ex. 1004-284

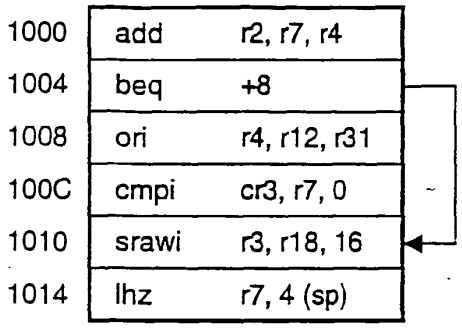


Figure 3

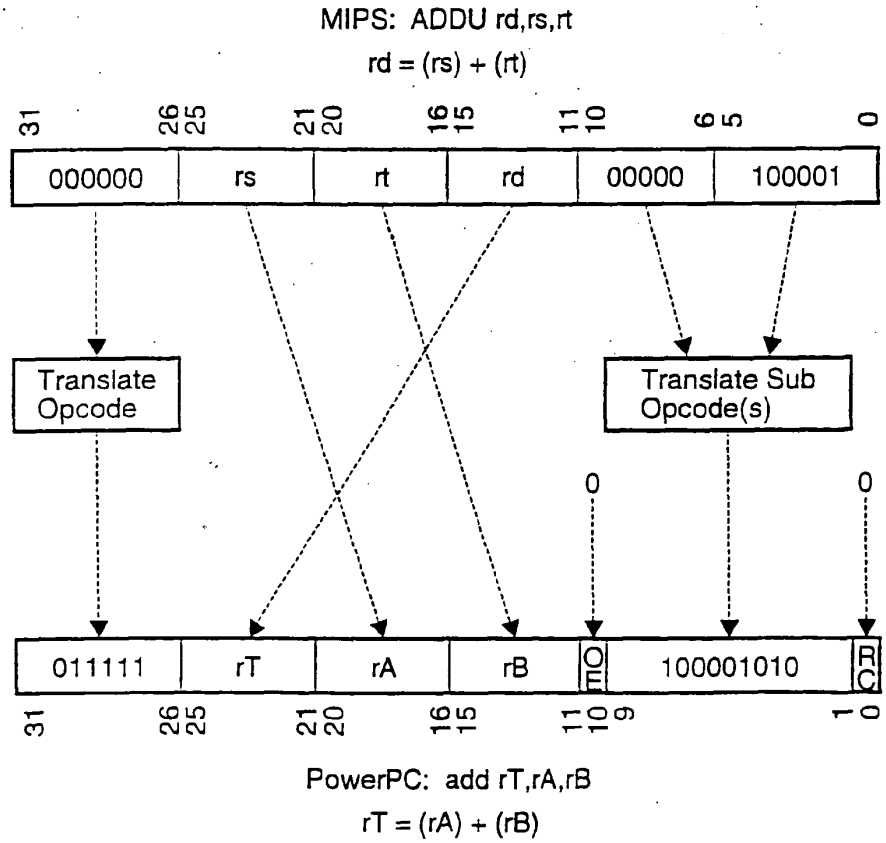


Figure 4

LzLabs GmbH. Ex. 1004-285

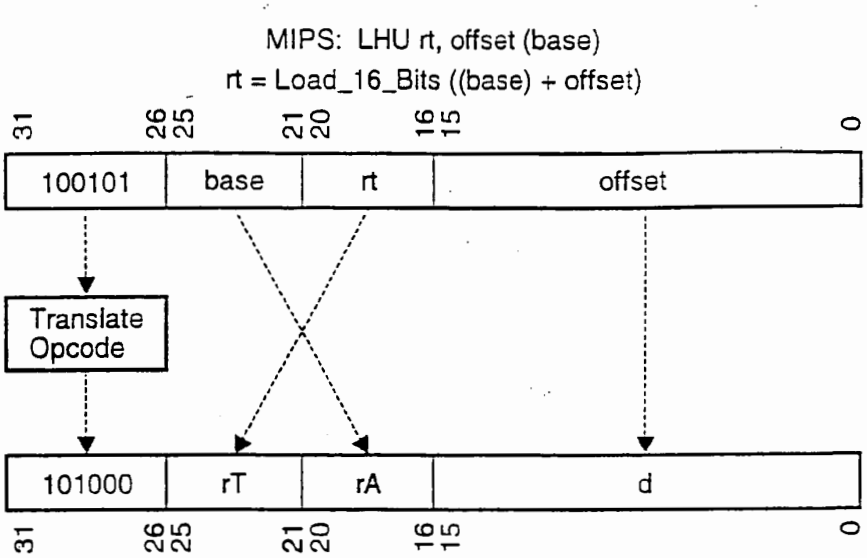


Figure 5

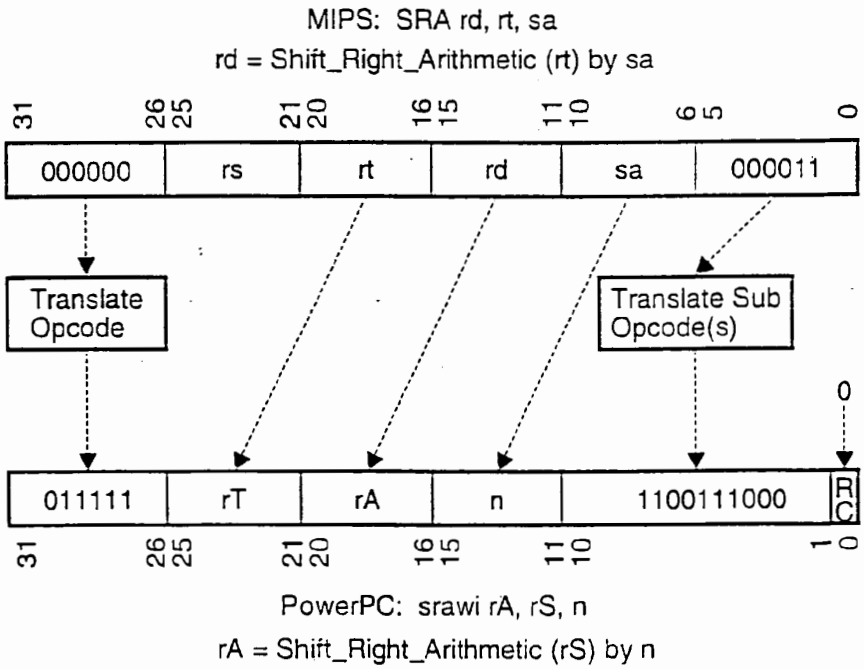


Figure 6

LzLabs GmbH. Ex. 1004-286

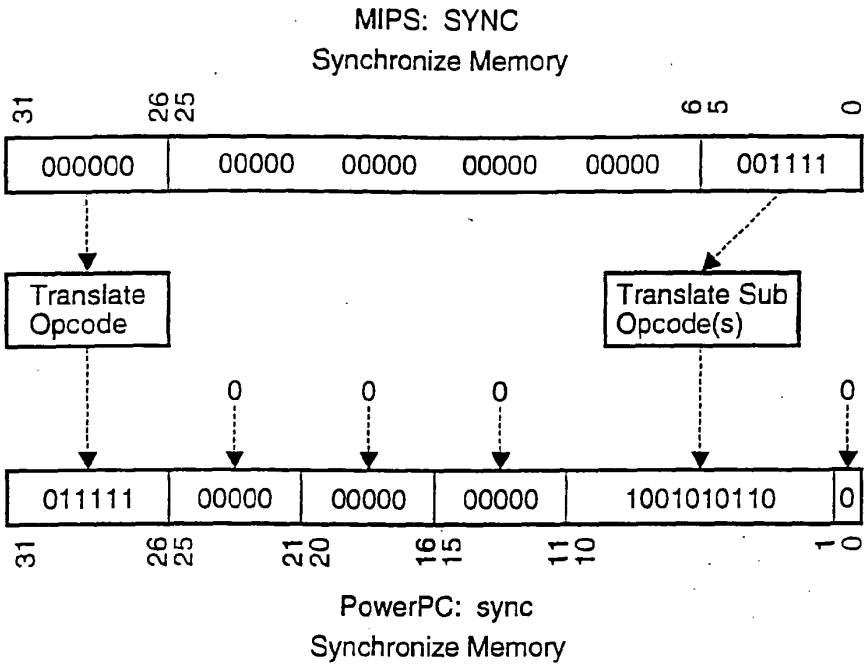


Figure 7

MIPS Instruction

LB rt, offset (base)	Load byte and sign extend

PowerPC Handler

Legacy binary	Binary of legacy instruction
Next pointer	Pointer to next of this type
lbz rT, d(rA)	Load byte and zero extend (execution starts here)
extsb rT	Sign extend loaded byte
blr	Branch back to instruction stream

Figure 8

LzLabs GmbH. Ex. 1004-287

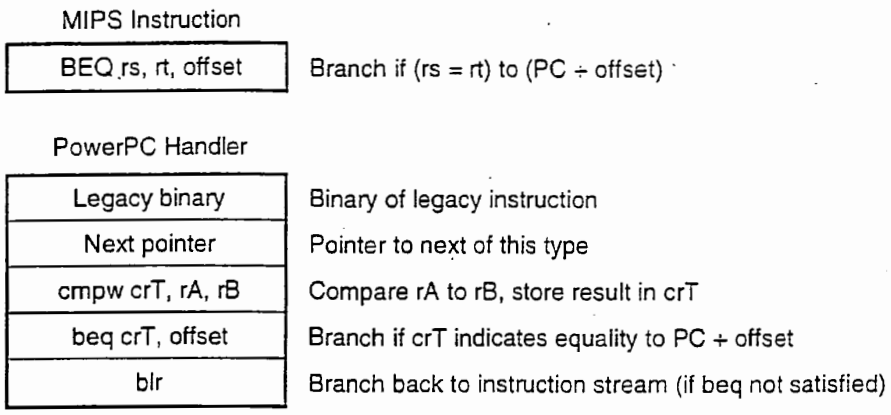


Figure 9

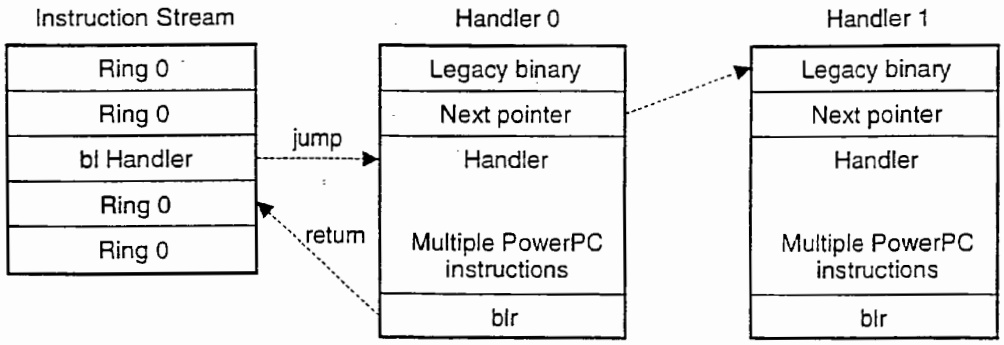


Figure 10

LzLabs GmbH. Ex. 1004-288

MIPS Instruction

SLT rd, rs, rt	Set rd to 1 if rs < rt, else set rd to 0

PowerPC Handler

Legacy binary	Binary of legacy instruction
Next pointer	Pointer to next of this type
stw rscratch, 0 (sp)	Store scratch register to stack
cmpw crT, rA, rB	Compare the two values
mfcr rscratch	Get compare result into scratch register
rlwinm rS, rscratch, 1, 30, 31	Rotate appropriate bit into rd (PowerPC rs)
lwz rscratch, 0 (sp)	Restore scratch register from stack
blr	Branch back to instruction stream

Figure 11

MIPS Instruction

MTC1 rt, fs	Move general register rt to floating point register fs

PowerPC Handler

Legacy Binary	Binary of legacy instruction
Next Pointer	Pointer to next of this type
stw rscratch, 0 (sp)	Store scratch register to stack
lui rscratch, buf@h	Load upper 16-bits of memory buffer address into rscratch
ori rscratch, rscratch, buf@1	Or in the lower 16-bits of memory buffer address
stw rS, 0 (rscratch)	Store rt (PowerPC rS) to memory buffer
lfs fT, 0 (scratch)	Load fs (PowerPC fT) from memory buffer
lwz rscratch, 0 (sp)	Restore scratch register from stack
blr	Branch back to instruction stream

Figure 12

LzLabs GmbH. Ex. 1004-289

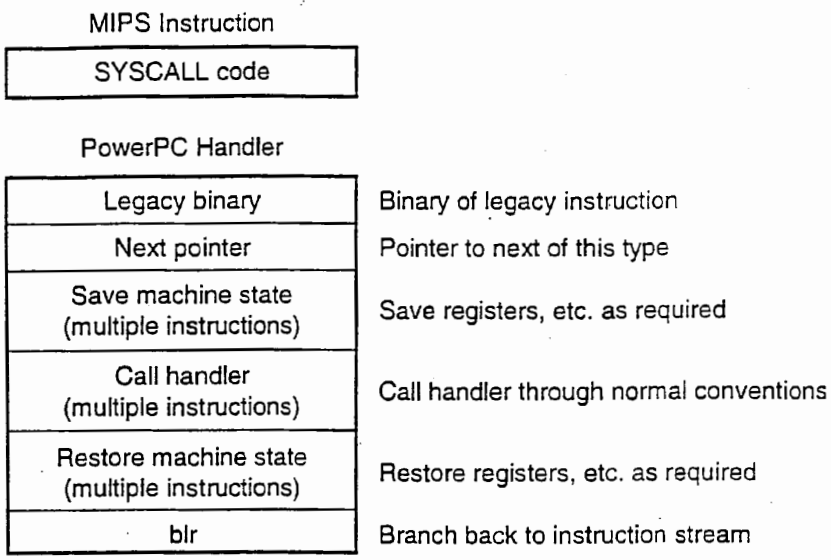


Figure 13A

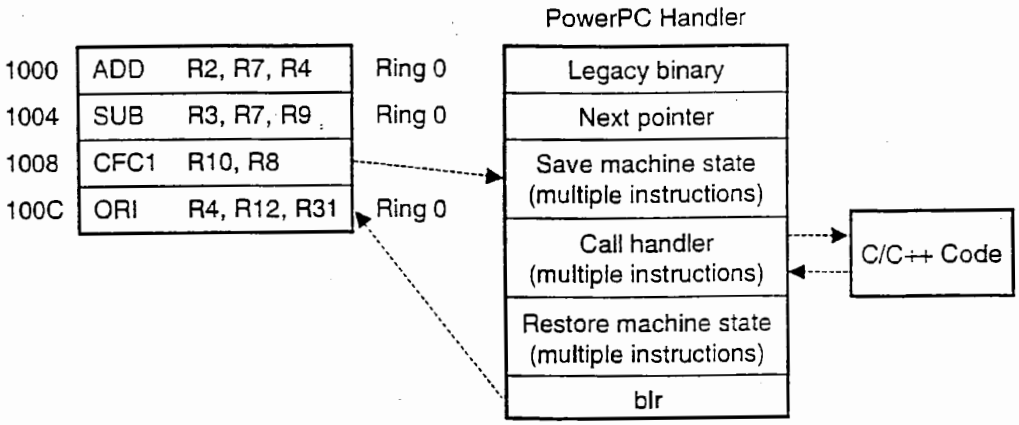


Figure 13B

LzLabs GmbH. Ex. 1004-290

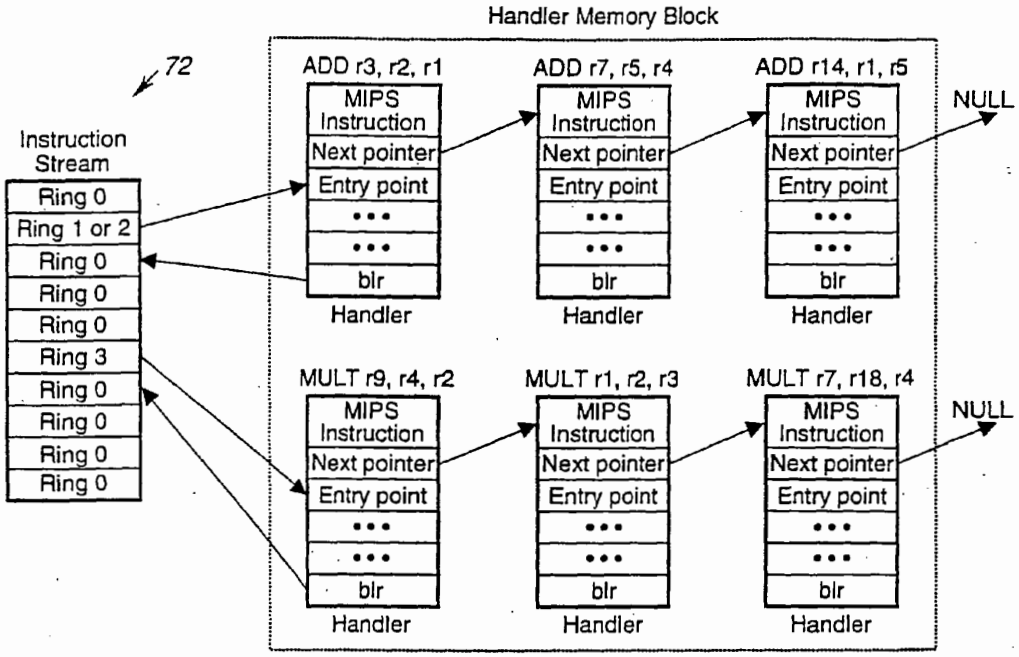


Figure 14

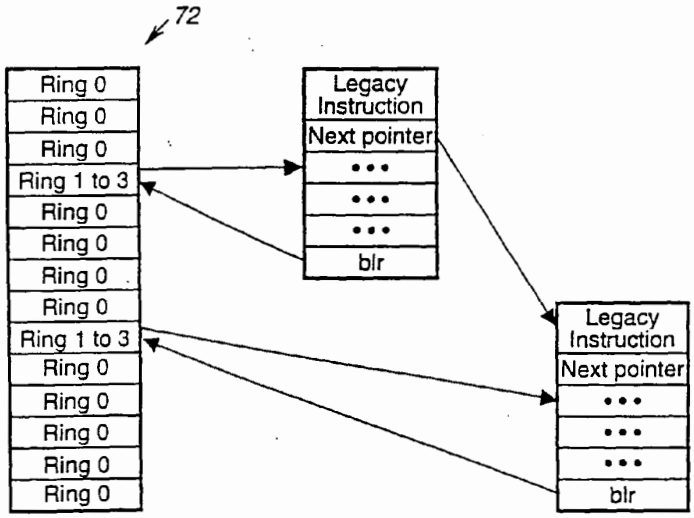
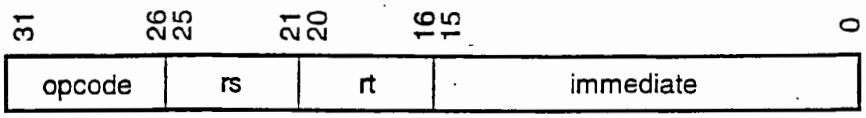


Figure 15

LzLabs GmbH. Ex. 1004-291

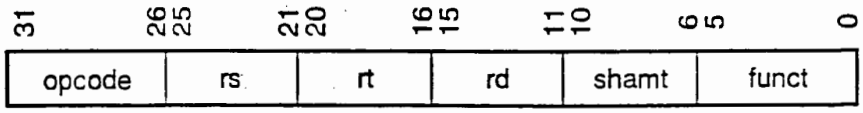
MIPS I-Type (Immediate)
Integer Instruction Format



opcode - 6-bit operation code
rs - 5-bit source register number
rt - 5-bit target (source or destination) or branch condition register
immediate - 16-bit immediate value used for branch displacement or address indexing

Figure 16

MIPS R-Type (Register)
Integer Instruction Format

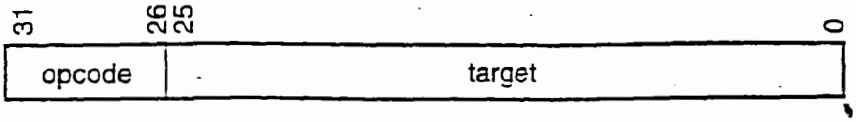


opcode - 6-bit operation code
rs - 5-bit source register number
rt - 5-bit target (source or destination) or branch condition register
rd - 5-bit destination register number
shamt - 5-bit shift amount (when applicable)
funct - 6-bit function field

Figure 17

LzLabs GmbH. Ex. 1004-292

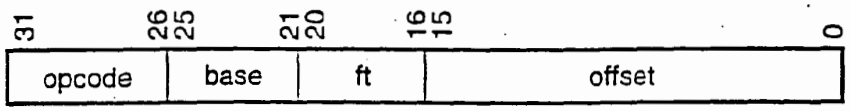
MIPS J-Type (Jump)
Integer Instruction Format



opcode - 6-bit operation code
target - 26-bit target address

Figure 18

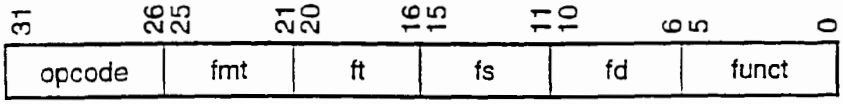
MIPS I-Type (Immediate)
Floating-Point Instruction Format



opcode - 6-bit operation code
base - 5-bit base register number
ft - 5-bit source register number
offset - 16-bit signed immediate offset value from base

Figure 19

MIPS R-Type (Register)
Floating-Point Instruction Format



opcode - 6-bit operation code
fmt - 5-bit format specifier
ft - 5-bit source register number
fs - 5-bit source register number
fd - 5-bit destination register number
func - 6-bit function field

Figure 20

LzLabs GmbH. Ex. 1004-293

PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : G06F 9/455 — A1

(11) International Publication Number: **WO 97/50031**

(43) International Publication Date: 31 December 1997 (31.12.97)

(21) International Application Number: PCT/US97/10341

(22) International Filing Date: 17 June 1997 (17.06.97)

(30) Priority Data:
08/669,772 27 June 1996 (27.06.96) US

(71) Applicant: INTEL CORPORATION [US/US]; 2200 Mission College Boulevard, Santa Clara, CA 95052 (US).

(72) Inventors: FARBER, Yaron; 26 Derech Ha-Yam, 34741 Haifa (IL). LEVHARI, Yossi; 11 Morad HaNachal Street, P.O. Box 238, 40295 Hofit (IL). BARAZ, Leonid; 81 HaOren Street, 28000 Kiryat Ata (IL). LADIRAY, Gallia; 12 Dan Street, 93509 Jerusalem (IL).

(74) Agents: TAYLOR, Edwin, H. et al.; Blakely, Sokoloff, Taylor & Zafman LLP, 7th floor, 12400 Wilshire Boulevard, Los Angeles, CA 90025 (US).

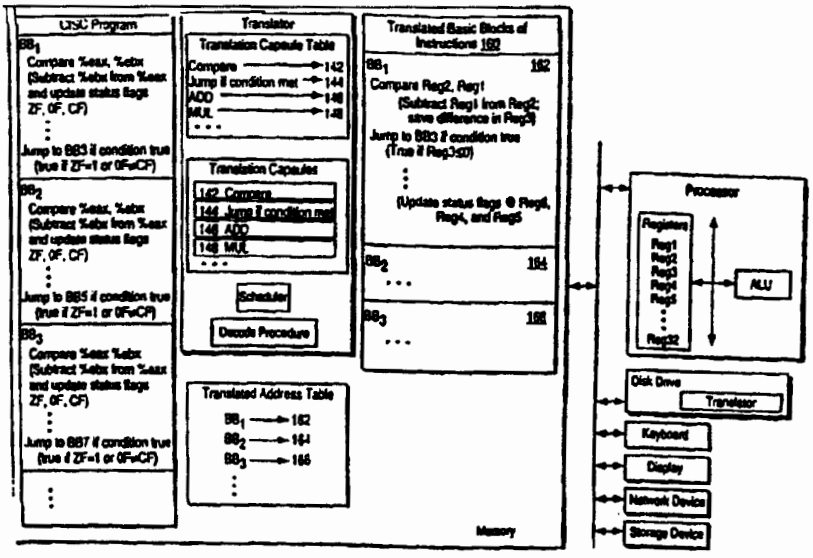
(81) Designated States: AL, AM, AT, AT (Utility model), AU (Petty patent), AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, CZ (Utility model), DE, DE (Utility model), DK, DK (Utility model), EE, EE (Utility model), ES, FI, FI (Utility model), GB, GE, GH, HU, IL, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SK (Utility model), SL, TJ, TM, TR, TT, UA, UG, UZ, VN, YU, ZW, ARIPO patent (GH, KE, LS, MW, SD, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).

Published

With international search report.

Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(54) Title: METHOD FOR INCREASING PERFORMANCE OF BINARY TRANSLATED CONDITIONAL INSTRUCTIONS



(57) Abstract

A method for increasing the performance of binary translated conditional instructions (162). According to one embodiment of the invention, a conditional instruction compatible with the first ISA is decoded (BB1). The condition of the conditional instruction is dependent on at least one status flag (ZF=1). The conditional instruction is translated to be compatible with a second ISA (144), wherein the condition of the conditional instructions is altered to be dependent on a previously computed difference between two values (162), the difference residing in a memory location (REG3).

LzLabs GmbH. Ex. 1004-294

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece			TR	Turkey
BG	Bulgaria	HU	Hungary	ML	Mali	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MN	Mongolia	UA	Ukraine
BR	Brazil	IL	Israel	MR	Mauritania	UG	Uganda
BY	Belarus	IS	Iceland	MW	Malawi	US	United States of America
CA	Canada	IT	Italy	MX	Mexico	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NE	Niger	VN	Viet Nam
CG	Congo	KE	Kenya	NL	Netherlands	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NO	Norway	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	NZ	New Zealand		
CM	Cameroon			PL	Poland		
CN	China	KR	Republic of Korea	PT	Portugal		
CU	Cuba	KZ	Kazakstan	RO	Romania		
CZ	Czech Republic	LC	Saint Lucia	RU	Russian Federation		
DE	Germany	LI	Liechtenstein	SD	Sudan		
DK	Denmark	LK	Sri Lanka	SE	Sweden		
EE	Estonia	LR	Liberia	SG	Singapore		

LzLabs GmbH. Ex. 1004-295

METHOD FOR INCREASING PERFORMANCE OF BINARY TRANSLATED CONDITIONAL INSTRUCTIONS

FIELD OF THE INVENTION

The invention relates to computer systems, and, in particular, to methods for increasing the performance of certain types of instructions.

BACKGROUND OF THE INVENTION

Most often, computer programs are initially written in high-level program statements. In order to be executed by a computer, the program statements are compiled into machine instructions that a microprocessor can recognize and execute. The machine instructions are selected from a set of machine instructions unique to a particular Instruction Set Architecture (ISA).

In the field of ISAs, there have arisen two fundamentally different types of instruction sets: the Complex Instruction Set Computer architecture (CISC) and the Reduced Instruction Set Computer architecture (RISC). The CISC architecture utilizes an instruction set that allows for complicated and flexible ways of calculating such elements as memory addresses. One of the defining characteristics of the CISC architecture is the use of variable length instructions, which adds additional complexity to the decoding hardware. Although very powerfully, all the complexity of the CISC instruction set usually requires additional clock cycles to execute each machine instruction.

A RISC processor, on the other hand, is designed to operate more efficiently by processing a relatively smaller set of instructions. The RISC design is based on the premise that most of the instructions a computer decodes and executes are simple. As a result, RISC architecture limits the number of instructions that are built into the microprocessor, and optimizes each so it can be carried out very rapidly, usually within a single clock cycle.

It is possible for computer program statements that have been compiled into machine instructions for a CISC processor to be translated in order to be executed on a

LzLabs GmbH. Ex. 1004-296

RISC processor. This translation process is normally carried out by a binary translator, which typically consist of a set of instructions stored in memory.

The RISC machine executes the translated instructions by using the RISC instruction set to emulate how the untranslated instructions would be executed on a CISC processor. Practitioners in the art will understand, that when a RISC processor emulates a CISC program, many of the inherent performance advantages of the RISC processor are missed.

Consider, for example, the conditional instruction. A conditional instruction performs an operation, such as transferring the flow of execution from one instruction to another, when a specified condition is true. If the condition is false, the conditional instruction is treated as a no-operation (i.e., an instruction having no effect on the machine's current architectural state).

A conditional instruction is typically based on the results of a previously executed arithmetic instruction, such as the compare instruction. A compare instruction, when executed on a CISC processor, compares two values by subtracting one value from a second value and updates a set of status flags based on the difference of the two values. The actual difference generated by the comparison, however, is not saved in a memory location (i.e., is architecturally transparent).

The status flags are typically represented by one-bit fields in a dedicated register. In one prior art CISC ISA, the status flags include a Zero Flag (ZF) indicating the difference is 0, a Sign Flag (SF) representing the most significant bit of the difference, and an Overflow Flag (OF) indicating the difference is too large or too small to be represented in the number of bits allocated to store the result.

The conditional instruction, when executed on a CISC processor, evaluates a logical combination of the status flags in order to determine whether the condition of the conditional instruction is met. In one example, the conditional instruction evaluates the complex expression (ZF = 1 or SF != OF).

The RISC microprocessor, however, typically has no native support for setting status flags (i.e., dedicated registers). Therefore, a RISC processor emulates a CISC processor executing a compare instruction by allocating bit fields within a general purpose register for the setting of status flags.

LzLabs GmbH. Ex. 1004-297

To emulate a conditional instruction evaluating the set of status flags, the RISC processor employs a series of instructions provided in the RISC instruction set. Use of the series of RISC instructions, however, misses the performance advantage of the RISC processors. Typically, the instruction set of a RISC processor includes one conditional instruction that efficiently evaluates a single value generated by a previously executed compare instruction.

Therefore, it can be appreciated that there is a need for a method of translating conditional instructions that are originally dependent on a logical combination of status flags, to be dependent on an actual difference generated by a previously executed compare instruction.

SUMMARY AND OBJECTS OF THE INVENTION

The present invention provides a method for increasing the performance of binary translated conditional instructions. According to one embodiment of the invention, a conditional instruction compatible with a first ISA is decoded. The condition of the conditional instruction is dependent on at least one status flag.

The conditional instruction is then translated to be compatible with a second ISA, wherein the condition of the conditional instruction is altered to be dependent on a previously computed difference between two values.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention is illustrated by way of example and not limitation in the figures of the accompanying drawings, in which like references indicate similar elements, and in which:

Figure 1 illustrates a computer system capable of implementing one embodiment of the present invention.

Figure 2 is a flow diagram describing the steps in one embodiment of the present invention.

LzLabs GmbH. Ex. 1004-298

DETAILED DESCRIPTION

A method is described for increasing the performance of binary translated conditional instructions executed on a computer system that has no native support for setting status flags.

The present invention translates instructions from a source architecture instruction set that includes support for status flags, to a target architecture instruction set that has no native support for status flags. For example, the source architecture instruction set may be the instruction set of a CISC, Very Long Instruction Word computer (VLIW), or a RISC architecture having native support for status flags, and the target architecture instruction set may be the instruction set of a RISC, or other computer architecture that has no support for status flags.

Each conditional instruction translated by the present invention is typically included in a separate basic block of instructions (BB). The basic block of instructions are translated one at a time.

A basic block of instructions typically includes a contiguous sequence of non-branch machine-instructions ending with a conditional branch instruction. Most computer programs, such as the CISC program of Figure 1, consists of multiple basic blocks of instructions stored in a physical static sequence (e.g., BB_1, BB_2, BB_3 . . .). The basic blocks of the CISC program shown in Figure 1 are instructions compiled to be executed on a CISC program.

The instructions of each basic block of instructions in the CISC program are translated sequentially by the translator shown in the memory of Figure 1. In alternative embodiments, the translator of the present invention may be stored on other computer-readable mediums, including magnetic and optical disk, and other random-access memory devices, accessible via the disk drive shown in Figure 1.

During the translation process each instruction is first decoded into an opcode and operands by the decode procedure shown stored in the translator of Figure 1. A decoded instruction is then translated by one of the translation capsules shown in the translator.

A hash value of a decoded instructions opcode is used to index the Translation Capsule Table shown in the translator. The table in turn provides the address of a

LzLabs GmbH. Ex. 1004-299

translation capsule corresponding to the opcode of the decoded instruction. For example, the opcode for the compare instruction may hash to an entry containing the address value 142 for the compare instruction's translation capsule, as shown in Figure 1.

The translation capsule that corresponds to a decoded instruction's opcode provides a sequence of instructions from the target architecture instruction set in place of the decoded instruction's opcode. In addition, the capsule function translates the memory addresses of the decoded instruction's operands to be compatible with the memory addresses of the target architecture.

CISC program instructions translated by the translator are stored in the area of memory allocated for storing Translated Basic Blocks of instructions 160 shown in Figure 1. These instructions are compatible with the target architecture instruction set.

Figure 2 is a flow diagram describing the steps performed to increase the performance of translated conditional instructions according to one embodiment of the present invention. In block 202, a program counter identifies a basic block of untranslated instructions.

In block 204, a first instruction from the basic block of instructions is decoded by the translator into an opcode and operands. The opcode indicates an operation to be performed by the instruction, and the operands identify a register, cache, or memory location storing data to be operated on by the opcode.

In one embodiment of the present invention, the instruction is decoded by a procedure that includes a separate sequence of instructions stored in the translator. In alternative embodiments, the instructions can be decoded by a procedure stored in memory outside the translator or provided in a hardware component.

In block 206, the translator selects a translation capsule function that corresponds to the opcode of the decoded instruction by hashing the opcode to get an index value from the Translation Capsule Table as described above. The capsule function is selected from a set of translation capsule functions stored in the translator. Each capsule function provides a separate sequence of instructions in place of the decoded instructions opcode in order to emulate the execution the decoded instruction on the target computer architecture. The instructions provided from the capsule function are selected from the instruction set of the target architecture.

LzLabs GmbH. Ex. 1004-300

In addition, the capsule functions translate the memory addresses of the decoded instruction's operands to be compatible with the memory addresses of the target architecture. The translation may consist of adding the decoded instructions operands to an offset base address of the target architecture.

In one embodiment, the translation capsules are indexed in a table by a hash value of the opcodes from the source computer architecture instruction set (e.g., the 12 least significant bits of each opcode). The hash value of the decoded instructions opcode is then used to determine the location of the corresponding translation capsule in the table. Each entry in the table includes the address of the translation capsule corresponding to the opcode, and may include other information regarding the translation capsule, such as whether the opcode takes memory operands.

In block 208, the selected translation capsule function is called and receives the decoded instructions operands and opcode as its parameters. In response, the capsule provides a series of instructions in place of the decoded instructions opcode, translates the memory addresses of the decoded instructions operands, and stores the results in an area of memory allocated to store translated basic blocks of instructions.

If the instruction being translated is a compare instruction, in block 210 the respective capsule function provides a series of instructions that generate a difference between two values and save the difference in a memory location. The memory location may include a general register or an address space of a cache on a processor, for when the translation is performed dynamically (i.e., during execution of a program), or the memory location may be an address space on a disk, for when the translation is performed statically (i.e., when program is not being executed).

For example, a compare instruction compiled to be executed on the source architecture may include the operations of (subtract register %ebx from register %eax and update status flags ZF, 0F, and CF), as shown in BB1 of the CISC program in Figure 1. On the other hand, a compare instruction translated by the present invention may include the operations of (subtract reg1 from reg2 and save difference in reg3), as shown in translated basic block of instructions 162 in Figure 1.

In one embodiment, an additional set of instructions is provided by the respective capsule function to emulate setting the status flags as part of the translated compare instruction. The flag updating operations are included because subsequent

LzLabs GmbH. Ex. 1004-301

instructions in the program may depend on the settings of the flags. In one embodiment, the flags are emulated by allocating bit fields within a general purpose register.

In alternative embodiments, other arithmetic instructions that update status flags based on the results of a data operation may be translated in a similar manner.

If the instruction being translated is a conditional instruction, in block 212 the respective capsule function provides a series of instructions in order to have the condition of the translated conditional operation depend directly on the difference generated by the most recently executed translated compare instruction. The instructions provided by the function reference the memory location where the difference generated by the most recently executed compare instruction has been saved. No status flag evaluating instructions, however, are provided.

For example, the condition of a conditional instruction compiled to be executed on the source architecture may include the operations of (true if ZF =1 or, OF != CF), as shown in BB1 of the CISC program in Figure 1. On the other hand, the condition of a conditional instruction translated by the present invention may include the operation (true if reg3 $\leq$ 0), as shown in translated basic block of instructions 162 in Figure 1.

Typically, computer architectures that have no native support for status flags provide a conditional instruction, in their instruction set, that depends on a single value stored in a register. Therefore, conditional instructions translated by the present invention benefit from better utilization of the more efficient target computer architecture.

In decision block 214, it is determined whether additional machine instructions remain in the basic block of instructions to be translated. If any instructions remain to be translated, the remaining instructions are translated in steps similar to those described in blocks 204-212.

Once the basic block of instructions has been translated, an address for the translated basic block of instructions can be entered into a translated address table. As a result, when the basic block of instructions is called to be executed, the translated address table will provide the address of the translated basic block of instructions.

LzLabs GmbH. Ex. 1004-302

In block 216 the translated instructions of the basic block of instructions are re-scheduled by the translator to be executed by a processor in an order based on inter-instruction dependencies, which is commonly referred to as the critical path of execution. In one embodiment, the critical path of execution includes the instructions of a translated compare instruction which generates and saves a difference between two values in a memory location. These instructions are followed by the translated conditional instruction which evaluates the difference in the memory location.

The instructions of the translated compare instruction that update the status flags are scheduled to be executed by a processor after the critical path of execution, as shown in the translated basic block of instructions 162 in Figure 1. As a result, the critical path of executing the translated compare and the translated conditional instruction is shorter and faster than it would be if it included the instructions to update the status flags.

In an alternative embodiment, the instructions of the translated compare instruction may be scheduled to save the difference to a memory location and update the status flags in parallel when the present invention is performed on a computer system having a processor that is able to execute multiple instructions per clock cycle. This type of processor is commonly referred to as a superscalar processor.

The parallel execution is performed by identifying a group of instructions that do not conflict with each other, or require use of the same functional unit. These instructions are then scheduled to be executed in the same clock cycle.

In the foregoing specification, the invention has been described with reference to specific exemplary embodiments thereof. It will, however, be evident that various modifications and changes may be made thereto without departing from the broader spirit and scope of the invention. For instance, the present invention may be performed dynamically during the execution of a computer program or statically when the program is not being executed. Moreover, where it has been shown that the binary translator, CISC program, translated basic blocks, and translated address table are stored in memory, these items and instructions can also be stored on other computer-readable mediums, including magnetic and optical disk, and other random-access memory devices. The specification and drawings are, accordingly, to be regarded in an illustrative rather than restrictive sense.

LzLabs GmbH. Ex. 1004-303

CLAIMS

What is claimed is:

1. A computer-implemented method for translating a conditioned instruction compatible with a first Instruction Set Architecture (ISA) into a conditional instruction compatible with a second ISA, the method comprising the steps of:

a) decoding a conditional instruction compatible with the first ISA, wherein a condition of the conditional instruction is dependent on at least one status flag: and

b) translating the conditional instruction to be compatible with the second ISA, wherein the condition of the conditional instruction is altered to be dependent on a previously computed difference between two values.

2. The method of claim 1, wherein the previously computed difference between two values is stored in a memory location.

3. The method of claim 2, further including the following steps prior to the step of decoding a conditional instruction:

c) decoding an arithmetic instruction compatible with the first instruction set, the arithmetic instruction when executed, generates the difference between two values and updates said status flag, wherein the difference is architecturally transparent to the first ISA; and

d) translating the arithmetic instruction to be compatible with the second ISA, wherein the arithmetic instruction is altered to have the arithmetic instruction when executed generate the difference between two values and place the difference in the memory location.

4. The method of claim 3, wherein the arithmetic instruction is a compare instruction.

5. The method of claim 4, wherein the method is performed during an execution of a computer program that includes the arithmetic instruction and the conditional instruction.

LzLabs GmbH. Ex. 1004-304

6. The method of claim 5, wherein the first ISA is a Complex Instruction Set Computer Architecture, and the second ISA is a Reduced Instruction Set Computer Architecture.

7. The method of claim 6, further including the step of:

e) scheduling the arithmetic instruction to have the arithmetic instruction update the status flag outside a critical path of execution.

8. The method of claim 6, further including the step of:

e) scheduling the arithmetic instruction to have the arithmetic instruction update the status flag and save the difference in the memory location, in parallel.

9. . A computer-implemented method for translating conditional instructions to increase a performance of conditional instructions when executed on a computer architecture exclusive of native support for status flags, the method comprising the steps of:

decoding an arithmetic instruction, the arithmetic instruction, which when executed, generates a difference between a first value and a second value and updates at least one status flag, wherein the difference is transparent to the computer architecture exclusive of native support for status flags;

translating the arithmetic instruction to have the arithmetic instruction when executed generate the difference between the first value and the second value, wherein the difference is placed in a memory location;

decoding a conditional instruction, wherein a condition of the conditional instruction is dependent on the status flag; and

translating the conditional instruction to have the condition of the conditional instruction dependent on the difference placed in the memory location.

10. A computer-implemented method for translating a first set instructions compatible with a first Instruction Set Architecture (ISA) into a second set of instructions compatible with a second ISA, the method comprising the steps of:

LzLabs GmbH. Ex. 1004-305

a) decoding an arithmetic instruction compatible with the first ISA, the arithmetic instruction which, when executed, generates a difference between two values and updates a status flag, wherein the difference is architecturally transparent to the first ISA; and

b) translating the arithmetic instruction to be compatible with the second ISA, wherein the arithmetic instruction is altered to have the arithmetic instruction when executed generate the difference between two values and place the difference in a memory location.

11. A binary translator comprising:

means for decoding a conditional instruction compatible with a first ISA, wherein a condition of the conditional instruction is dependent on at least one status flag; and

means for translating the conditional instruction to be compatible with a second ISA, wherein the condition of the conditional instruction is altered to be dependent on a previously computed difference between two values.

12. The binary translator of claim 11, wherein the previously computed difference is residing in a memory location.

13. The binary translator of claim 12, further comprising:

means for decoding an arithmetic instruction compatible with the first ISA, the arithmetic instruction which when executed, generates the difference between two values and updates said status flag, wherein the difference is architecturally transparent to the first ISA; and

means for translating the arithmetic instruction to be compatible with the second ISA, wherein the arithmetic instruction is altered to have the arithmetic instruction when executed generate the difference between two values and place the difference in a memory location.

14. The binary translator of claim 13, wherein the arithmetic instruction is a compare instruction.

LzLabs GmbH. Ex. 1004-306

15. The binary translator of claim 14, wherein the means for translating the arithmetic instruction and the means for translating the conditional instruction, translate the respective instructions during an execution of a computer program that includes the arithmetic instruction and the conditional instruction.

16. The binary translator of claim 15, wherein the first ISA is a Complex Instruction Set Computer Architecture, and the second ISA is a Reduced Instruction Set Computer Architecture.

LzLabs GmbH. Ex. 1004-307

17. The binary translator of claim 16, further comprising:

a means for scheduling the arithmetic instruction to have the arithmetic instruction update said status flag outside a critical path of execution.

18. The binary translator of claim 16, further comprising:

a means for scheduling the arithmetic instruction to have the arithmetic instruction update the status flag and save the difference at the memory location in parallel.

19. A computer readable-medium having stored thereon a plurality of a sequence of instructions for translating instructions, the sequence of instructions, which when executed by a processor, cause the processor to perform the steps of:

a) decoding a conditional instruction compatible with a first instruction set architecture (ISA), wherein a condition of the conditional instruction is dependent on at least one status flag; and

b) translating the conditional instruction to be compatible with a second ISA, wherein the condition of the conditional instruction is altered to be dependent on a previously computed difference between two values.

20. The computer readable-medium of claim 19, wherein the difference between the two values is residing in a memory location

21. The computer readable-medium of claim 20, wherein the plurality of sequences of instructions further include additional instructions, which when executed by the processor, cause said processor to perform the steps of:

c) decoding an arithmetic instruction compatible with the first instruction set, the arithmetic instruction which when executed, generates the difference between two values and updates said status flag, wherein the difference is architecturally transparent to the first ISA; and

d) translating the arithmetic instruction to be compatible with the second ISA, wherein the arithmetic instruction is altered to have the arithmetic instruction when

LzLabs GmbH. Ex. 1004-308

executed generate the difference between two values and place the difference in the memory location.

22. The computer readable-medium of claim 21, wherein the arithmetic instruction is a compare instruction.

23. The computer readable-medium of claim 22, wherein the step of translating the arithmetic instruction and the step of translating the conditional instruction are performed during an execution of a computer program that includes the arithmetic instruction and the conditional instruction.

24. The computer readable-medium of claim 23, wherein the first ISA is a Complex Instruction Set Computer Architecture, and the second ISA is a Reduced Instruction Set Computer Architecture.

25. The computer readable-medium of claim 24, wherein the plurality of sequences of instructions further include additional instructions, which when executed by the processor, cause said processor to perform the step of:

e) scheduling the arithmetic instruction to have the arithmetic instruction update said status flag outside a critical path of execution.

26. The computer readable-medium of claim 24, wherein the plurality of sequences of instructions further include additional instructions, which when executed by the processor, cause said processor to perform the step of:

e) scheduling the arithmetic instruction to have the arithmetic instruction update the status flag and save the difference in a memory location, in parallel.

LzLabs GmbH. Ex. 1004-309

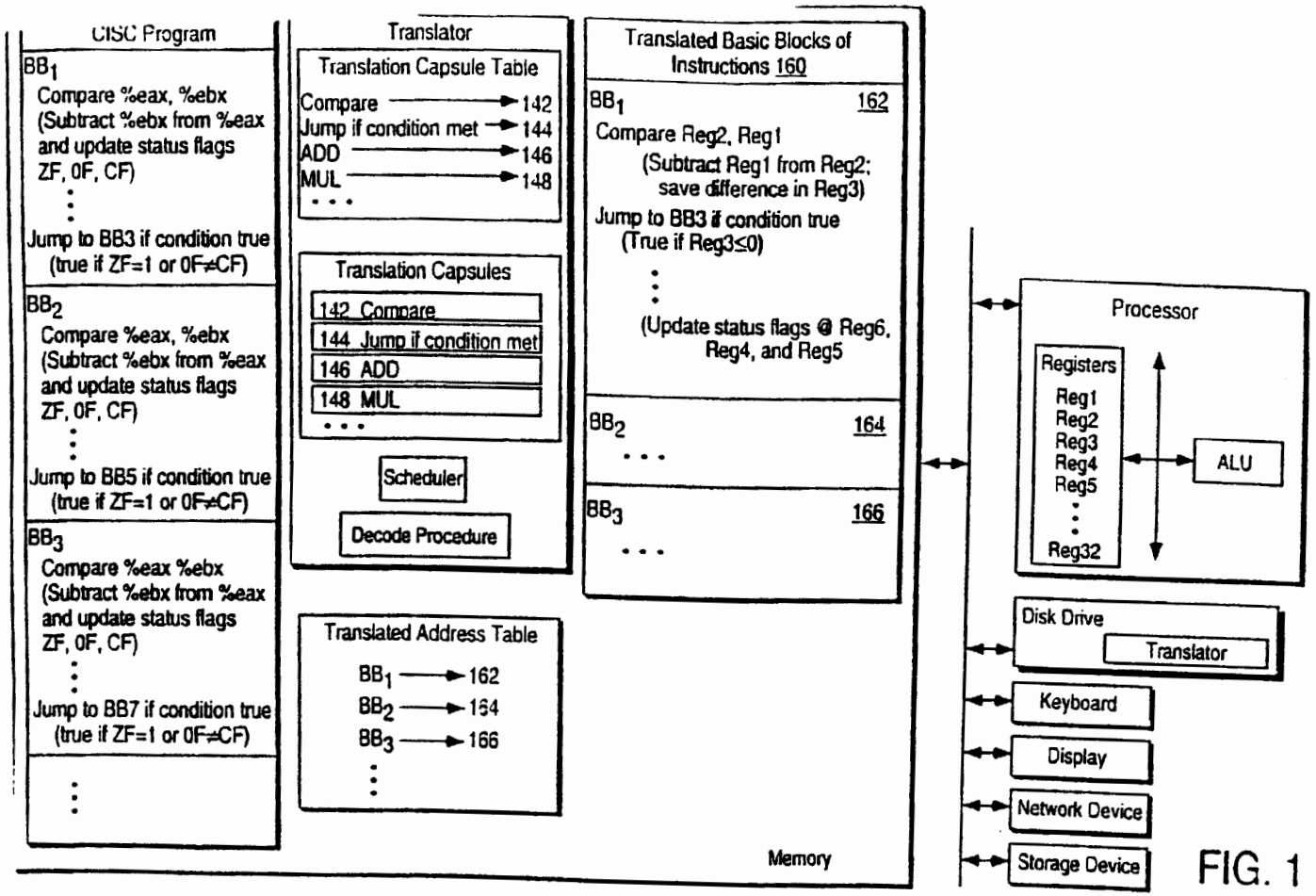
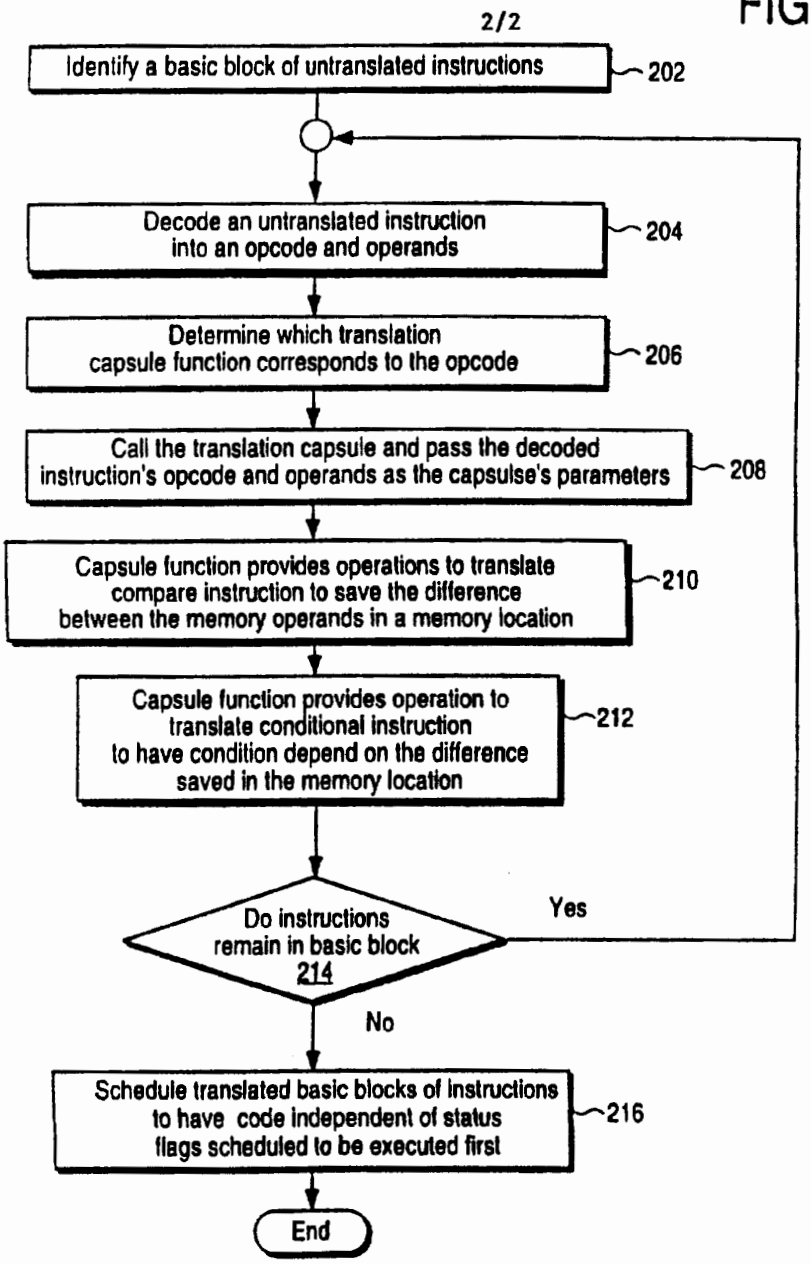


FIG. 1

LzLabs GmbH. Ex. 1004-310

FIG. 2


2/2
Identify a basic block of untranslated instructions
202
Decode an untranslated instruction into an opcode and operands
204
Determine which translation capsule function corresponds to the opcode
206
Call the translation capsule and pass the decoded instruction's opcode and operands as the capsulse's parameters
208
Capsule function provides operations to translate compare instruction to save the difference between the memory operands in a memory location
210
Capsule function provides operation to translate conditional instruction to have condition depend on the difference saved in the memory location
212
Do instructions remain in basic block 214
Yes
No
Schedule translated basic blocks of instructions to have code independent of status flags scheduled to be executed first
216
End

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US97/10341

A. CLASSIFICATION OF SUBJECT MATTER
IPC(6) :G06F 9/455
US CL :395/705
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
U.S. : 395/385, 586, 705, 707, 708, 709

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
APS (Fogg, translate, object, code, program, architecture, instruction, conditional)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 5,313,614 A (GOETTELMANN et al.,) 17 MAY 1994, see abstract, fig. 11, col. 3 line 55 to col. 4 line 50.	1-26
A	US 5,301,325 A (BENSON) 05 APRIL 1994, see abstract.	1-26
Y	US 5,241,678 A (FUTAMURA ET AL.) 31 AUGUST 1993, see abstract, col. 2 line 1 to col. 3 line 16.	1-26
A	US 4,951,195 A (FOGG, JR. ET AL.) 21 AUGUST 1990, see abstract.	1-26
A	MOTOROLA. MC68030 Enhanced 32-bit Microprocessor User's Manual, Second Edition, 1989, pg. 3-71 to 3-72.	1-26

[X] Further documents are listed in the continuation of Box C. [ ] See patent family annex.

*	Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A"	document defining the general state of the art which is not considered to be of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E"	earlier document published on or after the international filing date	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O"	document referring to an oral disclosure, use, exhibition or other means		
"P"	document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search
08 OCTOBER 1997

Date of mailing of the international search report
12 NOV 1997

Name and mailing address of the ISA/US
Commissioner of Patents and Trademarks
Box PCT
Washington, D.C. 20231
Facsimile No. (703) 305-3230

Authorized officer
RICHARD ELLIS
Telephone No. (703) 305-9600

Form PCT/ISA/210 (second sheet)(July 1992)*

LzLabs GmbH. Ex. 1004-312

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US97/10341

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	GAINES, R. Stockton. On the Translation of Machine Language Programs, Communications of the ACM, Volume 8, Number 12, December 1965, pp. 736-741.	1-26
A	BERGH, Arndt B. et al. HP3000 Emulation on HP Precision Architecture Computers, Hewlett-Packard Journal, December 1987, pp. 87-89.	1-26
A	HUNTER, Colin et al. DOS at RISC, Byte Magazine, November 1989, pp. 361-368.	1-26
A	SAARI, Michael. 68000 Binary Code.Translator, FORML Conference Proceedings, 1987, pp. 48-52.	1-26
A	BANNING, John. The XDOS Binary Code Conversion System, IEEE, 1989, pp. 282-287.	1-26

Form PCT/ISA/210 (continuation of second sheet)(July 1992)*

LzLabs GmbH. Ex. 1004-313

Electronic Patent Application Fee Transmittal

Application Number:	11668623
Filing Date:	30-Jan-2007
Title of Invention:	METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS
First Named Inventor/Applicant Name:	Reid T. Copeland
Filer:	John E. Campbell/Jennifer Keenan
Attorney Docket Number:	POU920060152US1

Filed as Large Entity

Utility under 35 USC 111(a) Filing Fees

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Basic Filing:				
Pages:				
Claims:				
Miscellaneous-Filing:				
Petition:				
Patent-Appeals-and-Interference:				
Post-Allowance-and-Post-Issuance:				
Extension-of-Time:				

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Miscellaneous:				
Submission- Information Disclosure Stmt	1806	1	180	180
		Total in USD ($)		**180**

Electronic Acknowledgement Receipt

EFS ID:	4252348
Application Number:	11668623
International Application Number:	
Confirmation Number:	4180
Title of Invention:	METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS
First Named Inventor/Applicant Name:	Reid T. Copeland
Customer Number:	46369
Filer:	John E. Campbell/Jennifer Keenan
Filer Authorized By:	John E. Campbell
Attorney Docket Number:	POU920060152US1
Receipt Date:	07-NOV-2008
Filing Date:	30-JAN-2007
Time Stamp:	15:05:00
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	yes
Payment Type	Deposit Account
Payment was successfully received in RAM	$180
RAM confirmation Number	330
Deposit Account	090463
Authorized User	
The Director of the USPTO is hereby authorized to charge indicated fees and credit any overpayment as follows: Charge any Additional Fees required under 37 C.F.R. Section 1.16 (National application filing, search, and examination fees) Charge any Additional Fees required under 37 C.F.R. Section 1.17 (Patent application and reexamination processing fees)	

Charge any Additional Fees required under 37 C.F.R. Section 1.19 (Document supply fees)					
File Listing:					
Document Number	**Document Description**	**File Name**	**File Size(Bytes)/ Message Digest**	**Multi Part /.zip**	**Pages (if appl.)**
1	Information Disclosure Statement Letter	POU920060152US1_IDS_Transmittal.pdf	30141 b22043dd5b5489f6d440e85b4d6a59001ae9ff2e	no	1
Warnings:					
Information:					
2	Information Disclosure Statement (IDS) Filed (SB/08)	POU920060152US1_IDS.pdf	30831 d4ca32241145367c49c011d5d9e65180b23987fe	no	3
Warnings:					
Information:					
This is not an USPTO supplied IDS fillable form					
3	Foreign Reference	EP1455273A.pdf	438199 1df53999b7fa1346db30c4f3e7f2799440875d6f	no	26
Warnings:					
Information:					
4	Foreign Reference	WO9750031.pdf	797969 f817d12f7f4dd2dc8cbe6329fec8a801b25a6421	no	20
Warnings:					
Information:					
5	Fee Worksheet (PTO-06)	fee-info.pdf	30489 e5913c23a13f410607b798b6e3c322082d155e3f	no	2
Warnings:					
Information:					
		Total Files Size (in bytes):	1327629		

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office
If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

TRANSMITTAL OF INFORMATION DISCLOSURE STATEMENT (Under 37 CFR 1.97(d))	Docket No. POU920060152US1

In Re Application Of: **Reid T. Copeland et. al.**

Application No.	Filing Date	Examiner	Customer No.	Group Art Unit	Confirmation No.
11/668,623	**1/30/2007**		**33558**	**2183**	**4180**

Title: **Method For Efficiently Emulating Computer Architecture Condition Code Settings**

Address to:
Commissioner for Patents

The Information Disclosure Statement submitted herewith is being filed after the period specified in 37 CFR 1.97(c), and on or before payment of the issue fee, and is accompanied by the Statement as specified in 37 CFR 1.97(e) and the fee set forth in 37 CFR 1.17(p).

☐ A check in the amount of is attached.

☒ The Director is hereby authorized to charge and credit Deposit Account No. **090463** as described below.

- ☒ Charge the amount of **$180.00**
- ☒ Credit any overpayment.
- ☒ Charge any additional fee required.

☐ Payment by credit card. Form PTO-2038 is attached.

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

Certificate of Transmission by Facsimile*

I certify that this document and authorization to charge deposit account is being facsimile transmitted to the United States Patent and Trademark Office (Fax no.) on

_____
(Date)

Signature

Typed or Printed Name of Person Signing Certificate

***This certificate may only be used if paying by deposit account.**

Certificate of Mailing by First Class Mail

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to "Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450" [37 CFR 1.8(a)] on

_____.
(Date)

Signature of Person Mailing Correspondence

Typed or Printed Name of Person Mailing Certificate

/John E. Campbell/
Signature

Dated: **November 7, 2008**

John E. Campbell, Reg. 52,687
IBM Corporation
IP Law Department
2455 South Road, MS P386
Poughkeepsie, NY 12601
(845) 433-1156

cc:

P10B/REV05

Doc code :IDS
Doc description: Information Disclosure Statement (IDS) Filed

PTO/SB/08a (10-08)
Approved for use through 11/30/2008. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (**Not for submission under 37 CFR 1.99)**		
	Application Number	11668623
	Filing Date	2007-01-30
	First Named Inventor	Reid T. Copeland
	Art Unit	2183
	Examiner Name	
	Attorney Docket Number	POU920060152US1

U.S.PATENTS						
Examiner Initial*	Cite No	Patent Number	Kind Code[1]	Issue Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	5774694		1998-06-30	Murari Srinivasan	
If you wish to add additional U.S. Patent citation information please click the Add button.						

U.S.PATENT APPLICATION PUBLICATIONS						
Examiner Initial*	Cite No	Publication Number	Kind Code[1]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear
	1	20060173931		2006-08-03	James Broadhurst	
If you wish to add additional U.S. Published Application citation information please click the Add button.						

FOREIGN PATENT DOCUMENTS								
Examiner Initial*	Cite No	Foreign Document Number[3]	Country Code[2] i	Kind Code[4]	Publication Date	Name of Patentee or Applicant of cited Document	Pages,Columns,Lines where Relevant Passages or Relevant Figures Appear	T[5]
	1	WO 97/50031	EP		1997-12-31	Intel Corp		☐
	2	EP 1455273	EP		2004-09-08	Northrop Grumman Corp.		☐
If you wish to add additional Foreign Patent Document citation information please click the Add button								

NON-PATENT LITERATURE DOCUMENTS

LzLabs GmbH. Ex. 1004-320

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	11668623
	Filing Date	2007-01-30
	First Named Inventor	Reid T. Copeland
	Art Unit	2183
	Examiner Name	
	Attorney Docket Number	POU920060152US1

Examiner Initials*	Cite No	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc), date, pages(s), volume-issue number(s), publisher, city and/or country where published.	T[5]
	1	International Search Report PCT/EP2008/050725 POU060152 4/18/2008	☐

If you wish to add additional non-patent literature document citation information please click the Add button

EXAMINER SIGNATURE

Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through a citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

[1] See Kind Codes of USPTO Patent Documents at www.USPTO.GOV or MPEP 901.04. [2] Enter office that issued the document, by the two-letter code (WIPO Standard ST.3). [3] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [4] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [5] Applicant is to place a check mark here if English language translation is attached.

LzLabs GmbH. Ex. 1004-321

INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Not for submission under 37 CFR 1.99)		
	Application Number	11668623
	Filing Date	2007-01-30
	First Named Inventor	Reid T. Copeland
	Art Unit	2183
	Examiner Name	
	Attorney Docket Number	POU920060152US1

CERTIFICATION STATEMENT

Please see 37 CFR 1.97 and 1.98 to make the appropriate selection(s):

☐ That each item of information contained in the information disclosure statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(1).

OR

☐ That no item of information contained in the information disclosure statement was cited in a communication from a foreign patent office in a counterpart foreign application, and, to the knowledge of the person signing the certification after making reasonable inquiry, no item of information contained in the information disclosure statement was known to any individual designated in 37 CFR 1.56(c) more than three months prior to the filing of the information disclosure statement. See 37 CFR 1.97(e)(2).

☐ See attached certification statement.

☐ Fee set forth in 37 CFR 1.17 (p) has been submitted herewith.

☒ None

SIGNATURE

A signature of the applicant or representative is required in accordance with CFR 1.33, 10.18. Please see CFR 1.4(d) for the form of the signature.

Signature	/John E. Campbell/	Date (YYYY-MM-DD)	2008-11-07
Name/Print	John E. Campbell	Registration Number	52,687

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 1 hour to complete, including gathering, preparing and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

LzLabs GmbH. Ex. 1004-322

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 1 455 273 A2

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
08.09.2004 Bulletin 2004/37

(51) Int Cl.[7]: **G06F 9/455**

(21) Application number: **04002677.5**

(22) Date of filing: **01.03.2004**

(84) Designated Contracting States:
AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HU IE IT LI LU MC NL PL PT RO SE SI SK TR
Designated Extension States:
AL LT LV MK

(30) Priority: **06.03.2003 US 382598**

(71) Applicant: **Northrop Grumman Corporation**
Los Angeles, CA 90067-2199 (US)

(72) Inventors:
- **Cannon, William J.**
  Centerville, Ohio 45459 (US)
- **Zwirner, Eric W.**
  Dayton Ohio 45405 (US)
- **Hoerig, Timothy R.**
  Beavercreek Ohio 45431 (US)
- **Ward, Paul D.**
  Centerville Ohio 45458 (US)

(74) Representative: **Lieck, Hans-Peter, Dipl.-Ing.**
Boeters & Lieck
Widenmayerstrasse 36
80538 München (DE)

(54) Emulation of microprocessor instructions

(57) A method and system (20) for emulating instructions of legacy microprocessors which execute a compiled high-ordered language (28), such as C/C++, in which the compiled code is structured such that data and instructions segments (22; 24) are segregated. In order to improve the real-time performance of the system, legacy instructions (32) are directly mapped to equivalent instructions (34) of the host processor where possible. Additional techniques may optionally be employed to further increase the real-time performance of the system. By utilizing the direct mapping of the legacy instructions to host instructions, the emulation system (20) in accordance with the present invention provides increased real-time performance for relatively modern RISC microprocessors.

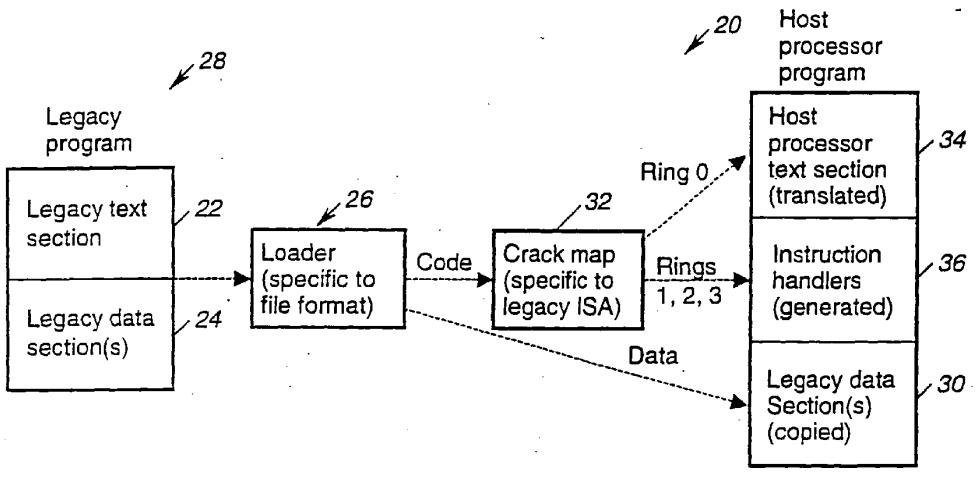


Figure 1

EP 1 455 273 A2

LzLabs GmbH. Ex. 1004-323

Description

[0001] The present invention relates to a method and apparatus for emulating instructions of a legacy microprocessor and, more particularly, to an emulation system for a legacy microprocessor, such as a 32-bit microprocessor, which executes a compiled higher-order language, such as C/C++, in which the compiled code is structured such that data and instruction segments are structured, for example, as in Reduced Instruction Set Computer (RISC) microprocessors, wherein the legacy instructions are directly mapped to an equivalent instruction of the host processor, where possible, to improve the real-time performance of the system.

[0002] It is known that microprocessors are configured with different instruction-set architectures (ISA). The ISA determines the instruction set for a particular microprocessor. Application programs are executed by a microprocessor, normally written in a relatively high-level language, which, in turn, are compiled into machine instructions compatible with the instruction set for a specific microprocessor.

[0003] Due to the age obsolescence of many existing microprocessors and corresponding slow speed, it is often desired to replace outdated existing microprocessors, hereinafter referred to as legacy microprocessors, with newer microprocessors, hereinafter referred to as host microprocessors. Unfortunately, depending on the particular upgrade, most times the instruction set of the host microprocessor is incompatible with the instruction set of the legacy microprocessor.

[0004] One way to solve this problem is to rewrite the existing application program taking into account the instruction set of the host microprocessor. Such a technique is relatively cumbersome and quite expensive. In order to work around instruction set incompatibilities, software systems are known for emulating instructions of legacy microprocessors. Such systems are often referred to as emulation systems. Such emulation systems are known to emulate the instruction set of the legacy microprocessor in order to enable the application programs written in terms of the instruction set for the legacy microprocessor to be executed by a newer, faster host microprocessor.

[0005] In many known emulation systems, legacy instructions are fetched by the host microprocessor, which, in turn, uses a look-up table to interpret the legacy instruction. With such emulation systems, since each legacy instruction must be interpreted, systems which incorporate cache memory are known to suffer from relatively high probability of cache misses, which decreases the overall throughput of the system.

[0006] In order to improve the throughput or speed of such emulation systems, U.S. Patent No. 6,041,402 discloses a direct vectored legacy instruction set emulation system. Rather than fetching legacy instructions and interpreting the instructions in software, the direct vectored legacy instruction set emulation system fetches direct vectors to software routines which emulate the legacy instructions. By fetching the direct vectors, the need for look-up tables for interpreting the legacy instructions is eliminated. Since look-up table accesses are eliminated with such a system, the probability of cache misses is greatly reduced, which, in turn, increases the throughput of the system. Unfortunately, the direct vectored legacy instruction set emulation requires too much performance overhead for relatively modern microprocessors, such as 32-bit RISC microprocessors. Thus, there is a need for an emulation system and method for emulating instructions of legacy microprocessors which provides improved real-time performance relative to known emulation systems.

[0007] Briefly, the present invention relates to a method and system for emulating instructions of legacy microprocessors which execute a compiled high-order language, such as C/C++, in which the compiled code is structured such that data and instructions segments are segregated, for example, as in reduced instruction set computer (RISC) microprocessors. In order to improve the real-time performance of the system, legacy instructions are directly mapped to equivalent instructions of the host processor, where possible. Additional techniques may optionally be employed to further increase the real-time performance of the system. When exact matches cannot be made, an unconditional branch instruction to an instruction handler is used. The instruction handler is made up of two or more host processor instructions that ultimately perform the same function as the replaced legacy instruction. For legacy instructions which cannot be emulated solely with host instructions, these instructions are emulated by instruction handlers which include several host instructions and which incorporate the use of scratch pad registers. For legacy instructions that involve architectural features of a legacy microprocessor that cannot be handled easily by the host processor's assembly language, an instruction handler is formed from a high level language, such as, C/C++ code. By utilizing the direct mapping of the legacy instructions to host instructions and one or more of the above-mentioned emulation techniques, the emulation system in accordance with the present invention provides increased real-time performance for relatively modern RISC microprocessors.

[0008] These and other advantages of the present invention will be readily understood with reference to the following specification and attached drawing wherein:

[0009] FIG. 1 is a high-level block diagram of the emulation system in accordance with the present invention.

[0010] FIG. 2 is a software flow diagram of an instruction cracking process used in accordance with the present invention.

[0011] FIG. 3 is a diagram of a portion of memory illustrating memory mapping in accordance with one aspect of the

LzLabs GmbH. Ex. 1004-324

invention.

[0012] FIG. 4 illustrates an exemplary example of direct mapping of the op code of an exemplary legacy ADDU (add unsigned) instruction to a host ADD instruction.

[0013] FIG. 5 is similar to FIG. 4 but illustrates the mapping of an exemplary legacy LHU (load half word unsigned) instruction to a host instruction.

[0014] FIG. 6 is similar to FIG. 4 and illustrates the mapping of an exemplary legacy SRA (shift right algebraic) instruction to a host instruction.

[0015] FIG. 7 is similar to FIG. 4 and illustrates the mapping of an exemplary legacy SYNC instruction to a host instruction.

[0016] FIG. 8 illustrates an exemplary host instruction handler for an exemplary legacy LB (load byte) instruction.

[0017] FIG. 9 is similar to FIG. 8 but for an exemplary legacy BEQ (branch if equal) instruction.

[0018] FIG. 10 illustrates an instruction stream which includes a number of direct mapped instructions, for example, as illustrated in FIGS. 4 through 7, illustrating unconditional branches to instruction handlers for legacy instructions which cannot be directly mapped to a host instruction.

[0019] FIG. 11 is an example of an emulation of an exemplary legacy instruction fully emulated by a plurality of host instructions which requires a scratch-pad register for a MIPS SLT (set if less than) instruction to a PowerPC handler.

[0020] FIG. 12 is similar to FIG. 11, except that it illustrates the emulation of an exemplary MTC1 (move to co-processor 1) instruction.

[0021] FIG. 13A is an example of an emulation of a legacy Ring 3 SYSCALL instruction that requires emulation by instruction handler, formed from higher-level instructions.

[0022] FIG. 13B is similar to FIG. 13A but for a Ring 3 CFCz legacy instruction.

[0023] FIG. 14 illustrates an instruction stream containing various legacy instructions and the corresponding handler memory space.

[0024] FIG. 15 illustrates execution of the emulation program in accordance with the present invention.

[0025] FIGS. 16-20 illustrate the various instruction formats for a MIPS type RISC microprocessor.

DETAILED DESCRIPTION

[0026] The present invention relates to a method and apparatus for emulating instructions of a particular category legacy microprocessors which execute a compiled higher-order language, such as C/C++, Fortran, or ADA. Such compiled code is typically structured such that data and instruction segments are segregated. The compiled code also conforms to other platform specific Application Binary Interfaces (ABI) which describe things like procedure calling and linkage conventions and register usage conventions.

[0027] The present invention relates to the emulation of a 32-bit RISC microprocessor using another 32-bit RISC microprocessor. As used herein, host processor denotes the processor that executes the emulation program. The term legacy processor is used to identify the processor being emulated.

[0028] The emulation technique in accordance with the present invention assumes similarities between the host and legacy processors to enable the legacy instructions to be categorized into one or more of the following four categories:

- Ring 0 - These legacy instructions can be replaced with a single host processor instruction that performs the same function. Modifying the op code of the instruction and rearranging the instruction field is generally all that is necessary for this category.
- Ring 1 - This type of legacy instruction can be replaced with two or more host processor instructions that perform the same function and do not require scratch-pad registers to perform the operation.
- Ring 2 - These legacy instructions can be replaced with two or more host processor instructions that perform the same function but require one or more scratch-pad registers to perform the operation.
- Ring 3 - These legacy instructions generally relate to advanced architectural features of the legacy microprocessor and further relate to instructions that are far too complex to be implemented utilizing only the host processor assembly language. These instructions are handled almost entirely in a high-level language, such as C/C++.

[0029] In order to enable the host processor to execute programs written for the legacy processor, the legacy processor's binary instructions must be rendered into the host processor's binary instructions in one of the four categories mentioned above. There are many operations performed by all microprocessors, such as simple arithmetic (i.e., add, subtract, multiply), logical operations (i.e., logical and, or, not), and data moving (i.e., load and store). Many of these instructions are very similar and identical between different brands and/or models of RISC microprocessors and tend to appear frequently throughout many application programs. Thus, by taking advantage of the similarities of the instruction sets between the host microprocessor and the legacy microprocessor and utilizing direct mapping as set forth in the Ring 0 category, wherever possible, the real-time performance of the system will be greatly improved. When

LzLabs GmbH. Ex. 1004-325

exact matches cannot be made, the legacy instruction is replaced with an unconditional branch instruction to a legacy instruction handler made up of two or more host processor instructions that ultimately perform the same function as the replaced legacy instruction, for example, for legacy instructions categorized as Ring 1 and Ring 2. The use of unconditional branches helps retain performance by taking advantage of the host microprocessor branch prediction capabilities. For advanced instructions that involve advanced features of the legacy microprocessor or cannot be easily handled with the host processor assembly language, Ring 3 category instructions are branched out to a higher-level language, such as C/C++, to handle the operation at a higher level.

[0030] In order to retain the performance advantage of Ring 0, Ring 1, and Ring 2 and because the register usage conventions differ between the legacy and host microprocessors, the legacy registers are re-mapped to the corresponding functional equivalent host register. For example, the stack pointer for a PowerPC host microprocessor is r1, while the stack pointer for a MIPS R4400 microprocessor is r29. Known emulator systems are known to keep legacy registers in memory, which require memory accesses and additional host instructions to manipulate a legacy register value, which substantially decreased the real-time performance. By re-mapping the legacy registers to the host processor registers, the real-time performance is enhanced relative to known emulation systems.

[0031] The invention is described in terms of a 32-bit R4400 RISC microprocessor by MIPS Technologies, Inc. (http://www.mips.com) as a legacy processor and a PowerPC host processor (http://www.motorola.com/powerpc). However, as mentioned above, the principles of the present invention are applicable to any compatible microprocessors in which some of the legacy processor instructions can be directly mapped to a host processor instruction. It is known that the number of 1:1 matches between instructions can be relatively high for compatible host/legacy processor combinations.

SYSTEM OVERVIEW

[0032] A high-level block diagram of the emulation system and method in accordance with the present invention is illustrated in FIG. 1 and generally identified with the reference numeral 20. As mentioned above, the emulation system and method in accordance with the present invention is amenable for use with legacy microprocessors in which the instructions and data are segregated, as indicated by the boxes 22 and 24. A file loader 26 locates the legacy program, generally identified with the reference numeral 28. File loaders are well known with the art. The file loader 26 reads the legacy file format and fetches the legacy binary files. As shown in FIG. 1, the file loader 26 fetches both the legacy instructions 22 as well as the legacy data files 24. These legacy instruction and data files 22 and 24 may be contained on a disk, a read-only memory, or fetched over a network. If the legacy data 24 is static, it is simply copied to a legacy data memory space 30 associated with the host microprocessor program. Alternatively, the legacy data may be created and initialized for data sections that are created dynamically as the program runs.

[0033] The legacy instructions 22 are provided by the file loader 26 to a crack map object 32. The crack map object 32 is specific to the legacy and host instruction set architectures, as will be discussed in more detail below. As will also be discussed in more detail below, the crack map object 32 analyzes each legacy instruction 22 and categorizes the instruction according to the four exemplary categories discussed above (i.e., Ring 0, Ring 1, Ring 2 and Ring 3). As shown in FIG. 1, Ring 1-3 instructions are placed in separate handler memory space from the Ring 0 memory space. The Ring 0 instructions are directly mapped by the crack map object 32 to the host processor memory space 34. All other instructions are mapped to the instruction handlers 36.

CRACK MAP OBJECT

[0034] The crack map object 32 converts each legacy binary instruction received from the file loader 26 into one or more host processor instructions. As mentioned above, the crack map object 32 described and illustrated herein is for emulating instructions of the legacy MIPS R4400 RISC microprocessor by way of a PowerPC G3/G4 processor.

[0035] Referring to FIG. 2, the crack map object 32 fetches and reads each legacy instruction 22 in steps 40 and 42. Each legacy instruction is then decoded based on its op-code in steps 44 and 46. More particularly, with reference to Tables 1 and 2 below, each legacy instruction is listed in a look-up table along with its legacy op-code along with other data fields, as described below. These look up tables, as will be discussed in more detail below, include the Ring category for each legacy instruction 22. Accordingly, the crack map object 32 checks in step 48 to determine whether the instruction falls into the Ring 0 category. If so, the crack map object 32 directly maps the legacy instruction 22 to a host instruction by translating the op-codes, as will be discussed in more detail below, in step 50. The translated host instruction is then written to the application program instruction stream in step 52.

[0036] If the legacy instruction 22 is not a Ring 0 category, an unconditional branch instruction to the next handler in the instruction stream is inserted in the instruction stream in step 54. The system then checks in steps 56, 58 and 60 to determine whether the legacy instruction falls into a Ring 1 category, Ring 2 category and Ring 3 category.

[0037] If the legacy instruction 22 falls into a Ring 1 category, as determined in 56, instructions to the instruction handler's 36 (FIG. 1) are written in step 58B and a return branch instruction is inserted at the end of the instruction

LzLabs GmbH. Ex. 1004-326

handler 36 in step 60A. Should the instruction be determined to be a Ring 2 category instruction, as determined in step 58A, the crack map object 32 inserts code to save data from the scratch pad registers and subsequently writes instructions to the instruction handler space 36 (FIG. 1) in step 64.

[0038] As mentioned above, Ring 2 category instructions require a plurality of host processor instructions along with scratch pad registers in order to provide an equivalent function to the legacy instruction. Once the instruction is emulated, the scratch pad registers associated with the Ring 2 instructions are restored in step 66. The system then returns to step 60A and inserts a branch at the end of the handler to return to the instruction stream.

[0039] If it is determined that the instruction is a Ring 3 category instruction, instruction code is inserted in the instruction handler 36 to save the machine state in step 60B. Subsequently in step 68, the instruction handler jumps to a higher order language program, for example C/C++, which provides an equivalent function to the legacy instruction. The higher order language by way of a compiler handles the return back to the instruction handler 36 which returns to step 70 where the machine state is restored and the system returns to the main instruction screen by inserting a return branch instruction at the end of the handler.

INSTRUCTION SPACING AND BRANCH FOLDING

[0040] As shown best in FIGS. 14 and 15, the host processor program is shown as an instruction stream, generally identified with the reference number 72, which includes Ring 0 type instructions and unconditional branches to instruction handlers 36 (FIG. 1). This configuration of placing Ring category 1-3 instructions in separate handlers space 36 from the Ring 0 space 34 and using the unconditional branches to branch to the Ring 1-3 instruction handlers space 36 (FIG. 1) is used to maintain distance between the instructions. More particularly, every program is anticipated to contain jump and/or branch instructions. If the spacing between instructions is not maintained, jumps and branches may end up at the wrong location, since many instructions operate in a relative fashion, jumping forward or backward a given distance from the current location rather than using an absolute destination address. Thus, when a 1:1 match cannot be made, a single unconditional branch is used in place of the legacy instruction as shown in FIGS. 14 and 15. The unconditional branch instruction vectors to the instruction handler space 36 (FIG. 1) to emulate the legacy instruction. When the handler completes its function, it returns to the instruction following the unconditional branch, as shown thus in FIGS. 14 and 15.

[0041] FIG. 3 illustrates an example of an conditional branch forward based upon relative addressing. More particularly, during the execution of the instruction at address 1004, the program counter points to the next instruction at address 1008. As shown in FIG. 3, the instruction BEQ is a conditional branch instruction. The conditional branch instruction, if taken, increments the current address by eight (8) to arrive at a destination address of 1010. If either of the instructions at 1008 or 100C cannot be translated 1:1 and if additional code is inserted in line, a branch would vector to an incorrect destination address. If instead an unconditional branch is used to branch to an instruction handler, for example, an unconditional branch inserted into the instruction stream 72 (FIGS. 14 and 15), the distance remains the same.

[0042] This technique relies on a feature of many RISC processors called "branch folding" which allows the unconditional branches to be automatically removed from the instruction stream by the processor with little or no loss of performance. When a branch is unconditional, it must be taken. The processor simply begins pre-fetching instructions down the new path, and little or no performance penalty is associated with the branch. On the other hand conditional branches can impose significant performance penalties because instruction pre-fetching can't occur until it is known which path to take.

REGISTER MAPPING

[0043] In accordance with one aspect of the present invention, register mapping is used to prevent conflicts between the host processor and the legacy processor. For example on a MIPS R4400 microprocessor, general register 29 is the stack pointer. However on a Power PC microprocessor, the stack pointer is register 1. In order to prevent conflicts, register fields in legacy instructions are remapped to the host processor registers. In an example where legacy instructions of a MIPS R4400 microprocessor instructions are mapped to a Power PC instructions, register 29 of the legacy instructions is remapped to register 1 of the host processor and register 1 of the legacy instructions is remapped to register 29 of the host processor. This may be done by treating the legacy register numbers as an index into an array which contains the host processor number to be used for the given legacy register number.

INSTRUCTION RENDERING

[0044] As mentioned above, legacy instructions are categorized into one of four categories: Ring 0, Ring 1, Ring 2 or Ring 3. As discussed above, the object code of the Ring 0 legacy instructions are directly mapped or converted to

LzLabs GmbH. Ex. 1004-327

a single host instruction which performs the same function. Examples of selected rendering of legacy instructions are described and illustrated herein. Mapping of Ring 0 instructions is illustrated in FIGS. 4-7. Examples of mapping of Ring 1 instructions are illustrated in FIG. 8 and 9. Examples of a Ring 2 instruction mapping are illustrated in FIGS. 11 and 12 while an example of a Ring 3 instruction mapping is illustrated in FIG. 13.

RING 0 INSTRUCTIONS

[0045] Ring 0 instructions are legacy processor instructions that can be replaced with a single host processor instruction. Binary reorganization (i.e. translating of the object code) of the legacy instructions is necessary most of the time even though the corresponding host processor instruction performs the same operation as the legacy processor instruction.

[0046] FIGS. 4-7 illustrate examples of conversions of several 32-bit MIPS R4400 RISC processor instructions to 32-bit Power PC RISC microprocessor equivalent instructions. In this example, the MIPS processor is the legacy processor while the Power PC is the host processor.

[0047] For Ring 0 instructions, the operation code (op code) is changed to match the appropriate instruction in the host processor. In some cases, a sub op code field must also be translated. Registers and immediate value fields are moved as necessary. In some cases, the host processor bit fields are set to an inactive state. Though not shown in the figures, legacy register mapping may also be done depending on the legacy/host processor combination.

[0048] FIG. 4 illustrates an example of a MIPS ADDU (add unsigned instruction) in its simplest binary representation. To convert the ADDU legacy instruction to an equivalent host ADD instruction, the op code and various sub op code fields as well as register fields are changed as illustrated in FIG. 4. The host instruction also has 2 additional single bit option fields (i.e., bits 0 and 10) that are not used for the legacy instruction. These fields are simply cleared.

[0049] Referring to FIG. 4, the ADDU legacy instruction (adds the values in the registers rs and rt and stores the results in a register rd). As shown, the legacy sub op code is 00000100001, which corresponds to bits 0-10 of the legacy instruction. The binary value 0000010001 corresponds to a decimal 33. The corresponding host instruction is an Add instruction.

[0050] In this case, the host instruction adds values in its registers rA and rB stores the results in a register rT. The format of the host processor instruction is slightly different from the format for the corresponding legacy instruction. In addition, the host instruction includes an RC (record condition) bit and an OE (overflow enable) bit as bits 0 and 10 of the 32 bit instruction. These options are not available for the legacy MIPS processor. As discussed above, these bits are simply zeroed out. After zeroing out bits 0 and 10, the equivalent sub op code of the host Add instruction is 100001010. Thus, in this case, the legacy op code 00000100001 is translated to 100001010 and, as mentioned above, and the registers are remapped. In particular, the legacy register rd is mapped to the host register rT while the legacy register rs is mapped to the host register rA and finally the legacy register rt is mapped to the host register rB. In addition, bits 26-31 of the legacy instruction are translated to the appropriate op code for the host instruction. As shown, the op code at bits 26-31, shown as 000000 is translated 011111.

[0051] FIGS. 5 and 6 illustrate other examples of direct mapping. For example, FIG. 5 illustrates direct mapping of a legacy LHU (load halfword unsigned) instruction and an SRA (shift right algebraic) instruction. In the case of the LHU instruction, immediate data is also transferred to the host processor instruction. The SRA instruction contains an immediate 5 bit shift count field. Finally FIG. 7 illustrates a legacy Sync instruction which has no argument. The corresponding Sync instruction similarly has no arguments. In this case the op code and sub op code are translated as illustrated in FIG. 7.

RING 1 INSTRUCTIONS

[0052] As mentioned above, Ring 1 category legacy instructions cannot be mapped directly to a single host processor instruction. Therefore, an unconditional branch instruction is inserted into the instruction stream 72 (FIGS. 14, 15) as discussed above. The branch vectors execution to an instruction handler that uses multiple host processor instructions to emulate the behavior of the emulated legacy instruction. Ring 1 instructions allow for a simple enough emulation that additional scratch registers are not required.

[0053] FIG. 8 illustrates an assembly language mnemonic for a legacy LB (load byte) instruction and a host instruction handler for the LB legacy instruction. The legacy LB instruction is categorized as a Ring 1 instruction because the legacy instruction also performs a sign extension which extends the sign bit in the destination register based on the sign of the byte loaded. An additional host processor instruction is required to perform the sign extension. More particularly, as illustrated in FIG. 8, five words are required to handle this instruction. The first word is the exact legacy binary instruction. The next word is a pointer to another handler for the same instruction with different operands. This is done to allow reusing a handler that has already been created. The singly linked list provides a mechanism for searching for a match. The third word starts the host processor handler of the legacy instruction. Any number of host

LzLabs GmbH. Ex. 1004-328

processor instructions may be present. Finally, a host processor sub routine return instruction is inserted to return to the main instruction stream.

[0054] FIG. 9 illustrates a legacy BEQ (branch if equal instruction) and a host instruction handler for emulating this instruction. In this case the BEQ legacy instruction is categorized as a Ring 1 instruction because the legacy BEQ instruction performs a comparison and branch all in one instruction. The host processor instructions do not allow for comparison branching in one instruction. Rather, the instructions for the host processor require that the compare must be done separately and the results stored in a condition code register. Then the conditional branch uses the stored condition code when deciding whether to branch.

[0055] In the case of the legacy target, the host handler may also incorporate the legacy instruction immediately following the branch of equal instruction since the legacy processor may execute it while calculating the destination address. The handler can accommodate this as well, although not shown in the figure. More particularly with reference to FIG. 10, a Ring 1 is shown in an instruction stream with several Ring 0 instructions. Whenever a Ring 1 instruction is executed, an unconditional branch and link instruction is used to vector execution to an instruction handler starting at the instruction following the two word header. As shown in FIG. 10, at the end of the instruction handler, execution returns to the instruction stream and specifically to the link address immediately following the branch.

RING 2 INSTRUCTION

[0056] Ring 2 instructions are similar to Ring 1 instructions except that one or more scratch registers are needed to perform the operation. This requires saving out one or more of the host processor registers and then restoring them before returning to the legacy instruction screen.

[0057] FIG. 11 is an example of a Ring 2 instruction. In particular, FIG. 11 illustrates a legacy SLT (set if less than instruction) and a host instruction handler for handling this instruction. The legacy instruction performs a compare and stores the zero for inequality and a zero for equality into a designated register. In this case the host processor, a Power PC processor, does not have a corresponding instruction. Thus, the host processor instead performs a compare and then extracts the results of the compare from a condition code register. That value must then be rotated into the correct position. These operations require a scratch register to work so one is allocated, saved out and restored when the handler is complete.

[0058] FIG. 12 illustrates another example of a legacy MTC1 instruction along with its corresponding host processor instruction handler. The MTC1 instruction is a move to co-processor 1 instruction which moves a value from a general purpose register into a floating-point register which is the legacy processor-coprocessor 1. In the case of the host microprocessor (i.e., PowerPC microprocessor), there are no instructions to move values directly from a general register to a floating-point register. Therefore, the instruction handler must use a scratch register to construct the address of a memory buffer. The general register value is stored in memory and then a floating-point load instruction is used to get it into the floating-point register as generally shown in FIG. 12.

[0059] It is anticipated that a Ring 2 instruction can be avoided if a host processor with adequate register resources is selected. In the case of a MIPS R4400 legacy microprocessor and a PowerPC host processor, each has a 32-bit general purpose register and 32-bit floating-point registers. This forces the requirement to save the registers through memory and restore them at one or more scratch registers are required.

RING 3 INSTRUCTIONS

[0060] Ring 3 instructions involve complex operations and advanced architectural features of the legacy processor that are not easily emulated using host processors or assembly language. Therefore, Ring 3 instructions are handled using a higher-order language, such as C/C++ instead. A host assembly language handler is set up for these instructions but its function is to call the higher-order language components as required.

[0061] FIG. 13A illustrates a MIPS SYSCALL (system call) instruction and its host handler. As shown, the machine's state is saved prior to making the call to the higher-level language and is returned and restored on return from the handler. The requirements of calling a function in a higher-order language are well documented by the application binary interface (ABI) document of the host processor. Such ABIs contain conventions that must be followed to ensure proper interaction of programs for any given architecture. The specific requirements. therefore, are dictated by the choice of the host processor. The handler for the Ring 3 instruction, then, is whatever code is necessary to call the higher-order language component.

[0062] FIG. 13B illustrates exemplary PowerPC assembly language and an exemplary handler for a Ring 3 CFCz instruction. The instruction handler is generated in PowerPC machine code. However, commented assembly is used for purposes of illustration.

[0063] The legacy instruction being emulated is a CFCz instruction, which moves a 32-bit value from a legacy co-processor register to a general-purpose register. The legacy microprocessor, in this case a MIPS RISC R4400 micro-

LzLabs GmbH. Ex. 1004-329

processor, allows for up to 4 coprocessors with coprocessor 1 being the floating-point unit. The instruction handler saves out the machine state, fetches the legacy instruction from the top of the handler and then extracts the three arguments consisting of the coprocessor register number, the general-purpose register number, and the coprocessor number.

```
// Save the machine state.  Registers get saved on the stack.
stwu      sp,-160(sp)     // adjust the stack pointer
mflr      r0                // get current link register
stw       r0,4(sp)          // save it on the stack
stw       r0,32(sp)         // save all PowerPC general registers to the stack
stw       r1,36(sp)         // starting at offset 32
.
stw       r30,152(sp)
stw       r31,156(sp)

// Setup to call the C/C++ handler code.  For CFCz, this involves fetching
// the legacy instruction from the top of the handler (second word, just
// before any executable code) and extracting the arguments from it.

// fetch the legacy instruction from the top of the handler so the arguments
// can be retrieved
bl        +4                // do a linking branch to get the program counter
nop                         //  into the link register (can't read it directly)
mflr      r13               // fetch the program counter from the link register
lwz       r13,-152(r13)     // load the legacy CFCz instruction

// get the first argument the contents of MIPS register rd
rlwinm    r14,r13,23,25,29 // extract the rd field and multiply by 2
addi      r14,r14,32        // form offset into stack for rd
lwzx      r3,r14,sp         // load the contents of rd from the stack

// get the second argument the z field
rlwinm    r4,r13,5,30,31    // extract the coprocessor number (0)

// Call the handler function.  The rest will be handled in C/C++ code.
bl    Execute_CFCz        // Execute_CFCz called as if it were a C function

// the return value comes back in r3 store it into the appropriate
// register on the stack in the next stage it will be placed into the
// appropriate register
rlwinm    r15,r13,18,25,29 // extract the rt field from the legacy instruction
addi      r15,r15,32        // form offset into stack for rd
stwx      r3,r15,sp         // store return value into appropriate register

// Restore the machine state.  Here all of the volatile registers are
// reloaded and the stack pointer is adjusted back.
lwz       r0,32(sp)         // restore non-volatile registers
lwz       r1,36(sp)
.
lwz       r30,152(sp)
lwz       r31,156(sp)
lwz       r0,4(sp)          // load the original link register
mtlr      r0                //  and restore it
addi      sp,sp,160         // restore the original stack pointer

blr                         // return to the instruction stream
```

[0064] In the set up section, the arguments are placed into the host, i.e. PowerPC, registers r3 and r4. This complies with the PowerPC application binary interface specification for calling a C function. The call to Execute_CFCz, is a designated C function, used for handling the CFCz instruction. An example for that function follows.

LzLabs GmbH. Ex. 1004-330

```
#include <CoprocessorControlManager.h>

// there is a pointer to the C++ class that manages the interaction with
// up to 4 MIPS coprocessors
extern CoprocessorControlManager *cp;


// we must give this function C style language naming so that it can be
// called from assembly/machine language code
extern ``C''
{
unsigned long Execute__CFC (unsigned long rd, unsigned long z);
};

// this C function calls the C++ object that manages coprocessor
// operations: for the CTCz instruction, only a register read or write
// is required but the consequences of a read or write are dependent on
// the nature of the coprocessor
unsigned long Execute__CFC (unsigned long rd, // passed in PowerPC register r3
                            unsigned long z)  // passed in PowerPC register r4
{
return cp->readReg (z, rd);
}
```

[0065] The following is an exemplary class specification for a C++ class that handles coprocessor operations.

```
// this is an example class specification for a coprocessor control
class CoprocessorControlManager
{
   public:

   // the public constructor initializes the object as necessary
   CoprocessorControlManager ();

   // the destructor performs cleanup when the class is destroyed
   ~CoprocessorControlManager ();

   // this function reads a coprocessor register and returns it value
   void readReg (unsigned long z, unsigned long rd);
};
```

[0066] The following is an exemplary implementation for the coprocessor read register function defined in the class above.

```
// this is an exemplary implementation of the method responsible for
// performing reads from coprocessor control registers
unsigned long CoprocessorControlManager ::readReg (unsigned long z,
                                                   unsigned long rd)
{
   // which coprocessor?
   switch (z)
   {
      // the floating point unit
      case 0:
      {
```


LzLabs GmbH. Ex. 1004-331

```
            // which register to read
            switch rd:
               // this is the coprocessor implementation and revision register which
               // contains information about the floating point unit (it is read only)
               case 0:
                  // return the version number
                  return 1;

               // this is the floating point control and status register
               case 1:
                  // rounding, trap enables, and flags are stored here
                  // CODE OMITTED
                  break;

               // all other regisers are reserved for this coprocessor so just
               // return 0;
              default:
                  return 0;
            } // end case

           // no other cases handled by default
           default:
             return 0;
         } // end switch

        // CODE OMITTED
    }
```

LOOK-UP TABLES

[0067] As mentioned above, Tables 1 and 2 represent the look-up tables used by the crack map object 32 for identifying legacy instructions and identifying the specific ring category for each of the legacy instructions for legacy instruction of a MIPS R4400 RISC legacy microprocessor.

TABLE 1
CLASSIFICATION OF MIPS 32 BIT INTEGER INSTRUCTIONS

Instruction	Opcode	Function	Type	Ring	RS	RT	RD	Immediate	Shamt
ADD	0	32	R	1					0
ADDI	8		I	0					
ADDIU	9		I	0					
ADDU	0	33	R	0					0
AND	0	36	R	0					0
ANDI	12		I	0					
BC0F	16		I	3	8	0			
BC0FL	16		I	3	8	2			
BC0T	16		I	3	8	1			
BC0TL	16		I	3	8	3			
BC1F	17		I	3	8	0			
BC1FL	17		I	3	8	2			
BC1T	17		I	3	8	1			

LzLabs GmbH. Ex. 1004-332

Instruction	Opcode	Function	Type	Ring	RS		RT	RD	Immediate	Shamt
BC1TL	17		I	3	8		3			0
BC2F	18		I	3	8		0			
BC2FL	18		I	3	8		2			
BC2T	18		I	3	8		1			0
BC2TL	18		I	3	8		3			
BEQ	8		I	2						
BEQL	20		I	2						
BGEZ	1		I	2			1			
BGEZAL	1		I	2			17			
BGEZALL	1		I	2			19			
BGEZL	1		I	2			3			
BGTZ	7		I	2			0			
BGTZL	23		I	2			0			
BLEZ	6		I	2			0			
BLEZL	22		I	2			0			
BLTZ	1		I	2			0			
BLTZAL	1		I	2			16			
BLTZALL	1		I	2			18			
BLTZL	1		I	2			2			
BNE	5		I	2						
BNEL	21		I	2						
BREAK	0	13	I	0						
CACHE	47		I	3						
CFC0	16	0	R	0	2					0
CFC1	17	0	R	3	2					0
CFC2	18	0	R	3	2					0
COP0	16		J	3					upper bit set	0
COP1	17		J	3					upper bit set	0
COP2	18		J	3					upper bit set	
CTC0	16	0	R	0	6					0
CTC1	17	0	R	3	6					
CTC2	18	0	R	3	6					0
CTC3	19	0	R	3	6					
DIV	0	26	R	2				0		0
DIVU	0	27	R	2				0		0
ERET	16	24	R	3	16		0	0		0
J	2		J	0						
JAL	3		J	2						
JALR	0	9	R	2			0			0
JR	0	8	R	1			0	0		0
LB	32		I	0	base				offset	
LBU	36		I	0	base				offset	
LH	33		I	0	base				offset	
LHU	37		I	0	base				offset	
LL	48		I	3	base				offset	
LUI	15		I	0		0				

LzLabs GmbH. Ex. 1004-333

Instruction	Opcode	Function	Type	Ring	RS	RT	RD	Immediate	Shamt
LW	35		I	0	base			offset	
LWC1	49		I	3	base			offset	
LWC2	50		I	3	base			offset	
LWL	34		I	0	base			offset	
LWR	38		I	0	base			offset	
LWU	39		I	0	base			offset	
MFC1	17	0	R	3	0				0
MFC2	18	0	R	3	0				
MFHI	0	16	R	1	0	0			0
MFLO	0	18	R	1	0	0			0
MTC1	17	0	R	3	4				
MTC2	18	0	R	3	4				
MTHI	0	17	R	2		0	0		0
MTLO	0	19	R	2		0	0		0
MULT	0	24	R	2			0		0
MULTU	0	25	R	3			0		0
NOR	0	39	R	0					0
OR	0	37	R	0					0
ORI	13		I	0					
SB	40		I	0					
SC	56		I	3	base			offset	
SH	41		I	0	base			offset	
SLL	0	0	R	0	0				
SLLV	0	4	R	0					0
SLT	0	42	R	2					0
SLTI	10		I	2					
SLTIU	11		I	2					
SLTU	0	43	R	2					0
SRA	0	3	R	0	0				
SRAV	0	7	R	0					0
SRL	0	2	R	0	0				
SRLV	0	6	R	0					0
SUB	0	34	R	1					0
SUBU	0	35	R	0					0
SW	43		I	0	base			offset	
SWC1	57		I	3	base			offset	
SWC2	58		I	3	base			offset	
SWL	42		I	0	base			offset	
SWR	46		I	0	base			offset	
SYNC	0	15	R	0	0	0	0		0
SYSCALL	0	12	R	3	code	code	code		code
TEQ	0	52	R	3			code		code
TEQI	1		I	3		12			
TGE	0	48	R	3					
TGEI	1		I	3		8			
TGEIU	1		I	3		9			

LzLabs GmbH. Ex. 1004-334

Instruction	Opcode	Function	Type	Ring	RS	RT	RD	Immediate	Shamt
TGEU	0	49	R	3			code		code
TLBP	16	8	R	3	16	0	0		0
TLBR	16	1	R	3	16	0	0		0
TLBWI	16	2	R	3	16	0	0		0
TLBWR	16	6	R	3	16	0	0		0
TLT	0	50	R	3			code		code
TLTI	1		I	3		10			
TLTIU	1		I	3		11			
TLTU	0	51	R	3			code		code
TNE	0	54	R	3			code		code
TNEI	1		I	3		14			
XOR	0	38	R	0					0
XORI	14		I	0					

TABLE 2
CLASSIFICATION OF MIPS 32 BIT FLOATING POINT INSTRUCTIONS

Instruction	Opcode	Function	Type	Ring	FT/Base	FS	FD	Offset	Format
ABS.fmt	17	5	R	0	0				
ADD.fmt	17	0	R	0					
C.cond.fmt	17	FC/cond	R	0			0		
CEIL.L.fmt	17	10	R	2	0				
CEIL.W.fmt	17	14	R	2	0				
CVT.D.fmt	17	33	R	2	0				
CVT.L.fmt	17	37	R	2	0				
CVT.S.fmt	17	32	R	2	0				
CVT.W.fmt	17	36	R	2	0				
DIV.fmt	17	3	R	0					
DMFC1	17	0	R	3			0		1
FLOOR.L.fmt	17	11	R	2	0				
FLOOR.W.fmt	17	15	R	2	0				
MOV.fmt	17	6	R	0	0				
MUL.fmt	17	2	R	0					
NEG.fmt	17	7	R	0	0				
ROUND.L.fmt	17	8	R	0	0				
ROUND.W.fmt	17	12	R	0	0				
SQRT.fmt	17	4	R	2	0				
TRUNC.L.fmt	17	9	R	0	0				
TRUNC.W.fmt	17	13	R	0	0				

[0068] As shown, Table 1 includes an exemplary ten fields. The first field is ider tified as an instruction field and identifies the assembly language "mnemonic" for each of the legacy instructions. The op-code field represents the op-code of the legacy instruction. The function field is used for register type instructions for both integer instruction format and a floating point instruction format instructions. For example, FIGS. 16-20 illustrate the integer and floating point binary instruction formats for a MIPS microprocessor. For register type instructions, the function field is used as a sub-op code and translates to a function field for the host processor as described, for example, in connection with FIG. 4.

LzLabs GmbH. Ex. 1004-335

example, bits 0 through 10 (See FIG. 4.). Thus, as shown, for the ADDU instruction, the decimal value of sub-op-code (i.e., bits 0 through 10) is 33. This value 33, in the case of the ADDU instruction is thus identified in the function field. The type field relates to the instruction type, for example, as illustrated in FIGS. 16-20. Tables 1 and 2 also include a ring field. The ring field corresponds to the Ring category as discussed above. The RS, RT and RD fields relate to the legacy registers and identify the register numbers associated with the registers rs, rt and rd. The immediate field is used for immediate type instructions, for example, as illustrated in FIG. 16. Finally, the "Shamt" field relates to the shift amount for example for legacy instructions as illustrated in FIG. 17.

[0069] Table 2 lists the 32 bit floating instructions for a MIPS R4400 RISC microprocessor. The "instruction," "op code," "function," "type" "ring" fields are the same as Table 1. The "FT/base" and "offset" fields are used for immediate type floating point instructions, for example, as shown in FIG. 19. The FT, FS and FD fields refer to register numbers for floating print instructions.

[0070] The format field indicates the size and format of the floating point value. Table 3 illustrates valid format field codes for designating single and double precision IEEE 754 floating point values as well as single and long word fixed point values. Other values of the format field are possible but used for decoding instructions only.

Format Code (fmt Field)	Mnemonic	Size	Format
16	S	Single	Binary floating-point
17	D	Double	Binary floating-point
18	-	-	-
19	-	-	-
20	W	Single	32-bit Binary Fixed-Point
21	L	Longword	64-bit Binary Fixed-Point
22-31	-	-	-

Claims

1. A method for emulating instructions of a legacy microprocessor with a host microprocessor comprising the steps of:

   (a) converting the object code of one or more legacy instructions to the object code of one or more host processor instructions with equivalent functions defining Ring 0 instructions;
   (b) replacing legacy instructions with said Ring 0 instructions.

2. The method as recited in claim 1,
   further including the step of re-mapping registers associated with said legacy instructions to corresponding host registers.

3. The method as recited in claim 1 or 2,
   further including the step of utilizing multiple host instructions which together perform the same function as a single legacy instruction defining Ring 1 instructions to replace one or more of said legacy instructions.

4. The method as recited in claim 3,
   further including the step of utilizing multiple host instructions which together perform the same function as a single scratch pad register but require the use of one or more scratch pad registers defining Ring 2 instructions to replace one or more of said legacy instructions.

5. The method as recited in claim 4,
   further including the step of emulating legacy instructions which can only be emulated with a high level language defining a Ring 3 instruction.

6. The method as recited in claim 3, 4 or 5,
   further including providing an instruction handler for said Ring 1 instructions.

LzLabs GmbH. Ex. 1004-336

7. The method as recited in claim 4, 5 or 6,
further including providing an instruction handler for said Ring 2 instructions.

8. The method as recited in claim 5, 6 or 7,
further including providing an instruction handler for said Ring 3 instructions.

9. The method as recited in claim 6, 7 or 8,
wherein said instruction handlers are mapped to separate address space from said Ring 0 instructions.

10. The method as recited in claim 9,
wherein said emulated instructions form an instruction stream containing Ring 0 instructions and unconditional branch instructions to instruction handlers for said Ring 1, Ring 2 and Ring 3 instructions.

LzLabs GmbH. Ex. 1004-337

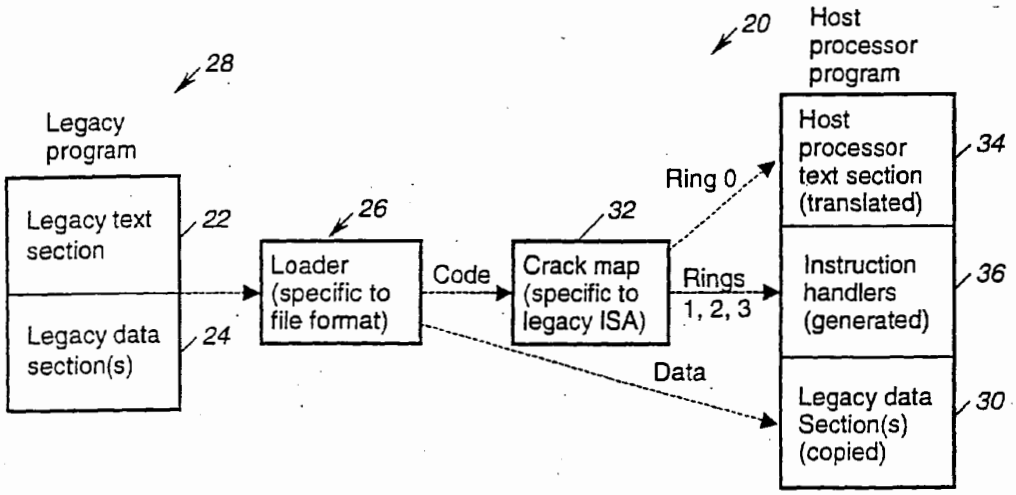


Figure 1

LzLabs GmbH. Ex. 1004-338

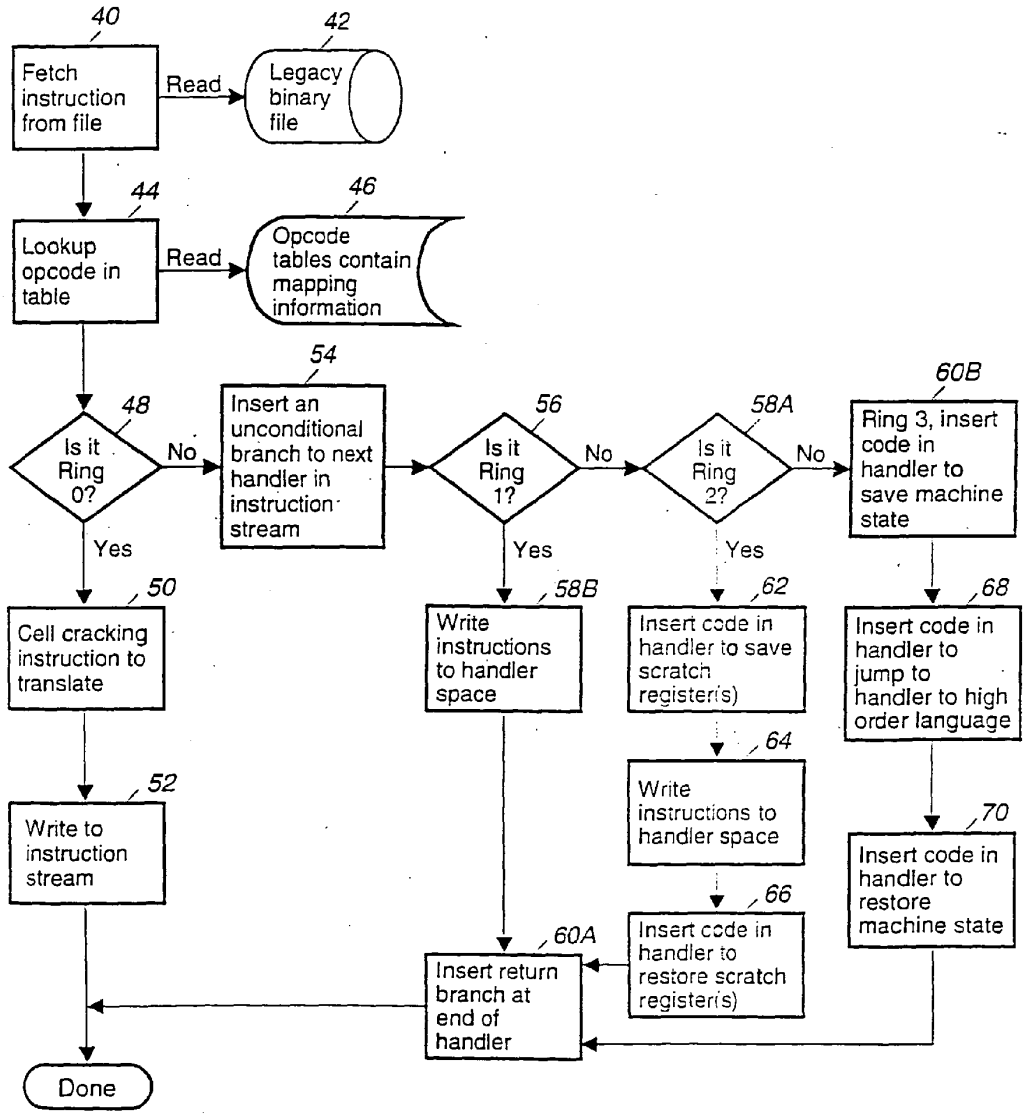


Figure 2

LzLabs GmbH. Ex. 1004-339

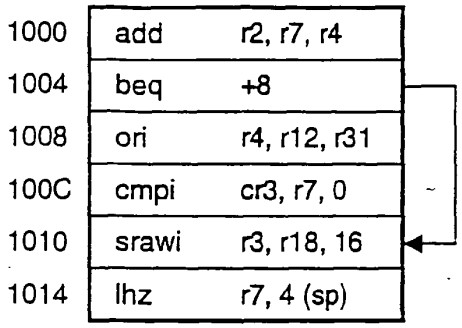


Figure 3

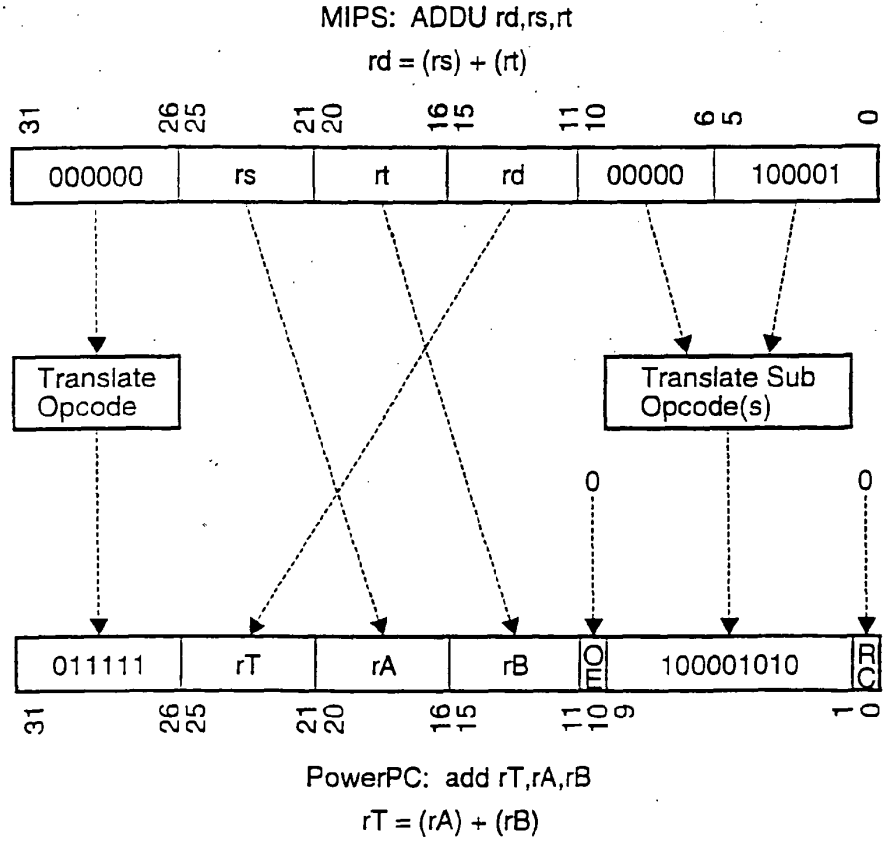


Figure 4

LzLabs GmbH. Ex. 1004-340

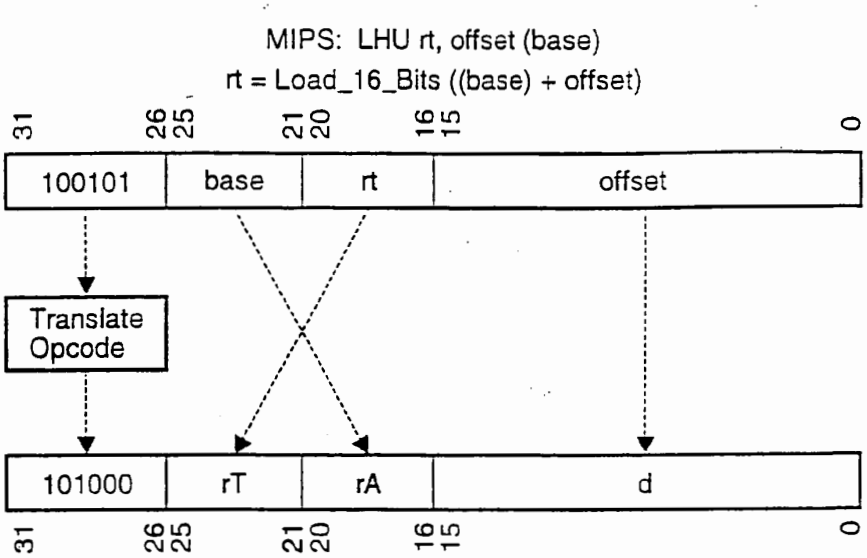


Figure 5

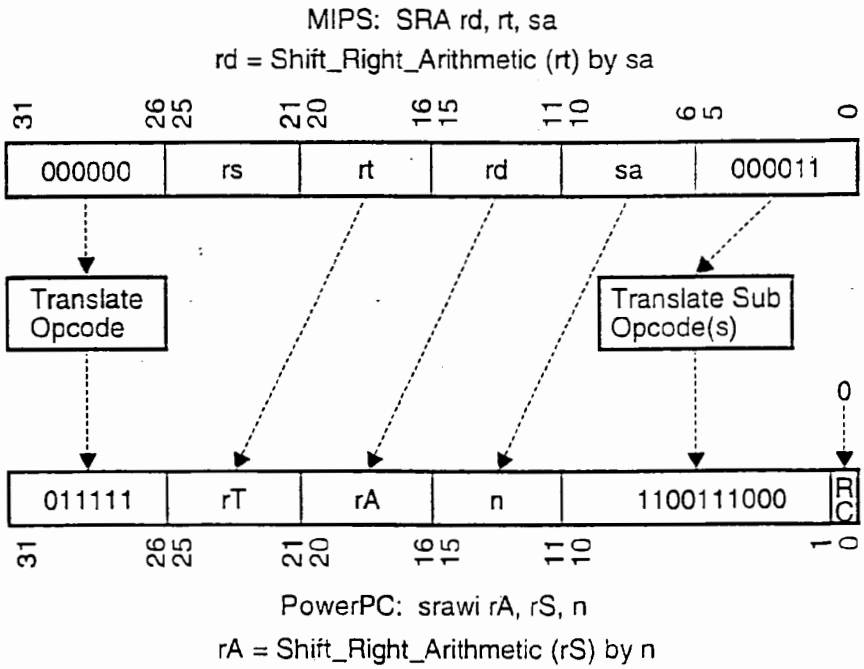


Figure 6

LzLabs GmbH. Ex. 1004-341

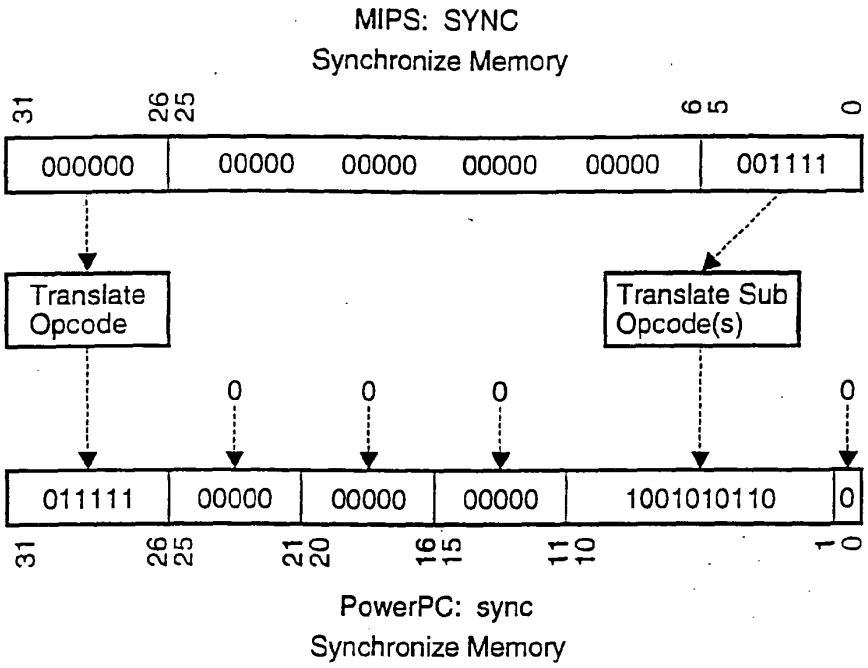


Figure 7

MIPS Instruction

LB rt, offset (base)	Load byte and sign extend

PowerPC Handler

Legacy binary	Binary of legacy instruction
Next pointer	Pointer to next of this type
lbz rT, d(rA)	Load byte and zero extend (execution starts here)
extsb rT	Sign extend loaded byte
blr	Branch back to instruction stream

Figure 8

LzLabs GmbH. Ex. 1004-342

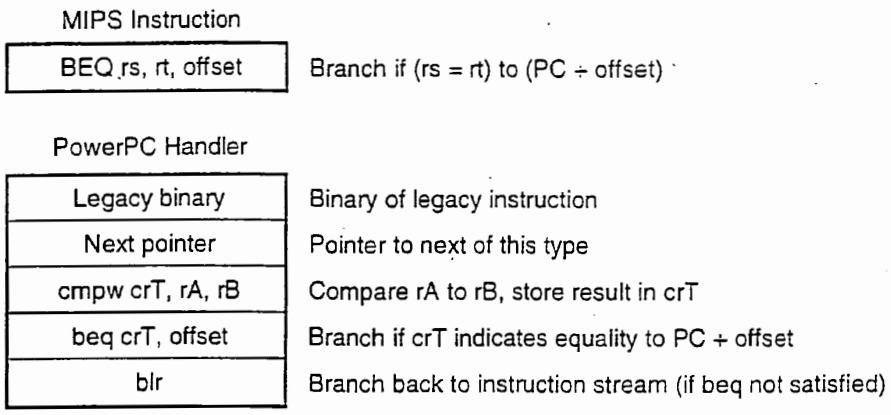


Figure 9

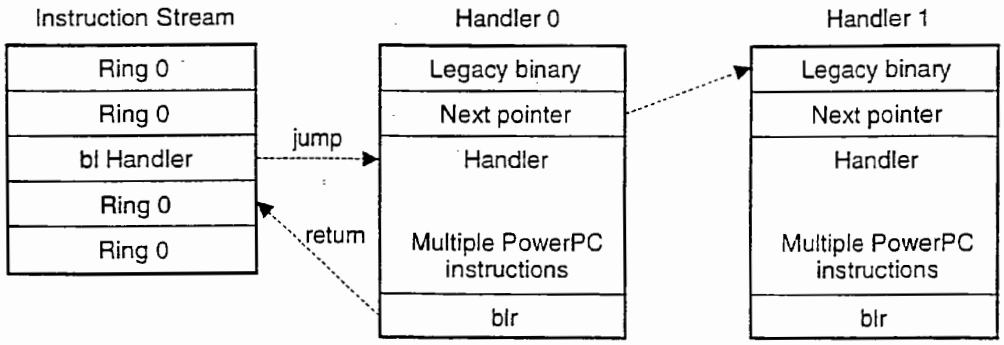


Figure 10

LzLabs GmbH. Ex. 1004-343

MIPS Instruction	
SLT rd, rs, rt	Set rd to 1 if rs < rt, else set rd to 0

PowerPC Handler	
Legacy binary	Binary of legacy instruction
Next pointer	Pointer to next of this type
stw rscratch, 0 (sp)	Store scratch register to stack
cmpw crT, rA, rB	Compare the two values
mfcr rscratch	Get compare result into scratch register
rlwinm rS, rscratch, 1, 30, 31	Rotate appropriate bit into rd (PowerPC rs)
lwz rscratch, 0 (sp)	Restore scratch register from stack
blr	Branch back to instruction stream

Figure 11

MIPS Instruction	
MTC1 rt, fs	Move general register rt to floating point register fs

PowerPC Handler	
Legacy Binary	Binary of legacy instruction
Next Pointer	Pointer to next of this type
stw rscratch, 0 (sp)	Store scratch register to stack
lui rscratch, buf@h	Load upper 16-bits of memory buffer address into rscratch
ori rscratch, rscratch, buf@1	Or in the lower 16-bits of memory buffer address
stw rS, 0 (rscratch)	Store rt (PowerPC rS) to memory buffer
lfs fT, 0 (scratch)	Load fs (PowerPC fT) from memory buffer
lwz rscratch, 0 (sp)	Restore scratch register from stack
blr	Branch back to instruction stream

Figure 12

LzLabs GmbH. Ex. 1004-344

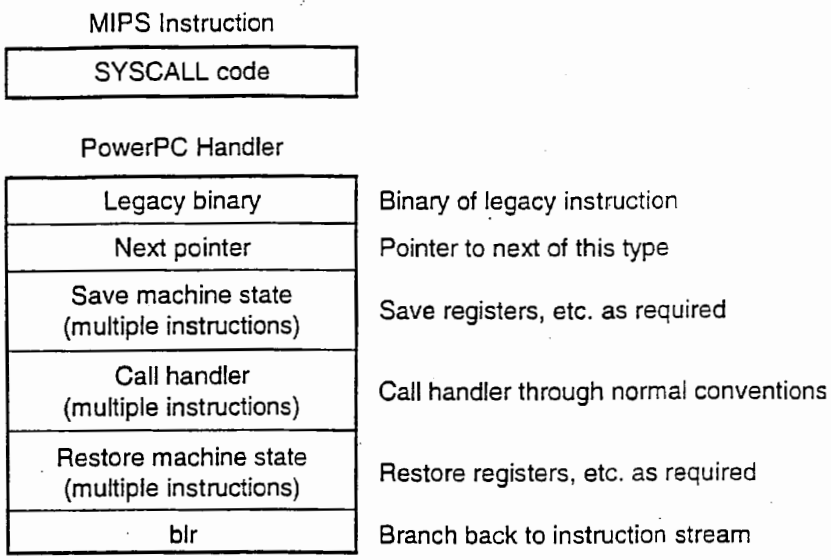


Figure 13A

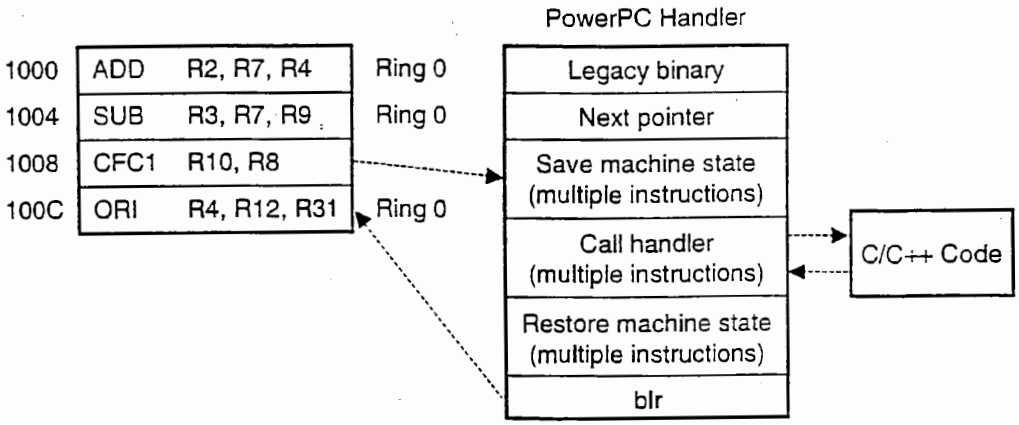


Figure 13B

LzLabs GmbH. Ex. 1004-345

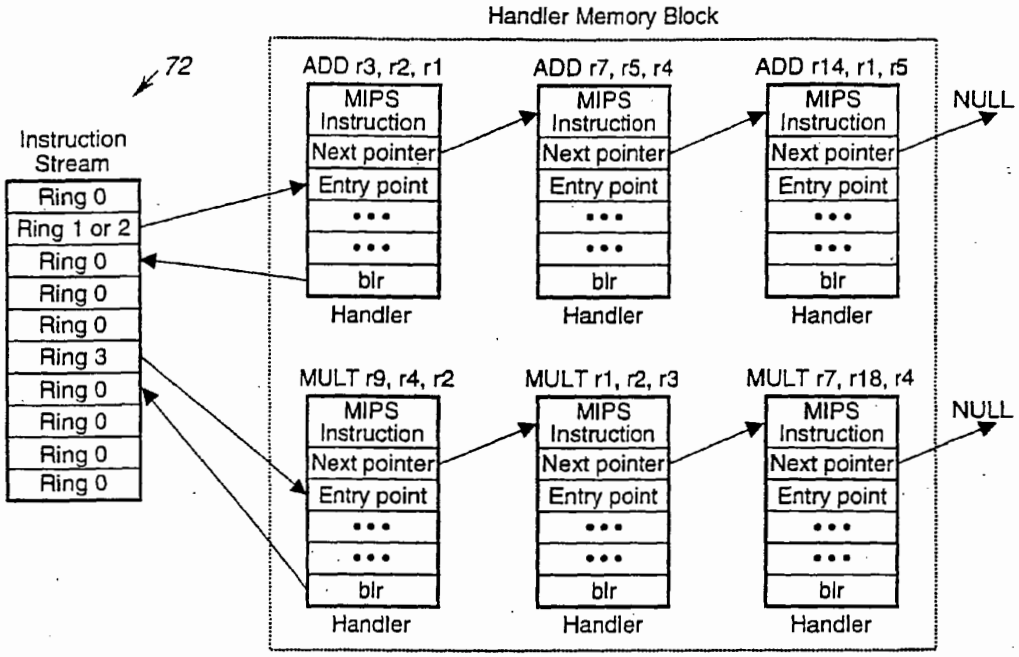


Figure 14

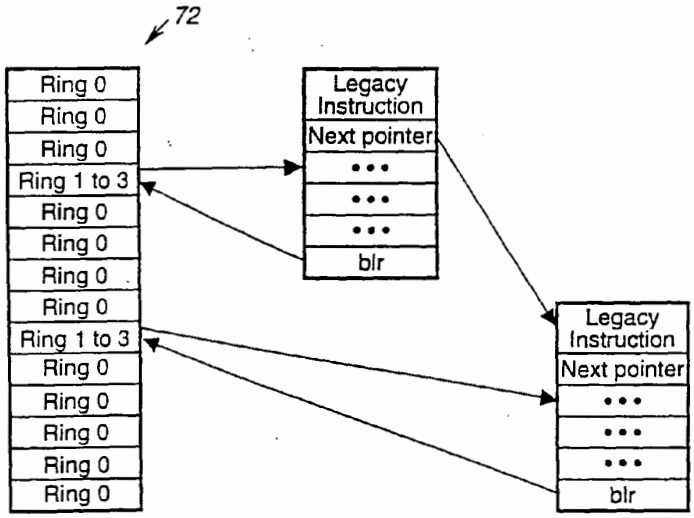
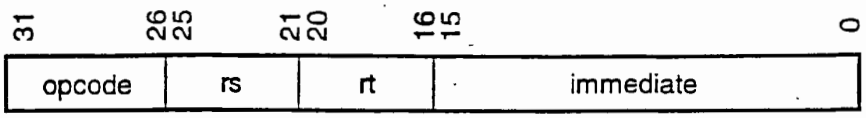


Figure 15

LzLabs GmbH. Ex. 1004-346

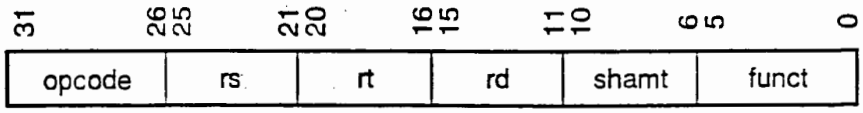
MIPS I-Type (Immediate)
Integer Instruction Format



opcode - 6-bit operation code
rs - 5-bit source register number
rt - 5-bit target (source or destination) or branch condition register
immediate - 16-bit immediate value used for branch displacement or address indexing

Figure 16

MIPS R-Type (Register)
Integer Instruction Format

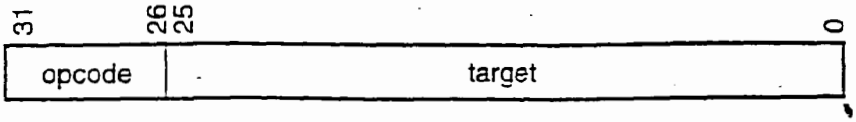


opcode - 6-bit operation code
rs - 5-bit source register number
rt - 5-bit target (source or destination) or branch condition register
rd - 5-bit destination register number
shamt - 5-bit shift amount (when applicable)
funct - 6-bit function field

Figure 17

LzLabs GmbH. Ex. 1004-347

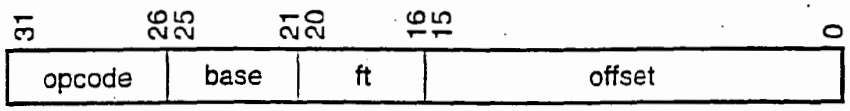
MIPS J-Type (Jump)
Integer Instruction Format



opcode - 6-bit operation code
target - 26-bit target address

Figure 18

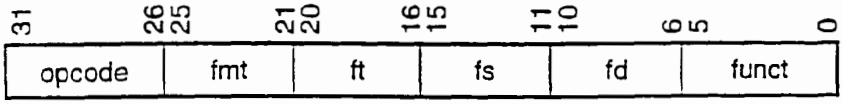
MIPS I-Type (Immediate)
Floating-Point Instruction Format



opcode - 6-bit operation code
base - 5-bit base register number
ft - 5-bit source register number
offset - 16-bit signed immediate offset value from base

Figure 19

MIPS R-Type (Register)
Floating-Point Instruction Format



opcode - 6-bit operation code
fmt - 5-bit format specifier
ft - 5-bit source register number
fs - 5-bit source register number
fd - 5-bit destination register number
func - 6-bit function field

Figure 20

LzLabs GmbH. Ex. 1004-348

PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : G06F 9/455 — A1

(11) International Publication Number: **WO 97/50031**

(43) International Publication Date: 31 December 1997 (31.12.97)

(21) International Application Number: PCT/US97/10341

(22) International Filing Date: 17 June 1997 (17.06.97)

(30) Priority Data:
08/669,772 27 June 1996 (27.06.96) US

(71) Applicant: INTEL CORPORATION [US/US]; 2200 Mission College Boulevard, Santa Clara, CA 95052 (US).

(72) Inventors: FARBER, Yaron; 26 Derech Ha-Yam, 34741 Haifa (IL). LEVHARI, Yossi; 11 Morad HaNachal Street, P.O. Box 238, 40295 Hofit (IL). BARAZ, Leonid; 81 HaOren Street, 28000 Kiryat Ata (IL). LADIRAY, Gallia; 12 Dan Street, 93509 Jerusalem (IL).

(74) Agents: TAYLOR, Edwin, H. et al.; Blakely, Sokoloff, Taylor & Zafman LLP, 7th floor, 12400 Wilshire Boulevard, Los Angeles, CA 90025 (US).

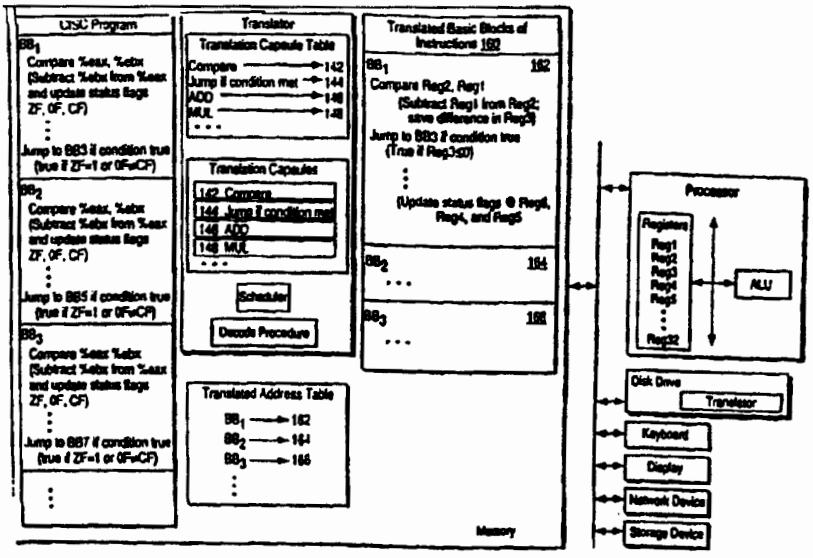
(81) Designated States: AL, AM, AT, AT (Utility model), AU (Petty patent), AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, CZ (Utility model), DE, DE (Utility model), DK, DK (Utility model), EE, EE (Utility model), ES, FI, FI (Utility model), GB, GE, GH, HU, IL, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SK (Utility model), SL, TJ, TM, TR, TT, UA, UG, UZ, VN, YU, ZW, ARIPO patent (GH, KE, LS, MW, SD, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).

Published

With international search report.

Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(54) Title: METHOD FOR INCREASING PERFORMANCE OF BINARY TRANSLATED CONDITIONAL INSTRUCTIONS



(57) **Abstract**

A method for increasing the performance of binary translated conditional instructions (162). According to one embodiment of the invention, a conditional instruction compatible with the first ISA is decoded (BB1). The condition of the conditional instruction is dependent on at least one status flag (ZF=1). The conditional instruction is translated to be compatible with a second ISA (144), wherein the condition of the conditional instructions is altered to be dependent on a previously computed difference between two values (162), the difference residing in a memory location (REG3).

LzLabs GmbH. Ex. 1004-349

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece			TR	Turkey
BG	Bulgaria	HU	Hungary	ML	Mali	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MN	Mongolia	UA	Ukraine
BR	Brazil	IL	Israel	MR	Mauritania	UG	Uganda
BY	Belarus	IS	Iceland	MW	Malawi	US	United States of America
CA	Canada	IT	Italy	MX	Mexico	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NE	Niger	VN	Viet Nam
CG	Congo	KE	Kenya	NL	Netherlands	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NO	Norway	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	NZ	New Zealand		
CM	Cameroon			PL	Poland		
CN	China	KR	Republic of Korea	PT	Portugal		
CU	Cuba	KZ	Kazakstan	RO	Romania		
CZ	Czech Republic	LC	Saint Lucia	RU	Russian Federation		
DE	Germany	LI	Liechtenstein	SD	Sudan		
DK	Denmark	LK	Sri Lanka	SE	Sweden		
EE	Estonia	LR	Liberia	SG	Singapore		

LzLabs GmbH. Ex. 1004-350

METHOD FOR INCREASING PERFORMANCE OF BINARY TRANSLATED CONDITIONAL INSTRUCTIONS

FIELD OF THE INVENTION

The invention relates to computer systems, and, in particular, to methods for increasing the performance of certain types of instructions.

BACKGROUND OF THE INVENTION

Most often, computer programs are initially written in high-level program statements. In order to be executed by a computer, the program statements are compiled into machine instructions that a microprocessor can recognize and execute. The machine instructions are selected from a set of machine instructions unique to a particular Instruction Set Architecture (ISA).

In the field of ISAs, there have arisen two fundamentally different types of instruction sets: the Complex Instruction Set Computer architecture (CISC) and the Reduced Instruction Set Computer architecture (RISC). The CISC architecture utilizes an instruction set that allows for complicated and flexible ways of calculating such elements as memory addresses. One of the defining characteristics of the CISC architecture is the use of variable length instructions, which adds additional complexity to the decoding hardware. Although very powerfully, all the complexity of the CISC instruction set usually requires additional clock cycles to execute each machine instruction.

A RISC processor, on the other hand, is designed to operate more efficiently by processing a relatively smaller set of instructions. The RISC design is based on the premise that most of the instructions a computer decodes and executes are simple. As a result, RISC architecture limits the number of instructions that are built into the microprocessor, and optimizes each so it can be carried out very rapidly, usually within a single clock cycle.

It is possible for computer program statements that have been compiled into machine instructions for a CISC processor to be translated in order to be executed on a

LzLabs GmbH. Ex. 1004-351

RISC processor. This translation process is normally carried out by a binary translator, which typically consist of a set of instructions stored in memory.

The RISC machine executes the translated instructions by using the RISC instruction set to emulate how the untranslated instructions would be executed on a CISC processor. Practitioners in the art will understand, that when a RISC processor emulates a CISC program, many of the inherent performance advantages of the RISC processor are missed.

Consider, for example, the conditional instruction. A conditional instruction performs an operation, such as transferring the flow of execution from one instruction to another, when a specified condition is true. If the condition is false, the conditional instruction is treated as a no-operation (i.e., an instruction having no effect on the machine's current architectural state).

A conditional instruction is typically based on the results of a previously executed arithmetic instruction, such as the compare instruction. A compare instruction, when executed on a CISC processor, compares two values by subtracting one value from a second value and updates a set of status flags based on the difference of the two values. The actual difference generated by the comparison, however, is not saved in a memory location (i.e., is architecturally transparent).

The status flags are typically represented by one-bit fields in a dedicated register. In one prior art CISC ISA, the status flags include a Zero Flag (ZF) indicating the difference is 0, a Sign Flag (SF) representing the most significant bit of the difference, and an Overflow Flag (OF) indicating the difference is too large or too small to be represented in the number of bits allocated to store the result.

The conditional instruction, when executed on a CISC processor, evaluates a logical combination of the status flags in order to determine whether the condition of the conditional instruction is met. In one example, the conditional instruction evaluates the complex expression (ZF = 1 or SF != OF).

The RISC microprocessor, however, typically has no native support for setting status flags (i.e., dedicated registers). Therefore, a RISC processor emulates a CISC processor executing a compare instruction by allocating bit fields within a general purpose register for the setting of status flags.

LzLabs GmbH. Ex. 1004-352

To emulate a conditional instruction evaluating the set of status flags, the RISC processor employs a series of instructions provided in the RISC instruction set. Use of the series of RISC instructions, however, misses the performance advantage of the RISC processors. Typically, the instruction set of a RISC processor includes one conditional instruction that efficiently evaluates a single value generated by a previously executed compare instruction.

Therefore, it can be appreciated that there is a need for a method of translating conditional instructions that are originally dependent on a logical combination of status flags, to be dependent on an actual difference generated by a previously executed compare instruction.

SUMMARY AND OBJECTS OF THE INVENTION

The present invention provides a method for increasing the performance of binary translated conditional instructions. According to one embodiment of the invention, a conditional instruction compatible with a first ISA is decoded. The condition of the conditional instruction is dependent on at least one status flag.

The conditional instruction is then translated to be compatible with a second ISA, wherein the condition of the conditional instruction is altered to be dependent on a previously computed difference between two values.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention is illustrated by way of example and not limitation in the figures of the accompanying drawings, in which like references indicate similar elements, and in which:

Figure 1 illustrates a computer system capable of implementing one embodiment of the present invention.

Figure 2 is a flow diagram describing the steps in one embodiment of the present invention.

LzLabs GmbH. Ex. 1004-353

DETAILED DESCRIPTION

A method is described for increasing the performance of binary translated conditional instructions executed on a computer system that has no native support for setting status flags.

The present invention translates instructions from a source architecture instruction set that includes support for status flags, to a target architecture instruction set that has no native support for status flags. For example, the source architecture instruction set may be the instruction set of a CISC, Very Long Instruction Word computer (VLIW), or a RISC architecture having native support for status flags, and the target architecture instruction set may be the instruction set of a RISC, or other computer architecture that has no support for status flags.

Each conditional instruction translated by the present invention is typically included in a separate basic block of instructions (BB). The basic block of instructions are translated one at a time.

A basic block of instructions typically includes a contiguous sequence of non-branch machine-instructions ending with a conditional branch instruction. Most computer programs, such as the CISC program of Figure 1, consists of multiple basic blocks of instructions stored in a physical static sequence (e.g., BB_1, BB_2, BB_3 . . .). The basic blocks of the CISC program shown in Figure 1 are instructions compiled to be executed on a CISC program.

The instructions of each basic block of instructions in the CISC program are translated sequentially by the translator shown in the memory of Figure 1. In alternative embodiments, the translator of the present invention may be stored on other computer-readable mediums, including magnetic and optical disk, and other random-access memory devices, accessible via the disk drive shown in Figure 1.

During the translation process each instruction is first decoded into an opcode and operands by the decode procedure shown stored in the translator of Figure 1. A decoded instruction is then translated by one of the translation capsules shown in the translator.

A hash value of a decoded instructions opcode is used to index the Translation Capsule Table shown in the translator. The table in turn provides the address of a

LzLabs GmbH. Ex. 1004-354

translation capsule corresponding to the opcode of the decoded instruction. For example, the opcode for the compare instruction may hash to an entry containing the address value 142 for the compare instruction's translation capsule, as shown in Figure 1.

The translation capsule that corresponds to a decoded instruction's opcode provides a sequence of instructions from the target architecture instruction set in place of the decoded instruction's opcode. In addition, the capsule function translates the memory addresses of the decoded instruction's operands to be compatible with the memory addresses of the target architecture.

CISC program instructions translated by the translator are stored in the area of memory allocated for storing Translated Basic Blocks of instructions 160 shown in Figure 1. These instructions are compatible with the target architecture instruction set.

Figure 2 is a flow diagram describing the steps performed to increase the performance of translated conditional instructions according to one embodiment of the present invention. In block 202, a program counter identifies a basic block of untranslated instructions.

In block 204, a first instruction from the basic block of instructions is decoded by the translator into an opcode and operands. The opcode indicates an operation to be performed by the instruction, and the operands identify a register, cache, or memory location storing data to be operated on by the opcode.

In one embodiment of the present invention, the instruction is decoded by a procedure that includes a separate sequence of instructions stored in the translator. In alternative embodiments, the instructions can be decoded by a procedure stored in memory outside the translator or provided in a hardware component.

In block 206, the translator selects a translation capsule function that corresponds to the opcode of the decoded instruction by hashing the opcode to get an index value from the Translation Capsule Table as described above. The capsule function is selected from a set of translation capsule functions stored in the translator. Each capsule function provides a separate sequence of instructions in place of the decoded instructions opcode in order to emulate the execution the decoded instruction on the target computer architecture. The instructions provided from the capsule function are selected from the instruction set of the target architecture.

LzLabs GmbH. Ex. 1004-355

In addition, the capsule functions translate the memory addresses of the decoded instruction's operands to be compatible with the memory addresses of the target architecture. The translation may consist of adding the decoded instructions operands to an offset base address of the target architecture.

In one embodiment, the translation capsules are indexed in a table by a hash value of the opcodes from the source computer architecture instruction set (e.g., the 12 least significant bits of each opcode). The hash value of the decoded instructions opcode is then used to determine the location of the corresponding translation capsule in the table. Each entry in the table includes the address of the translation capsule corresponding to the opcode, and may include other information regarding the translation capsule, such as whether the opcode takes memory operands.

In block 208, the selected translation capsule function is called and receives the decoded instructions operands and opcode as its parameters. In response, the capsule provides a series of instructions in place of the decoded instructions opcode, translates the memory addresses of the decoded instructions operands, and stores the results in an area of memory allocated to store translated basic blocks of instructions.

If the instruction being translated is a compare instruction, in block 210 the respective capsule function provides a series of instructions that generate a difference between two values and save the difference in a memory location. The memory location may include a general register or an address space of a cache on a processor, for when the translation is performed dynamically (i.e., during execution of a program), or the memory location may be an address space on a disk, for when the translation is performed statically (i.e., when program is not being executed).

For example, a compare instruction compiled to be executed on the source architecture may include the operations of (subtract register %ebx from register %eax and update status flags ZF, 0F, and CF), as shown in BB1 of the CISC program in Figure 1. On the other hand, a compare instruction translated by the present invention may include the operations of (subtract reg1 from reg2 and save difference in reg3), as shown in translated basic block of instructions 162 in Figure 1.

In one embodiment, an additional set of instructions is provided by the respective capsule function to emulate setting the status flags as part of the translated compare instruction. The flag updating operations are included because subsequent

LzLabs GmbH. Ex. 1004-356

instructions in the program may depend on the settings of the flags. In one embodiment, the flags are emulated by allocating bit fields within a general purpose register.

In alternative embodiments, other arithmetic instructions that update status flags based on the results of a data operation may be translated in a similar manner.

If the instruction being translated is a conditional instruction, in block 212 the respective capsule function provides a series of instructions in order to have the condition of the translated conditional operation depend directly on the difference generated by the most recently executed translated compare instruction. The instructions provided by the function reference the memory location where the difference generated by the most recently executed compare instruction has been saved. No status flag evaluating instructions, however, are provided.

For example, the condition of a conditional instruction compiled to be executed on the source architecture may include the operations of (true if ZF =1 or, OF != CF), as shown in BB1 of the CISC program in Figure 1. On the other hand, the condition of a conditional instruction translated by the present invention may include the operation (true if reg3 $\leq$ 0), as shown in translated basic block of instructions 162 in Figure 1.

Typically, computer architectures that have no native support for status flags provide a conditional instruction, in their instruction set, that depends on a single value stored in a register. Therefore, conditional instructions translated by the present invention benefit from better utilization of the more efficient target computer architecture.

In decision block 214, it is determined whether additional machine instructions remain in the basic block of instructions to be translated. If any instructions remain to be translated, the remaining instructions are translated in steps similar to those described in blocks 204-212.

Once the basic block of instructions has been translated, an address for the translated basic block of instructions can be entered into a translated address table. As a result, when the basic block of instructions is called to be executed, the translated address table will provide the address of the translated basic block of instructions.

LzLabs GmbH. Ex. 1004-357

In block 216 the translated instructions of the basic block of instructions are re-scheduled by the translator to be executed by a processor in an order based on inter-instruction dependencies, which is commonly referred to as the critical path of execution. In one embodiment, the critical path of execution includes the instructions of a translated compare instruction which generates and saves a difference between two values in a memory location. These instructions are followed by the translated conditional instruction which evaluates the difference in the memory location.

The instructions of the translated compare instruction that update the status flags are scheduled to be executed by a processor after the critical path of execution, as shown in the translated basic block of instructions 162 in Figure 1. As a result, the critical path of executing the translated compare and the translated conditional instruction is shorter and faster than it would be if it included the instructions to update the status flags.

In an alternative embodiment, the instructions of the translated compare instruction may be scheduled to save the difference to a memory location and update the status flags in parallel when the present invention is performed on a computer system having a processor that is able to execute multiple instructions per clock cycle. This type of processor is commonly referred to as a superscalar processor.

The parallel execution is performed by identifying a group of instructions that do not conflict with each other, or require use of the same functional unit. These instructions are then scheduled to be executed in the same clock cycle.

In the foregoing specification, the invention has been described with reference to specific exemplary embodiments thereof. It will, however, be evident that various modifications and changes may be made thereto without departing from the broader spirit and scope of the invention. For instance, the present invention may be performed dynamically during the execution of a computer program or statically when the program is not being executed. Moreover, where it has been shown that the binary translator, CISC program, translated basic blocks, and translated address table are stored in memory, these items and instructions can also be stored on other computer-readable mediums, including magnetic and optical disk, and other random-access memory devices. The specification and drawings are, accordingly, to be regarded in an illustrative rather than restrictive sense.

LzLabs GmbH. Ex. 1004-358

CLAIMS

What is claimed is:

1. A computer-implemented method for translating a conditioned instruction compatible with a first Instruction Set Architecture (ISA) into a conditional instruction compatible with a second ISA, the method comprising the steps of:

a) decoding a conditional instruction compatible with the first ISA, wherein a condition of the conditional instruction is dependent on at least one status flag; and

b) translating the conditional instruction to be compatible with the second ISA, wherein the condition of the conditional instruction is altered to be dependent on a previously computed difference between two values.

2. The method of claim 1, wherein the previously computed difference between two values is stored in a memory location.

3. The method of claim 2, further including the following steps prior to the step of decoding a conditional instruction:

c) decoding an arithmetic instruction compatible with the first instruction set, the arithmetic instruction when executed, generates the difference between two values and updates said status flag, wherein the difference is architecturally transparent to the first ISA; and

d) translating the arithmetic instruction to be compatible with the second ISA, wherein the arithmetic instruction is altered to have the arithmetic instruction when executed generate the difference between two values and place the difference in the memory location.

4. The method of claim 3, wherein the arithmetic instruction is a compare instruction.

5. The method of claim 4, wherein the method is performed during an execution of a computer program that includes the arithmetic instruction and the conditional instruction.

LzLabs GmbH. Ex. 1004-359

6. The method of claim 5, wherein the first ISA is a Complex Instruction Set Computer Architecture, and the second ISA is a Reduced Instruction Set Computer Architecture.

7. The method of claim 6, further including the step of:

e) scheduling the arithmetic instruction to have the arithmetic instruction update the status flag outside a critical path of execution.

8. The method of claim 6, further including the step of:

e) scheduling the arithmetic instruction to have the arithmetic instruction update the status flag and save the difference in the memory location, in parallel.

9. . A computer-implemented method for translating conditional instructions to increase a performance of conditional instructions when executed on a computer architecture exclusive of native support for status flags, the method comprising the steps of:

decoding an arithmetic instruction, the arithmetic instruction, which when executed, generates a difference between a first value and a second value and updates at least one status flag, wherein the difference is transparent to the computer architecture exclusive of native support for status flags;

translating the arithmetic instruction to have the arithmetic instruction when executed generate the difference between the first value and the second value, wherein the difference is placed in a memory location;

decoding a conditional instruction, wherein a condition of the conditional instruction is dependent on the status flag; and

translating the conditional instruction to have the condition of the conditional instruction dependent on the difference placed in the memory location.

10. A computer-implemented method for translating a first set instructions compatible with a first Instruction Set Architecture (ISA) into a second set of instructions compatible with a second ISA, the method comprising the steps of:

LzLabs GmbH. Ex. 1004-360

a) decoding an arithmetic instruction compatible with the first ISA, the arithmetic instruction which, when executed, generates a difference between two values and updates a status flag, wherein the difference is architecturally transparent to the first ISA; and

b) translating the arithmetic instruction to be compatible with the second ISA, wherein the arithmetic instruction is altered to have the arithmetic instruction when executed generate the difference between two values and place the difference in a memory location.

11. A binary translator comprising:

means for decoding a conditional instruction compatible with a first ISA, wherein a condition of the conditional instruction is dependent on at least one status flag; and

means for translating the conditional instruction to be compatible with a second ISA, wherein the condition of the conditional instruction is altered to be dependent on a previously computed difference between two values.

12. The binary translator of claim 11, wherein the previously computed difference is residing in a memory location.

13. The binary translator of claim 12, further comprising:

means for decoding an arithmetic instruction compatible with the first ISA, the arithmetic instruction which when executed, generates the difference between two values and updates said status flag, wherein the difference is architecturally transparent to the first ISA; and

means for translating the arithmetic instruction to be compatible with the second ISA, wherein the arithmetic instruction is altered to have the arithmetic instruction when executed generate the difference between two values and place the difference in a memory location.

14. The binary translator of claim 13, wherein the arithmetic instruction is a compare instruction.

LzLabs GmbH. Ex. 1004-361

15. The binary translator of claim 14, wherein the means for translating the arithmetic instruction and the means for translating the conditional instruction, translate the respective instructions during an execution of a computer program that includes the arithmetic instruction and the conditional instruction.

16. The binary translator of claim 15, wherein the first ISA is a Complex Instruction Set Computer Architecture, and the second ISA is a Reduced Instruction Set Computer Architecture.

LzLabs GmbH. Ex. 1004-362

17. The binary translator of claim 16, further comprising:

a means for scheduling the arithmetic instruction to have the arithmetic instruction update said status flag outside a critical path of execution.

18. The binary translator of claim 16, further comprising:

a means for scheduling the arithmetic instruction to have the arithmetic instruction update the status flag and save the difference at the memory location in parallel.

19. A computer readable-medium having stored thereon a plurality of a sequence of instructions for translating instructions, the sequence of instructions, which when executed by a processor, cause the processor to perform the steps of:

a) decoding a conditional instruction compatible with a first instruction set architecture (ISA), wherein a condition of the conditional instruction is dependent on at least one status flag; and

b) translating the conditional instruction to be compatible with a second ISA, wherein the condition of the conditional instruction is altered to be dependent on a previously computed difference between two values.

20. The computer readable-medium of claim 19, wherein the difference between the two values is residing in a memory location

21. The computer readable-medium of claim 20, wherein the plurality of sequences of instructions further include additional instructions, which when executed by the processor, cause said processor to perform the steps of:

c) decoding an arithmetic instruction compatible with the first instruction set, the arithmetic instruction which when executed, generates the difference between two values and updates said status flag, wherein the difference is architecturally transparent to the first ISA; and

d) translating the arithmetic instruction to be compatible with the second ISA, wherein the arithmetic instruction is altered to have the arithmetic instruction when

LzLabs GmbH. Ex. 1004-363

executed generate the difference between two values and place the difference in the memory location.

22. The computer readable-medium of claim 21, wherein the arithmetic instruction is a compare instruction.

23. The computer readable-medium of claim 22, wherein the step of translating the arithmetic instruction and the step of translating the conditional instruction are performed during an execution of a computer program that includes the arithmetic instruction and the conditional instruction.

24. The computer readable-medium of claim 23, wherein the first ISA is a Complex Instruction Set Computer Architecture, and the second ISA is a Reduced Instruction Set Computer Architecture.

25. The computer readable-medium of claim 24, wherein the plurality of sequences of instructions further include additional instructions, which when executed by the processor, cause said processor to perform the step of:

e) scheduling the arithmetic instruction to have the arithmetic instruction update said status flag outside a critical path of execution.

26. The computer readable-medium of claim 24, wherein the plurality of sequences of instructions further include additional instructions, which when executed by the processor, cause said processor to perform the step of:

e) scheduling the arithmetic instruction to have the arithmetic instruction update the status flag and save the difference in a memory location, in parallel.

LzLabs GmbH. Ex. 1004-364

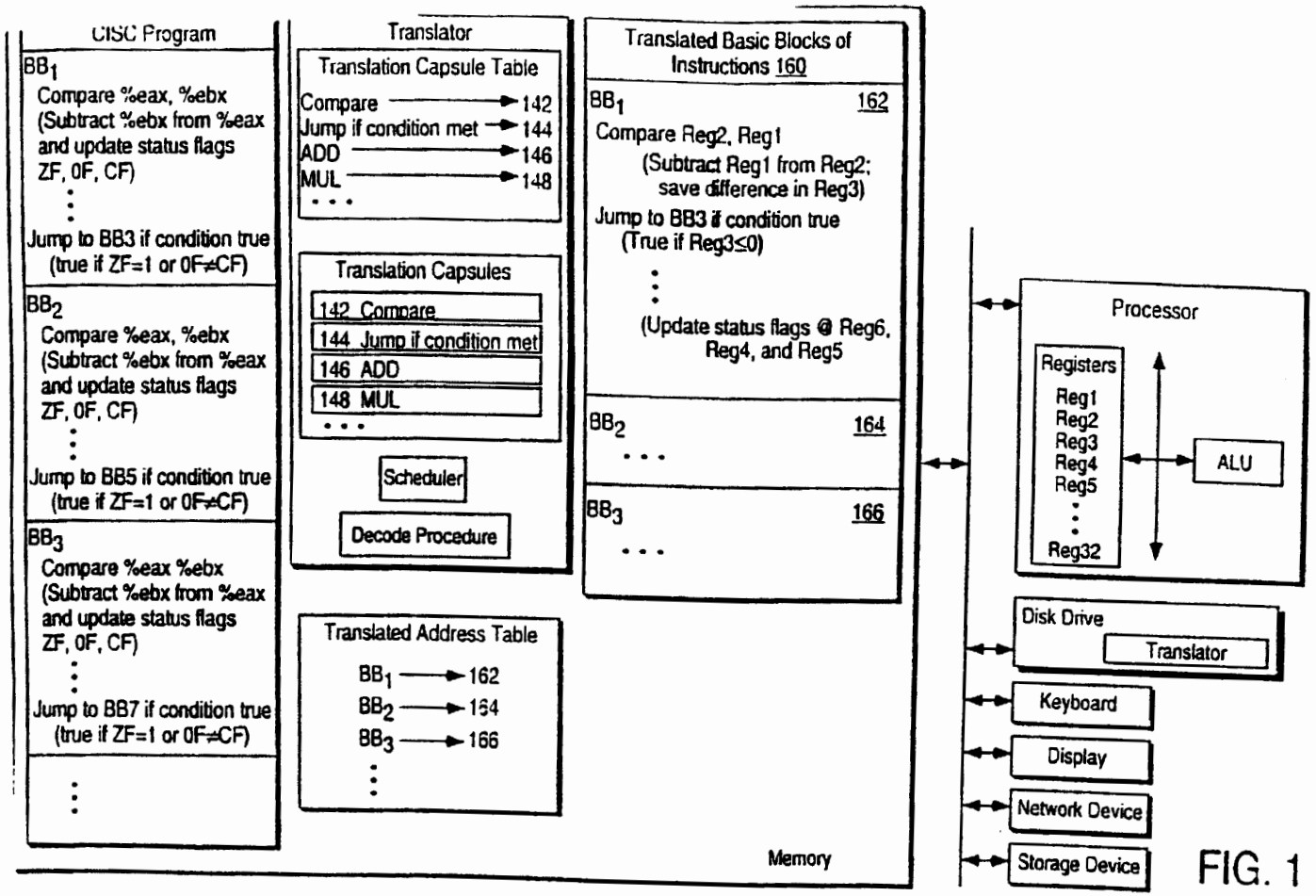
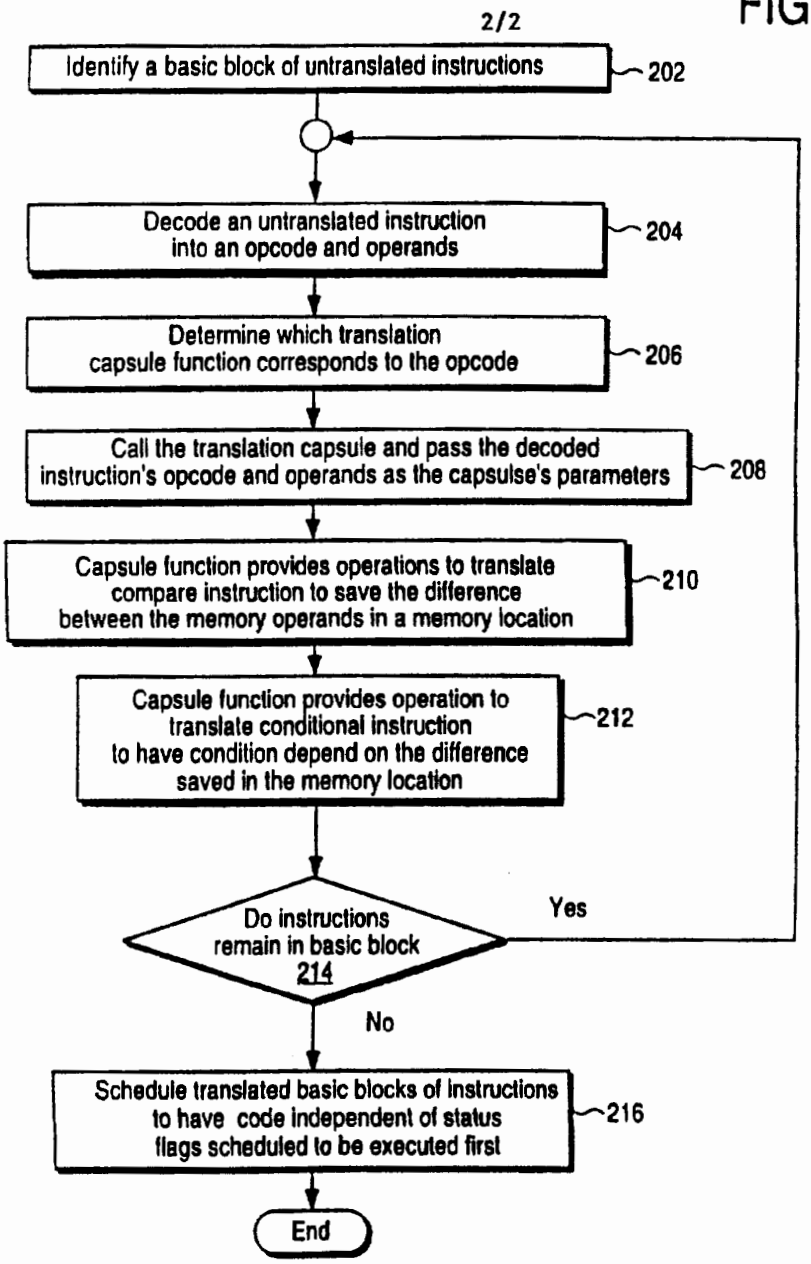


FIG. 1

LzLabs GmbH. Ex. 1004-365

FIG. 2



LzLabs GmbH. Ex. 1004-366

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US97/10341

A. CLASSIFICATION OF SUBJECT MATTER
IPC(6) :G06F 9/455
US CL :395/705
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
U.S. : 395/385, 586, 705, 707, 708, 709

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
APS (Fogg, translate, object, code, program, architecture, instruction, conditional)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 5,313,614 A (GOETTELMANN et al.,) 17 MAY 1994, see abstract, fig. 11, col. 3 line 55 to col. 4 line 50.	1-26
A	US 5,301,325 A (BENSON) 05 APRIL 1994, see abstract.	1-26
Y	US 5,241,678 A (FUTAMURA ET AL.) 31 AUGUST 1993, see abstract, col. 2 line 1 to col. 3 line 16.	1-26
A	US 4,951,195 A (FOGG, JR. ET AL.) 21 AUGUST 1990, see abstract.	1-26
A	MOTOROLA. MC68030 Enhanced 32-bit Microprocessor User's Manual, Second Edition, 1989, pg. 3-71 to 3-72.	1-26

[X] Further documents are listed in the continuation of Box C. [ ] See patent family annex.

*	Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A"	document defining the general state of the art which is not considered to be of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E"	earlier document published on or after the international filing date	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O"	document referring to an oral disclosure, use, exhibition or other means		
"P"	document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search
08 OCTOBER 1997

Date of mailing of the international search report
12 NOV 1997

Name and mailing address of the ISA/US
Commissioner of Patents and Trademarks
Box PCT
Washington, D.C. 20231
Facsimile No. (703) 305-3230

Authorized officer
RICHARD ELLIS
Telephone No. (703) 305-9600

Form PCT/ISA/210 (second sheet)(July 1992)*

LzLabs GmbH. Ex. 1004-367

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US97/10341

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	GAINES, R. Stockton. On the Translation of Machine Language Programs, Communications of the ACM, Volume 8, Number 12, December 1965, pp. 736-741.	1-26
A	BERGH, Arndt B. et al. HP3000 Emulation on HP Precision Architecture Computers, Hewlett-Packard Journal, December 1987, pp. 87-89.	1-26
A	HUNTER, Colin et al. DOS at RISC, Byte Magazine, November 1989, pp. 361-368.	1-26
A	SAARI, Michael. 68000 Binary Code.Translator, FORML Conference Proceedings, 1987, pp. 48-52.	1-26
A	BANNING, John. The XDOS Binary Code Conversion System, IEEE, 1989, pp. 282-287.	1-26

LzLabs GmbH. Ex. 1004-368

PATENT COOPERATION TREATY

Pending TW + US

UKIPLAW

From the INTERNATIONAL SEARCHING AUTHORITY

PCT

To:
IBM UNITED KINGDOM LIMITED
Attn. Williams, Julian David
Intellectual Property Law
Hursley Park
Winchester
Hampshire SO21 2JN
GRANDE BRETAGNE

18 APR 2008
Kevin/CO

NOTIFICATION OF TRANSMITTAL OF THE INTERNATIONAL SEARCH REPORT AND THE WRITTEN OPINION OF THE INTERNATIONAL SEARCHING AUTHORITY, OR THE DECLARATION

(PCT Rule 44.1)

	Date of mailing (day/month/year) 18/04/2008
Applicant's or agent's file reference POU080152	FOR FURTHER ACTION See paragraphs 1 and 4 below
International application No. PCT/EP2008/050725	International filing date (day/month/year) 22/01/2008
Applicant INTERNATIONAL BUSINESS MACHINES CORPORATION	

1. [x] The applicant is hereby notified that the international search report and the written opinion of the International Searching Authority have been established and are transmitted herewith.

   Filing of amendments and statement under Article 19:
   The applicant is entitled, if he so wishes, to amend the claims of the International Application (see Rule 46):

   When? The time limit for filing such amendments is normally two months from the date of transmittal of the International Search Report.

   Where? Directly to the International Bureau of WIPO, 34 chemin des Colombettes 1211 Geneva 20, Switzerland, Facsimile No.: (41-22) 338.82.70

   For more detailed instructions, see the notes on the accompanying sheet.

2. [ ] The applicant is hereby notified that no international search report will be established and that the declaration under Article 17(2)(a) to that effect and the written opinion of the International Searching Authority are transmitted herewith.

3. [ ] **With regard to the protest** against payment of (an) additional fee(s) under Rule 40.2, the applicant is notified that:

   [ ] the protest together with the decision thereon has been transmitted to the International Bureau together with the applicant's request to forward the texts of both the protest and the decision thereon to the designated Offices.

   [ ] no decision has been made yet on the protest; the applicant will be notified as soon as a decision is made.

4. **Reminders**

   Shortly after the expiration of 18 months from the priority date, the international application will be published by the International Bureau. If the applicant wishes to avoid or postpone publication, a notice of withdrawal of the international application, or of the priority claim, must reach the International Bureau as provided in Rules 90bis.1 and 90bis.3, respectively, before the completion of the technical preparations for international publication.

   The applicant may submit comments on an informal basis on the written opinion of the International Searching Authority to the International Bureau. The International Bureau will send a copy of such comments to all designated Offices unless an international preliminary examination report has been or is to be established. These comments would also be made available to the public but not before the expiration of 30 months from the priority date.

   Within 19 months from the priority date, but only in respect of some designated Offices, a demand for international preliminary examination must be filed if the applicant wishes to postpone the entry into the national phase until 30 months from the priority date (in some Offices even later); otherwise, the applicant must, within 20 months from the priority date, perform the prescribed acts for entry into the national phase before those designated Offices.

   In respect of other designated Offices, the time limit of 30 months (or later) will apply even if no demand is filed within 19 months.

   See the Annex to Form PCT/IB/301 and, for details about the applicable time limits, Office by Office, see the *PCT Applicant's Guide*, Volume II, National Chapters and the WIPO Internet site.

Name and mailing address of the International Searching Authority European Patent Office, P.B. 5818 Patentlaan 2 NL-2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Authorized officer Karine Lambert

Form PCT/ISA/220 (October 2005) *(See notes on accompanying sheet)*

LzLabs GmbH. Ex. 1004-369

PATENT COOPERATION TREATY

PCT

INTERNATIONAL SEARCH REPORT

(PCT Article 18 and Rules 43 and 44)

Applicant's or agent's file reference	FOR FURTHER ACTION	see Form PCT/ISA/220 as well as, where applicable, item 5 below.
POU060152		
International application No.	International filing date (day/month/year)	(Earliest) Priority Date (day/month/year)
PCT/EP2008/050725	22/01/2008	30/01/2007

Applicant

INTERNATIONAL BUSINESS MACHINES CORPORATION

This international search report has been prepared by this International Searching Authority and is transmitted to the applicant according to Article 18. A copy is being transmitted to the International Bureau.

This international search report consists of a total of 3 sheets.

[X] It is also accompanied by a copy of each prior art document cited in this report.

1. **Basis of the report**

a. With regard to the **language**, the international search was carried out on the basis of:

[X] the international application in the language in which it was filed

[ ] a translation of the international application into _____, which is the language of a translation furnished for the purposes of international search (Rules 12.3(a) and 23.1(b))

b. [ ] This international search report has been established taking into account the **rectification of an obvious mistake** authorized by or notified to this Authority under Rule 91 (Rule 43.6bis(a)).

c. [ ] With regard to any **nucleotide and/or amino acid sequence** disclosed in the international application, see Box No. I.

2. [ ] **Certain claims were found unsearchable** (See Box No. II)

3. [ ] **Unity of invention is lacking** (see Box No III)

4. With regard to the **title**,

[X] the text is approved as submitted by the applicant

[ ] the text has been established by this Authority to read as follows:

5. With regard to the **abstract**,

[X] the text is approved as submitted by the applicant

[ ] the text has been established, according to Rule 38.2(b), by this Authority as it appears in Box No. IV. The applicant may, within one month from the date of mailing of this international search report, submit comments to this Authority

6. With regard to the **drawings**,

a. the figure of the drawings to be published with the abstract is Figure No. 2

[X] as suggested by the applicant

[ ] as selected by this Authority, because the applicant failed to suggest a figure

[ ] as selected by this Authority, because this figure better characterizes the invention

b. [ ] none of the figures is to be published with the abstract

LzLabs GmbH. Ex. 1004-370

INTERNATIONAL SEARCH REPORT

International application No
PCT/EP2008/050725

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06F9/318 G06F9/455

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)
EPO-Internal, INSPEC, IBM-TDB

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	WO 97/50031 A (INTEL CORP [US]) 31 December 1997 (1997-12-31) page 2, line 13 - page 3, line 18 page 6, line 17 - page 7, line 23	1-7
A	US 5 774 694 A (MURARI SRINIVASAN [US]) 30 June 1998 (1998-06-30) column 1, line 45 - column 2, line 5 column 3, line 22 - line 39	1-7
A	EP 1 455 273 A (NORTHROP GRUMMAN CORP [US]) 8 September 2004 (2004-09-08) paragraph [0057]	1-7
A	US 2006/173931 A1 (BROADHURST JAMES [US]) 3 August 2006 (2006-08-03) paragraphs [0009], [0040] - [0042]	1-7

☐ Further documents are listed in the continuation of Box C. ☒ See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
8 April 2008	18/04/2008

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Thibaudeau, Jean

Form PCT/ISA/210 (second sheet) (April 2005)

LzLabs GmbH. Ex. 1004-371

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/EP2008/050725

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 9750031 A	31-12-1997	AU 4144697 A	14-01-1998
		US 5903760 A	11-05-1999
US 5774694 A	30-06-1998	NONE	
EP 1455273 A	08-09-2004	AU 2004200602 A1	23-09-2004
		CA 2455479 A1	06-09-2004
		CN 1538291 A	20-10-2004
		JP 2004272922 A	30-09-2004
		KR 20040079851 A	16-09-2004
		US 2004177346 A1	09-09-2004
US 2006173931 A1	03-08-2006	WO 2006083808 A2	10-08-2006

LzLabs GmbH. Ex. 1004-372

PATENT COOPERATION TREATY

From the
INTERNATIONAL SEARCHING AUTHORITY

To:

see form PCT/ISA/220

PCT

WRITTEN OPINION OF THE INTERNATIONAL SEARCHING AUTHORITY

(PCT Rule 43*bis*.1)

Date of mailing
(day/month/year) see form PCT/ISA/210 (second sheet)

Applicant's or agent's file reference
see form PCT/ISA/220

FOR FURTHER ACTION
See paragraph 2 below

International application No.	International filing date *(day/month/year)*	Priority date *(day/month/year)*
PCT/EP2008/050725	22.01.2008	30.01.2007

International Patent Classification (IPC) or both national classification and IPC
INV. G06F9/318 G06F9/455

Applicant
INTERNATIONAL BUSINESS MACHINES CORPORATION

1. This opinion contains indications relating to the following items:

- ☒ Box No. I Basis of the opinion
- ☐ Box No. II Priority
- ☐ Box No. III Non-establishment of opinion with regard to novelty, inventive step and industrial applicability
- ☐ Box No. IV Lack of unity of invention
- ☒ Box No. V Reasoned statement under Rule 43*bis*.1(a)(i) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement
- ☐ Box No. VI Certain documents cited
- ☒ Box No. VII Certain defects in the international application
- ☐ Box No. VIII Certain observations on the international application

2. **FURTHER ACTION**

If a demand for international preliminary examination is made, this opinion will usually be considered to be a written opinion of the International Preliminary Examining Authority ("IPEA") except that this does not apply where the applicant chooses an Authority other than this one to be the IPEA and the chosen IPEA has notified the International Bureau under Rule 66.1*bis*(b) that written opinions of this International Searching Authority will not be so considered.

If this opinion is, as provided above, considered to be a written opinion of the IPEA, the applicant is invited to submit to the IPEA a written reply together, where appropriate, with amendments, before the expiration of 3 months from the date of mailing of Form PCT/ISA/220 or before the expiration of 22 months from the priority date, whichever expires later.

For further options, see Form PCT/ISA/220.

3. For further details, see notes to Form PCT/ISA/220.

Name and mailing address of the ISA:	Date of completion of this opinion	Authorized Officer
European Patent Office D-80298 Munich Tel. +49 89 2399 - 0 Tx: 523656 epmu d Fax: +49 89 2399 - 4465	see form PCT/ISA/210	Thibaudeau, Jean Telephone No. +49 89 2399-2349

Form (PCT/ISA/237) (Cover Sheet) (April 2005)

LzLabs GmbH. Ex. 1004-373

WRITTEN OPINION OF THE INTERNATIONAL SEARCHING AUTHORITY

International application No.
PCT/EP2008/050725

Box No. I Basis of the opinion

1. With regard to the **language**, this opinion has been established on the basis of:

   ☒ the international application in the language in which it was filed

   ☐ a translation of the international application into , which is the language of a translation furnished for the purposes of international search (Rules 12.3(a) and 23.1 (b)).

2. ☐ This opinion has been established taking into account the **rectification of an obvious mistake** authorized by or notified to this Authority under Rule 91 (Rule 43bis.1(a))

3. With regard to any **nucleotide and/or amino acid sequence** disclosed in the international application and necessary to the claimed invention, this opinion has been established on the basis of:

   a. type of material:

      ☐ a sequence listing

      ☐ table(s) related to the sequence listing

   b. format of material:

      ☐ on paper

      ☐ in electronic form

   c. time of filing/furnishing:

      ☐ contained in the international application as filed.

      ☐ filed together with the international application in electronic form.

      ☐ furnished subsequently to this Authority for the purposes of search.

4. ☐ In addition, in the case that more than one version or copy of a sequence listing and/or table relating thereto has been filed or furnished, the required statements that the information in the subsequent or additional copies is identical to that in the application as filed or does not go beyond the application as filed, as appropriate, were furnished.

5. Additional comments:

LzLabs GmbH. Ex. 1004-374

Box No. V Reasoned statement under Rule 43*bis*.1(a)(i) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement

1. Statement

Novelty (N)	Yes:	Claims	1-7
	No:	Claims	
Inventive step (IS)	Yes:	Claims	1-7
	No:	Claims	
Industrial applicability (IA)	Yes:	Claims	1-7
	No:	Claims	

2. Citations and explanations

see separate sheet

Box No. VII Certain defects in the international application

The following defects in the form or contents of the international application have been noted:

see separate sheet

LzLabs GmbH. Ex. 1004-375

1. Reference is made to the following document:

   D1: WO 97/50031 A (INTEL CORP [US]) 31 December 1997 (1997-12-31)

2. **Item V: Reasoned statement under Rule 66.2(a)(ii) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement**

   Independent claims 1, 6 and 7 seem to meet the criteria set forth in Article 33(1) PCT with respect to the available prior art for the following reasons:

   D1 discloses a method for emulating computer instructions from a source machine to produce sequences of instructions on a target machine. Said method mimicks an instruction present in a source instruction stream (compare instruction - see page 6, lines 24-26) in a manner which sets one or more target CPU flag bits (see page 6, lines 30-32) and which places a result in a first storage location (register reg3 - see page 6, lines 27-29).
   However D1 further discloses that said result stored in a first storage location (reg3) is directly used by a conditional instruction that depends directly on said stored value (see page 7, lines 6-9, line 12 and lines 15-21) but does not seem to disclose nor render obvious the second to fourth steps of claim 1.

3. **Item VII: Certain defects in the international application**

   i) To meet the requirements of Rule 6.3(b) PCT the independent claims should be properly cast in the two part form, with those features which in combination are part of the prior art (see document D1) being placed in the preamble.

   ii) The features of the claims are not provided with reference signs placed in parentheses (Rule 6.2(b) PCT).

   iii) The description is not in conformity with the claims as required by Rule 5.1(a)(iii)

LzLabs GmbH. Ex. 1004-376

PCT.

iv) Contrary to the requirements of Rule 5.1(a)(ii) PCT, the relevant background art disclosed in the document D1 is not mentioned in the description, nor is this document identified therein.

v) The vague and imprecise statement in the description on page 13 ("spirit of the ....") implies that the subject-matter for which protection is sought may be different to that defined by the claims, thereby resulting in lack of clarity (Article 6 PCT) when used to interpret them.

LzLabs GmbH. Ex. 1004-377

Electronic Patent Application Fee Transmittal

Application Number:	11668623
Filing Date:	30-Jan-2007
Title of Invention:	METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS
First Named Inventor/Applicant Name:	Reid T. Copeland
Filer:	John E. Campbell/Jennifer Keenan
Attorney Docket Number:	POU920060152US1

Filed as Large Entity

Utility under 35 USC 111(a) Filing Fees

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Basic Filing:				
Pages:				
Claims:				
Miscellaneous-Filing:				
Petition:				
Patent-Appeals-and-Interference:				
Post-Allowance-and-Post-Issuance:				
Extension-of-Time:				

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Miscellaneous:				
Submission- Information Disclosure Stmt	1806	1	180	180
		Total in USD ($)		**180**

Electronic Acknowledgement Receipt

EFS ID:	4252647
Application Number:	11668623
International Application Number:	
Confirmation Number:	4180
Title of Invention:	METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS
First Named Inventor/Applicant Name:	Reid T. Copeland
Customer Number:	46369
Filer:	John E. Campbell/Jennifer Keenan
Filer Authorized By:	John E. Campbell
Attorney Docket Number:	POU920060152US1
Receipt Date:	07-NOV-2008
Filing Date:	30-JAN-2007
Time Stamp:	15:22:36
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	yes
Payment Type	Deposit Account
Payment was successfully received in RAM	$180
RAM confirmation Number	614
Deposit Account	090463
Authorized User	
The Director of the USPTO is hereby authorized to charge indicated fees and credit any overpayment as follows: Charge any Additional Fees required under 37 C.F.R. Section 1.16 (National application filing, search, and examination fees) Charge any Additional Fees required under 37 C.F.R. Section 1.17 (Patent application and reexamination processing fees)	

Charge any Additional Fees required under 37 C.F.R. Section 1.19 (Document supply fees)					
File Listing:					
Document Number	**Document Description**	**File Name**	**File Size(Bytes)/ Message Digest**	**Multi Part /.zip**	**Pages (if appl.)**
1	Information Disclosure Statement Letter	POU920060152US1_IDS_Transmittal.pdf	30141 b22043dd5b5489f6d440e85b4d6a59001ae9ff2e	no	1
Warnings:					
Information:					
2	Information Disclosure Statement (IDS) Filed (SB/08)	POU920060152US1_IDS.pdf	30831 d4ca32241145367c49c011d5d9e65180b23987fe	no	3
Warnings:					
Information:					
This is not an USPTO supplied IDS fillable form					
3	Foreign Reference	EP1455273A.pdf	438199 1df53999b7fa1346db30c4f3e7f2799440875d6f	no	26
Warnings:					
Information:					
4	Foreign Reference	WO9750031.pdf	797969 f817d12f7f4dd2dc8cbe6329fec8a801b25a6421	no	20
Warnings:					
Information:					
5	Foreign Reference	POU920060152US1_Inernational_Search_Report.pdf	466557 3ea1808ba4212b8d12548da00128fed7490085f0	no	9
Warnings:					
Information:					
6	Fee Worksheet (PTO-06)	fee-info.pdf	30489 75eb369112b5dec666c70a5186b3eb387536bbc4	no	2
Warnings:					
Information:					
		Total Files Size (in bytes):	1794186		

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office
If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NUMBER	FILING OR 371(C) DATE	FIRST NAMED APPLICANT	ATTY. DOCKET NO./TITLE
11/668,623	01/30/2007	Reid T. Copeland	POU920060152US1

CONFIRMATION NO. 4180

46369
HESLIN ROTHENBERG FARLEY & MESITI P.C.
5 COLUMBIA CIRCLE
ALBANY, NY 12203

PUBLICATION NOTICE



Date Mailed: 07/31/2008

Title:METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS

Publication No.US-2008-0184014-A1
Publication Date:07/31/2008

NOTICE OF PUBLICATION OF APPLICATION

The above-identified application will be electronically published as a patent application publication pursuant to 37 CFR 1.211, et seq. The patent application publication number and publication date are set forth above.

The publication may be accessed through the USPTO's publically available Searchable Databases via the Internet at www.uspto.gov. The direct link to access the publication is currently http://www.uspto.gov/patft/.

The publication process established by the Office does not provide for mailing a copy of the publication to applicant. A copy of the publication may be obtained from the Office upon payment of the appropriate fee set forth in 37 CFR 1.19(a)(1). Orders for copies of patent application publications are handled by the USPTO's Office of Public Records. The Office of Public Records can be reached by telephone at (703) 308-9726 or (800) 972-6382, by facsimile at (703) 305-8759, by mail addressed to the United States Patent and Trademark Office, Office of Public Records, Alexandria, VA 22313-1450 or via the Internet.

In addition, information on the status of the application, including the mailing date of Office actions and the dates of receipt of correspondence filed in the Office, may also be accessed via the Internet through the Patent Electronic Business Center at www.uspto.gov using the public side of the Patent Application Information and Retrieval (PAIR) system. The direct link to access this status information is currently http://pair.uspto.gov/. Prior to publication, such status information is confidential and may only be obtained by applicant using the private side of PAIR.

Further assistance in electronically accessing the publication, or about PAIR, is available by calling the Patent Electronic Business Center at 1-866-217-9197.

Office of Data Managment, Application Assistance Unit (571) 272-4000, or (571) 272-4200, or 1-888-786-0101

LzLabs GmbH. Ex. 1004-383

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NUMBER	FILING or 371(c) DATE	GRP ART UNIT	FIL FEE REC'D	ATTY.DOCKET.NO	DRAWINGS	TOT CLAIMS	IND CLAIMS
11/668,623	01/30/2007	2183	1000	POU920060152US1	4	8	3

CONFIRMATION NO. 4180

46369
HESLIN ROTHENBERG FARLEY & MESITI P.C.
5 COLUMBIA CIRCLE
ALBANY, NY12203

FILING RECEIPT

Date Mailed: 02/22/2007

Receipt is acknowledged of this regular Patent Application. It will be considered in its order and you will be notified as to the results of the examination. Be sure to provide the U.S. APPLICATION NUMBER, FILING DATE, NAME OF APPLICANT, and TITLE OF INVENTION when inquiring about this application. Fees transmitted by check or draft are subject to collection. Please verify the accuracy of the data presented on this receipt. **If an error is noted on this Filing Receipt, please mail to the Commissioner for Patents P.O. Box 1450 Alexandria Va 22313-1450. Please provide a copy of this Filing Receipt with the changes noted thereon. If you received a "Notice to File Missing Parts" for this application, please submit any corrections to this Filing Receipt with your reply to the Notice. When the USPTO processes the reply to the Notice, the USPTO will generate another Filing Receipt incorporating the requested corrections (if appropriate).**

Applicant(s)

Reid T. Copeland, Toronto, CANADA;
Patrick R. Doyle, Toronto, CANADA;
Charles B. Hall, Calgary, CANADA;
Andrew Johnson, Hampshire, UNITED KINGDOM;
Ali I. Sheikh, Toronto, CANADA;

Assignment For Published Patent Application

INTERNATIONAL BUSINESS MACHINES CORPORATION, Armonk, NY

Power of Attorney: The patent practitioners associated with Customer Number 46369

Domestic Priority data as claimed by applicant

Foreign Applications

If Required, Foreign Filing License Granted: 02/22/2007

The country code and number of your priority application, to be used for filing abroad under the Paris Convention, is **US11/668,623**

Projected Publication Date: 07/31/2008

Non-Publication Request: No

Early Publication Request: No

Title

LzLabs GmbH. Ex. 1004-384

METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS

Preliminary Class

712

PROTECTING YOUR INVENTION OUTSIDE THE UNITED STATES

Since the rights granted by a U.S. patent extend only throughout the territory of the United States and have no effect in a foreign country, an inventor who wishes patent protection in another country must apply for a patent in a specific country or in regional patent offices. Applicants may wish to consider the filing of an international application under the Patent Cooperation Treaty (PCT). An international (PCT) application generally has the same effect as a regular national patent application in each PCT-member country. The PCT process **simplifies** the filing of patent applications on the same invention in member countries, but **does not result** in a grant of "an international patent" and does not eliminate the need of applicants to file additional documents and fees in countries where patent protection is desired.

Almost every country has its own patent law, and a person desiring a patent in a particular country must make an application for patent in that country in accordance with its particular laws. Since the laws of many countries differ in various respects from the patent law of the United States, applicants are advised to seek guidance from specific foreign countries to ensure that patent rights are not lost prematurely.

Applicants also are advised that in the case of inventions made in the United States, the Director of the USPTO must issue a license before applicants can apply for a patent in a foreign country. The filing of a U.S. patent application serves as a request for a foreign filing license. The application's filing receipt contains further information and guidance as to the status of applicant's license for foreign filing.

Applicants may wish to consult the USPTO booklet, "General Information Concerning Patents" (specifically, the section entitled "Treaties and Foreign Patents") for more information on timeframes and deadlines for filing foreign patent applications. The guide is available either by contacting the USPTO Contact Center at 800-786-9199, or it can be viewed on the USPTO website at http://www.uspto.gov/web/offices/pac/doc/general/index.html.

For information on preventing theft of your intellectual property (patents, trademarks and copyrights), you may wish to consult the U.S. Government website, http://www.stopfakes.gov. Part of a Department of Commerce initiative, this website includes self-help "toolkits" giving innovators guidance on how to protect intellectual property in specific countries such as China, Korea and Mexico. For questions regarding patent enforcement issues, applicants may call the U.S. Government hotline at 1-866-999-HALT (1-866-999-4158).

LICENSE FOR FOREIGN FILING UNDER

Title 35, United States Code, Section 184

Title 37, Code of Federal Regulations, 5.11 & 5.15

GRANTED

The applicant has been granted a license under 35 U.S.C. 184, if the phrase "IF REQUIRED, FOREIGN FILING LICENSE GRANTED" followed by a date appears on this form. Such licenses are issued in all applications where the conditions for issuance of a license have been met, regardless of whether or not a license may be required as set forth in 37 CFR 5.15. The scope and limitations of this license are set forth in 37 CFR 5.15(a) unless an earlier license has been issued under 37 CFR 5.15(b). The license is subject to revocation upon written notification. The date indicated is the effective date of the license, unless an earlier license of similar scope has been granted under 37 CFR 5.13 or 5.14.

This license is to be retained by the licensee and may be used at any time on or after the effective date thereof unless it is revoked. This license is automatically transferred to any related applications(s) filed under 37 CFR 1.53(d). This license is not retroactive.

The grant of a license does not in any way lessen the responsibility of a licensee for the security of the subject matter as imposed by any Government contract or the provisions of existing laws relating to espionage and the national security or the export of technical data. Licensees should apprise themselves of current regulations especially with respect to certain countries, of other agencies, particularly the Office of Defense Trade Controls, Department of State (with respect to Arms, Munitions and Implements of War (22 CFR 121-128)); the Bureau of Industry and Security, Department of Commerce (15 CFR parts 730-774); the Office of Foreign AssetsControl, Department of Treasury (31 CFR Parts 500+) and the Department of Energy.

NOT GRANTED

No license under 35 U.S.C. 184 has been granted at this time, if the phrase "IF REQUIRED, FOREIGN FILING LICENSE GRANTED" DOES NOT appear on this form. Applicant may still petition for a license under 37 CFR 5.12, if a license is desired before the expiration of 6 months from the filing date of the application. If 6 months has lapsed from the filing date of this application and the licensee has not received any indication of a secrecy order under 35 U.S.C. 181, the licensee may foreign file the application pursuant to 37 CFR 5.15(b).

PTO/SB/14 (08-05)
Approved for use through 07/31/2006. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	POU920060152US1
	Application Number	
Title of Invention	METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS	

The application data sheet is part of the provisional or nonprovisional application for which it is being submitted. The following form contains the bibliographic data arranged in a format specified by the United States Patent and Trademark Office as outlined in 37 CFR 1.76.
This document may be completed electronically and submitted to the Office in electronic format using the Electronic Filing System (EFS) or the document may be printed and included in a paper filed application.

Secrecy Order 37 CFR 5.2

☐ Portions or all of the application associated with this Application Data Sheet may fall under a Secrecy Order pursuant to 37 CFR 5.2 (Paper filers only. Applications that fall under Secrecy Order may not be filed electronically.)

Applicant Information:

Applicant 1 Remove

Applicant Authority ◉ Inventor ○ Legal Representative under 35 U.S.C. 117 ○ Party of Interest under 35 U.S.C. 118

Prefix	Given Name	Middle Name	Family Name	Suffix
	Reid	T.	Copeland	

Residence Information (Select One) ○ US Residency ◉ Non US Residency ○ Active US Military Service

City	Toronto	Country Of Residence[i]	CA

Citizenship under 37 CFR 1.41(b) [i] CA

Mailing Address of Applicant:

Address 1	64 Nipigon Avenue		
Address 2			
City	Toronto	State/Province	ON
Postal Code	M2M 2W1	Country[i]	CA

Applicant 2 Remove

Applicant Authority ◉ Inventor ○ Legal Representative under 35 U.S.C. 117 ○ Party of Interest under 35 U.S.C. 118

Prefix	Given Name	Middle Name	Family Name	Suffix
	Patrick	R.	Doyle	

Residence Information (Select One) ○ US Residency ◉ Non US Residency ○ Active US Military Service

City	Toronto	Country Of Residence[i]	CA

Citizenship under 37 CFR 1.41(b) [i] CA

Mailing Address of Applicant:

Address 1	88-50 Green Belt Drive		
Address 2			
City	Toronto	State/Province	ON
Postal Code	M3C 1M4	Country[i]	CA

Applicant 3 Remove

Applicant Authority ◉ Inventor ○ Legal Representative under 35 U.S.C. 117 ○ Party of Interest under 35 U.S.C. 118

Prefix	Given Name	Middle Name	Family Name	Suffix
	Charles	B.	Hall	

Residence Information (Select One) ○ US Residency ◉ Non US Residency ○ Active US Military Service

City	Calgary	Country Of Residence[i]	CA

LzLabs GmbH. Ex. 1004-387

PTO/SB/14 (08-05)
Approved for use through 07/31/2006. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	POU920060152US1
	Application Number	
Title of Invention	METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS	

Citizenship under 37 CFR 1.41(b) [i]	CA		
Mailing Address of Applicant:			
Address 1	2307 Lancing Avenue SW		
Address 2			
City	Calgary	**State/Province**	AB
Postal Code	T3E 5P5	**Country**[i]	CA

Applicant 4 Remove

Applicant Authority (•) Inventor	( ) Legal Representative under 35 U.S.C. 117	( ) Party of Interest under 35 U.S.C. 118	
Prefix	**Given Name**	**Middle Name**	**Family Name** / **Suffix**
	Andrew		Johnson

Residence Information (Select One) ( ) US Residency (•) Non US Residency ( ) Active US Military Service

City	Hampshire	**Country Of Residence**[i]	GB
Citizenship under 37 CFR 1.41(b) [i]	GB		
Mailing Address of Applicant:			
Address 1	Brunswick House		
Address 2	High Street, Twyford, Winchester		
City	Hampshire	**State/Province**	
Postal Code	SO21 1RF	**Country**[i]	GB

Applicant 5 Remove

Applicant Authority (•) Inventor	( ) Legal Representative under 35 U.S.C. 117	( ) Party of Interest under 35 U.S.C. 118	
Prefix	**Given Name**	**Middle Name**	**Family Name** / **Suffix**
	Ali	I.	Sheikh

Residence Information (Select One) ( ) US Residency (•) Non US Residency ( ) Active US Military Service

City	Toronto	**Country Of Residence**[i]	CA
Citizenship under 37 CFR 1.41(b) [i]	PK		
Mailing Address of Applicant:			
Address 1	131-11 Sherwood Avenue		
Address 2			
City	Toronto	**State/Province**	ON
Postal Code	M4P 2A6	**Country**[i]	CA

All Inventors Must Be Listed - Additional Inventor Information blocks may be generated within this form by selecting the **Add** button. Add

Correspondence Information:

Enter either Customer Number or complete the Correspondence Information section below.
For further information see 37 CFR 1.33(a).

☐ **An Address is being provided for the correspondence Information of this application.**

Customer Number	46369

LzLabs GmbH. Ex. 1004-388

PTO/SB/14 (08-05)
Approved for use through 07/31/2006. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	POU920060152US1
	Application Number	
Title of Invention	METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS	

Email Address	ldc@hrfmlaw.com	Add Email	Remove Email

Application Information:

Title of the Invention	METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS		
Attorney Docket Number	POU920060152US1	**Small Entity Status Claimed**	☐
Application Type	Nonprovisional		
Subject Matter	Utility		
Suggested Class (if any)		**Sub Class (if any)**	
Suggested Technology Center (if any)			
Total Number of Drawing Sheets (if any)	4	**Suggested Figure for Publication (if any)**	
Publication Information:			
☐ Request Early Publication (Fee required at time of Request 37 CFR 1.219)			
☐ Request Not to Publish. I hereby request that the attached application not be published under 35 U.S.C. 122(b) and certify that the invention disclosed in the attached application has not been and will not be the subject of an application filed in another country, or under a multilateral agreement, that requires publication at eighteen months after filing.			

Representative Information:

Representative information should be provided for all practitioners having a power of attorney in the application. Providing this information in the Application Data Sheet does not constitute a power of attorney in the application (see 37 CFR 1.32).
Enter either Customer Number or complete the Representative Name section below. If both sections are completed the Customer Number will be used for the Representative Information during processing.

Please Select One:	◉ Customer Number	○ US Patent Practitioner	○ US Representative (37 CFR 11.9)
Customer Number	46369		

Domestic Priority Information:

This section allows for the applicant to claim benefit under 35 U.S.C. 119(e), 120, 121, or 365(c). Providing this information in the application data sheet constitutes the specific reference required by 35 U.S.C. 119(e) or 120, and 37 CFR 1.78(a)(2) or CFR 1.78(a)(4), and need not otherwise be made part of the specification.

Prior Application Status			Remove
Application Number	Continuity Type	Prior Application Number	Filing Date (YYYY-MM-DD)

Additional Domestic Priority Data may be generated within this form by selecting the **Add** button. Add

Foreign Priority Information:

LzLabs GmbH. Ex. 1004-389

PTO/SB/14 (08-05)
Approved for use through 07/31/2006. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Application Data Sheet 37 CFR 1.76	Attorney Docket Number	POU920060152US1
	Application Number	
Title of Invention	METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS	

This section allows for the applicant to claim benefit of foreign priority and to identify any prior foreign application for which priority is not claimed. Providing this information in the application data sheet constitutes the claim for priority as required by 35 U.S.C. 119(b) and 37 CFR 1.55(a).

Remove

Application Number	Country [i]	Parent Filing Date (YYYY-MM-DD)	Priority Claimed
			◉ Yes ○ No

Additional Foreign Priority Data may be generated within this form by selecting the **Add** button. Add

Assignee Information:

Providing this information in the application data sheet does not substitute for compliance with any requirement of part 3 of Title 37 of the CFR to have an assignment recorded in the Office.

Remove

Assignee 1

If the Assignee is an Organization check here. ☒

Organization Name	INTERNATIONAL BUSINESS MACHINES CORPORATION		
Mailing Address Information:			
Address 1	New Orchard Road		
Address 2			
City	Armonk	**State/Province**	NY
Country [i]	US	Postal Code	10504
Phone Number		Fax Number	
Email Address			

Additional Assignee Data may be generated within this form by selecting the **Add** button. Add

Signature:

A signature of the applicant or representative is required in accordance with 37 CFR 1.33 and 10.18. Please see 37 CFR 1.4(d) for the form of the signature.

Signature	/Lawrence D. Cutter/			Date (YYYY-MM-DD)	2007-01-26
First Name	Lawrence D.	Last Name	Cutter	Registration Number	28501

This collection of information is required by 37 CFR 1.76. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 23 minutes to complete, including gathering, preparing, and submitting the completed application data sheet form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

LzLabs GmbH. Ex. 1004-390

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C. 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether the Freedom of Information Act requires disclosure of these records.

2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.

3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.

4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).

5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.

6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).

7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.

8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspections or an issued patent.

9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

LzLabs GmbH. Ex. 1004-391

METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS

Technical Field

[0001] This invention relates in general to the emulation of computer system architectures, and more particularly, to methods and systems for handling condition code settings encountered in the emulation process. Even more particularly, the present invention is directed to providing sequences of instructions that produce valid condition code settings without the use of branching instructions from the target architecture.

Background of the Invention

[0002] In virtually all modern data processing systems, the execution of various operations such as arithmetic operations, logical operations and even data transfer operations, may result in the generation of several bits of data to indicate the outcome status of instruction execution. These bits are typically referred to as condition codes. As a simple example, a special condition code setting may be set after an arithmetic addition which results in an overflow due to the addends being too large for the number of bits available for the result. The use of condition codes permeates the execution of almost every instruction

[0003] A classic example of an instruction which produces condition code changes upon execution is the *compare* instruction which sets a condition code to "zero" if the operands are equal, to "one" if the first operand is strictly less than the second operand and to "two" if the first operand is strictly greater than the second operand. The *compare* instruction represents an archetypical use of condition code settings.

[0004] For a number of reasons, it may be desirable to emulate the instructions designed for one computer architecture on another system with a different set of executable instructions. For example, emulation may be employed in system design or test. It may also be employed to expand the capabilities of one data processing system so

LzLabs GmbH. Ex. 1004-392

that it is enabled to handle instructions written for another system. The present invention relates to the handling of condition code settings in the context of instruction emulation. While the systems and methods of the present invention are widely applicable to any emulation method where condition codes are present, it is particularly applicable to the emulation of the z/Architecture. However, the principles set forth herein are applicable to any source architecture and to any target architecture.

[0005] In the principle emulation environment considered in the present description, it is the job of emulation software to accept, as input, strings of source architecture instructions and to generate therefrom strings of instructions that, when run on the target architecture, produce the same results. These results include the setting of various condition codes, such as sign, carry, overflow and various others indicating exceptions and machine states. It is noted that while an emulation environment preferably results in the setting of hardware or condition code elements in the target architecture, the present invention also contemplates the situation in which condition codes are generated and stored in locations other than condition code registers in the target machine.

[0006] It is to be particularly noted that the present invention, deliberately avoids the conventional handling of condition code generation. An example of this difference is provided through a brief consideration of the compare instruction. This instruction compares two operands and sets a two bit condition code according to the outcome of the comparison. For example, if the comparison of the two operands determines that they are the same, the condition code is set to zero. If it is determined that the first operand is strictly less than the second operand the condition code is set to one. Lastly, if it is determined that the first operand is greater than the second operand, the condition code is set to two. In conventional approaches to the emulation of a compare instruction, the result is the construction of a sequence of instructions, which include three branch instructions. For the reasons set forth immediately below the presence of branch instructions in the target architecture instruction stream is undesirable.

LzLabs GmbH. Ex. 1004-393

[0007] Branch instructions are undesirable for at least two reasons. In particular, it is noted that most modern data processing architectures include features known as branch prediction. In these architectures, a guess is made as to which of two or more paths that the instruction stream will follow after encountering a branch instruction. If a correct guess is made, then all is well and machine processing time is thereby speeded up. However, if an incorrect to guess is made the machine hardware must backtrack through the path taken, and then take another path. At this point in time, the branch instruction is seen to be a detriment to overall processing speed. Accordingly, it is seen that branch instructions introduce complications which are not otherwise present. Furthermore, as a second reason for their avoidance, is noted that branch instructions actually consume the aforementioned branch prediction resources so that they are thus not available for other instruction streams being executed by a processor. Thus branch instructions are not only potentially wasteful in and of themselves, they also deprive other instruction streams of limited, yet valuable, computer resources.

[0008] Accordingly, it is seen that the designer of emulation systems is faced with the paradoxical choice of needing branch instructions to successfully emulate the generation of condition code settings in target architectures while at the same time desiring to avoid branching instructions because of their disadvantages. This problem is especially severe when condition code generation and functionality in the target architecture are quite different from that found in the architecture of the source machine.

[0009] It is to be particularly noted that computer programs that emulate the machine state of the z/Architecture deal with many z/Architecture instructions that modify the condition codes. In short, the z/Architecture is a prime exemplar of an architecture in which condition code settings are typically quite different than that found in other architectures, especially ones that have historically grown up from relatively simple microprocessor designs. Additionally, the modification of condition code settings in the z/Architecture is pervasive. The generation and use of condition code settings is most typically found as the result of performing an arithmetic, logical or compare operations

LzLabs GmbH. Ex. 1004-394

after which one or more condition code settings are changed based on the result or other factors. The pervasiveness of condition code modifying instructions in the z/Architecture and the sometimes arbitrary semantics of these instructions introduces complicated control flow to the stream of instructions that are ultimately executed on the target architecture. This control flow adds considerable space and performance overhead to the emulated instructions. The present invention is directed to more efficiently handling this situation. While the method and system herein are particularly applicable to the so-called z/Architecture which is present in large data processing systems manufactured and sold by the assignee of the present invention, it is by no means limited to that architecture as a base of supply for source instructions.

[0010] It should also be noted that the present invention is employed in two contexts or modes. In one mode, source computer instructions are converted into target machine instructions for later execution. In another mode of operation, more akin to the operation of interpreters, source instructions are converted into target instructions for immediate execution. The present invention, in its broadest scope, contemplates both of these modalities of operation.

Summary of the Invention

[0011] In a method for emulating computer instructions from a source machine to produce sequences of instructions on a target machine, the present invention generates a sequence of target machine instructions which together operate to directly calculate target machine condition codes from carry, sign and overflow codes without the use of branch instructions from the target machine. The direct calculation avoids the use of branching instructions whose disadvantages are cited above.

[0012] The present invention provides specific guiding techniques and several sequences derived from these techniques to efficiently set conditions codes or detect exceptional cases in an emulated binary translation environment for the z/Architecture. These techniques are specifically directed to situations in which the PowerPC

LzLabs GmbH. Ex. 1004-395

architecture and the Intel IA32 architectures are employed to emulate the z/Architecture. The sequences of the present inventions are more efficient and generally smaller as opposed to a more straightforward method that requires more flow control. However, it is noted that the principals, techniques and methods of the present invention are not limited to any particular target machine architecture. The two exemplar architectures discussed herein are merely the most currently ones anticipated to be of the greatest value.

[0013] Additional features and advantages are realized through the techniques of the present invention. Other embodiments and aspects of the invention are described in detail herein and are considered a part of the claimed invention.

[0014] The recitation herein of a list of desirable objects which are met by various embodiments of the present invention is not meant to imply or suggest that any or all of these objects are present as essential features, either individually or collectively, in the most general embodiment of the present invention or in any of its more specific embodiments.

Brief Description of the Drawings

[0015] The subject matter which is regarded as the invention is particularly pointed out and distinctly claimed in the concluding portion of the specification. The invention, however, both as to organization and method of practice, together with the further objects and advantages thereof, may best be understood by reference to the following description taken in connection with the accompanying drawings in which:

[0016] Figure 1 is a flow chart indicating the fact that conventional handling of condition code settings in an emulation environment employs up to three branch instructions;

[0017] Figure 2 is a flow chart of the present process in which branch instructions are avoided in the emulation of computer instructions;

LzLabs GmbH. Ex. 1004-396

[0018] Figure 3 is a block diagram illustrating an example of the environment in which the present invention is employed;

[0019] Figure 4 is a top view of a typical computer readable medium containing program code which implements the methods of the present invention, as for example, as shown a Compact Disc (CD); and

[0020] Figure 5 is a block diagram illustrating the environment in which the present invention operates and is employed.

Detailed Description

[0021] The technique used herein to derive the sequences is to implement very short one or two instruction sequence fragments that set a bit or bits in a result or temporary register to distinguish each possible outcome of a condition code setting. These small code fragments typically manipulate the carry, sign and overflow bits and are tied together into slightly larger sequences by standard and usually high-performing shifts, rotates and various arithmetic or Boolean instructions available on most computer architectures. Very efficient sequences result by avoiding both branch instructions and more complex instructions that are less likely to be optimized in hardware.

[0022] In some cases it is possible and efficient to manipulate the PowerPC condition code register itself to derive the z/Architecture condition code settings. In these cases a PowerPC record form instruction is used and the resulting PowerPC condition register is manipulated by rotations and logical operations to derive the corresponding z/Architecture condition code setting.

[0023] The examples below are in assembler pseudocode and are applicable to most architectures that can manipulate a carry bit and have the usual shift/rotate/negate instructions. The assembler pseudocode used is purposefully verbose so as to make the implied semantics clear. In some cases non-standard mnemonics are used when an efficient way to implement a particular operation is more likely to vary on different

LzLabs GmbH. Ex. 1004-397

architectures. These non-standard mnemonics are explained more fully in the table below. In each case rX and rY are named target machine registers and "b" is an immediate value. The carry_bit is the carry out of the high order bit position.

Instruction		Description
get_bit	rX, rY, b	The least significant bit in rX receives the bit value (0 or 1) from position b in rY and the rest of rX is set to zero
get_sign_bit	rX, rY	places the sign bit from rY in rX and clears the rest of rX
double	rX, rY	doubles the value in rY and places the result in rX
set_bit_on_zero	rX, rY	if rY contains 0 then a 1 is placed in rX otherwise a 0 is placed in rX
set_bit_on_not_zero	rX, rY	if rY contains a non-zero value then a 1 is placed in rX otherwise a 0 is placed in rX
add_to_carry	rX, rY, rZ	rX = rY + rZ + carry_bit. This operation is also assumed to set the carry bit based on the result of the add.
add_set_carry	rX, rY, rZ	rX = rY + rZ and the carry bit is set based on the result of the add. Similar semantics for sub_set_carry (subtract).
add_to_carry_immed	rX, rY, imm	rX = rY + imm + carry_bit. This operation is also assumed to set the carry bit based on the result of the add.
add_set_carry_immed	rX, rY, imm	rX = rY + imm and set the carry bit based on the result of the add.
move_from_carry	rX	rX = carry bit
flip_bit	rX, rY, b	The bit value at position b of rY is changed from either 0 to 1 or 1 to 0 and the entire changed register value is placed in rX

Table I

[0024] Bits are numbered from 0 - 63 for a 64 bit register and 0 - 31 for a 32 bit register. 0 is the most significant position and 31 or 63 is the least significant position. In the description below, the following register naming conventions are used:

rA, rB - the first and second operand, respectively, of the z/Architecture instruction evaluated into a register;

rT - the result value of the z/Architecture instruction computed into a register;

LzLabs GmbH. Ex. 1004-398

rX, rY, rZ - temporary registers used to hold intermediate results;

rC - the register that will hold the condition code value at the end of the sequence.

[0025] Various sequences derived via this technique are listed and discussed below. As indicated above, the compare instruction is an archetypical example of an instruction which sets condition codes. Accordingly, a sequence for emulating condition code settings is provided below. The sequence provided is typical of the approach taken in the practice of the present invention. In particular, the subject source instruction is the z/Architecture Instruction called the Compare Logical operation. The example assumes that rA and rB are zero-extended in a 64 bit environment (only when compare instruction operates on 32 bit operands).

[1]	sub_set_carry	rC, rA, rB
[2]	set_bit_on_zero	rX, rC
[3]	set_bit_on_not_zero	rC, rC
[4]	add_to_carry_immed	rC, rC, 0
[5]	sub	rC, rC, rX

	rA > rB			rA =rB			rA < rB		
Instruction	Register Contents			Register Contents			Register Contents		
	rC	rX	Carry	rC	rX	Carry	rC	rX	Carry
[1]	rC > 0	N/A	1	rC = 0	N/A	1	rC < 0	N/A	0
[2]	rC > 0	0	1	rC = 0	1	1	rC < 0	0	0
[3]	1	0	1	0	1	1	1	0	0
[4]	2	0	N/A	1	1	N/A	1	0	N/A
[5]	2	0	N/A	0	1	N/A	1	0	N/A

Table II

[0026] As the table above indicates, after the execution of "sub_set_carry" (instruction [1]), the condition of register rX is not applicable. Execution of instruction

LzLabs GmbH. Ex. 1004-399

[1] in the target machine, however, does set a carry bit in the CPU state which is accessed by later instructions. This is indicated in the "Carry" column in the table which refers to the carry bit flag in the target machine. It is important to note and to realize that this carry bit, like many other flag bits in the target machine, is not set in the target machine in the same manner or under the same conditions as are present in the source machine. At this point, conventional approaches to setting a corresponding value in the register location rC would employ multiple branch instructions as shown in Figure 1. These conventional approaches, as well as the present process, operate so as to provide a proper indication of the carry bit for use by the emulation software.

[0027] With respect to instruction [1], its execution sets the carry bit (that is, the CPU carry bit) to "1" in the case that rA ≥ rB and to "0" in the case that rA < rB. Additionally, rC contains the result of the subtraction, which, notably could be "0." The entries "rC > 0," "rC = 0" and "rC < 0" in the table above are meant to provide an indication of the resulting condition. After the execution of instruction [2] (set_bit_on_zero), the status of register rC is unchanged but the contents of rX are set equal to "1" if the two operands, rA and rB, are the same based on the contents or rC (limited to zero or not in this case). The execution of instruction [2] does not affect the contents of rC. Additionally, instruction [2] does not affect the CPU carry bit. With respect to instruction [3] (set_bit_on_not_zero), rC is set equal to "1" whenever rC is not zero, that is, whenever rA is not equal to rB. The CPU carry bit is unaffected by instruction [3]. Thus, at this point, if rA > rB or if rA < rB, then rC = 1, but if rA = rB, then rC = 0. Note that at this point, rX is set up to provide discrimination information distinguishing equality from inequality and that this occurs outside of (that is, apart from) both rC and the CPU carry bit.

[0028] Instruction [4] (add_to_carry_immed) is then executed with the arguments shown (rC, rC, 0), with "0" being an immediate operand. With the operands shown, it carries out the operation: rC + "CPU carry bit" + 0. While it also sets the CPU carry bit as well, this result is not required for subsequent processing. It is seen in Table II above

LzLabs GmbH. Ex. 1004-400

that if rA > rB then the contents of rC are now "2"; if rA = rB, then the contents of rC are "1"; and if rA < rB, then the contents of rC are also "1." At this stage it is relevant to note that there is provided an indication in rC for which the case rA > rB is distinguished from the other two cases (rA = rB and rA < rB).

[0029] The execution of instruction [5] (sub), with operands "rC, rC, rX" provides the last step in which the contents of rX, now denoting equality, are subtracted from rC as mechanism for distinguishing the case that rA = rB from the case that rA < rB since the case of equality results in the subtraction of "1" from "1" and placing the result "0" in rC. Thus, at the end of the instruction sequence set out above, the following results are obtained: rC = 2 if rA > rB; rC = 1 if rA < rB; and rC = 0 if rA = rB.

[0030] In this manner, then, it is seen that desired emulator results for condition code settings are obtained without the execution of any branching instructions. The concepts presented above are equally applicable to the emulation of any source instruction which produces a condition code change. While the above example is specifically directed to the setting of a carry bit, it is equally applicable to other target architecture condition code bits, such as the sign and overflow bits.

[0031] As another example of the application of the present invention to providing condition code generation in an emulation environment the Add Logical (32 bit) and Add Logical (64 bit) instructions are considered below. As with the Compare Logical example discussed above, rA and rB are assumed to be zero extended for a 64 bit target architecture environment for Add Logical (32 bit). The following is a sequence in pseudo-assembly code which provides the proper setting in the location rC at the end of the process. Below, c is the carry bit.

[1] add_set_carry rT, rA, rB

[2] move_from_carry rC

[3] double rC

LzLabs GmbH. Ex. 1004-401

[4] set_bit_on_not_zero rX, rT

[5] or rC, rC, rX

	rT is zero no carry cc = 0				rT not zero no carry cc = 1				rT is zero carry cc = 2				rT not zero carry cc = 3			
	Register Contents				Register Contents				Register Contents				Register Contents			
	rC	rT	rX	c	rC	rT	rX	c	rC	rT	rX	c	rC	rT	rX	c
[1]	n/a	0	n/a	0	n/a	not 0	n/a	0	n/a	0	n/a	1	n/a	not 0	n/a	1
[2]	0	0	n/a	0	0	not 0	n/a	0	1	0	n/a	1	1	not 0	n/a	1
[3]	0	0	n/a	0	0	not 0	n/a	0	2	0	n/a	1	2	not 0	n/a	1
[4]	0	0	0	0	0	not 0	1	0	2	0	0	1	2	not 0	1	1
[5]	**0**	0	0	0	**1**	not 0	1	0	**2**	0	0	1	**3**	not 0	1	1

[0032] In general, this process has the following steps, none of which includes the use or execution of any branch instructions. First, an instruction (step 100 in Figure 2) in the target machine's architecture is executed which mimics the instruction present in the source instruction stream in a manner which sets one or more target CPU flag bits and which places a result in a storage location (first location, such as rC above) accessible to the emulator. Next, an instruction (step 105) is executed which uses that result to set a bit or bits in another emulator controlled storage location (second location, such as rX above) to distinguish one or more case results. Next, the aforementioned result is used to reset itself (step 110) to a shorter bit configuration (one bit in the above example) which also serves to distinguish one or more case results. Next, an instruction is executed (step 115) which employs the first storage location to produce a result which distinguishes a different set of case results. Lastly, a target machine instruction is executed (step 120)

LzLabs GmbH. Ex. 1004-402

which uses the results in the first and second locations to provide an indication in one of the two emulator accessible instructions in which at least three cases are distinguished.

[0033] Even more generally, the present process is directed to emulation methods which do not employ target machine branch instructions but rather employ target machines instructions whose executions result in the control of target machine condition codes which are used in subsequently executed non-branch instructions in ways that are used to distinguish one or more result states which are made available in a location which an emulator can employ as a condition code emulation location.

[0034] It is noted that the process set forth herein contemplates that it encompasses both the generation of suitable sequences of instructions to be executed on a target machine and the actual execution of those instructions on a target machine, whether that execution occurs immediately upon the sequence for a source instruction being generated, as one might find in an "interpretive" environment or in a "compilation-like" environment, where actual execution might occur at a later time, if necessarily at all.

[0035] In any event the environment in which the present invention operates is shown in Figure 3. The present invention operates in a data processing environment which effectively includes one or more of the computer elements shown in Figure 3. In particular, computer 500 includes central processing unit (CPU) 520 which accesses programs and data stored within random access memory 510. Memory 510 is typically volatile in nature and accordingly such systems are provided with nonvolatile memory typically in the form of rotatable magnetic memory 540. While memory 540 is preferably a nonvolatile magnetic device, other media may be employed. CPU 530 communicates with users at consoles such as terminal 550 through Input/Output unit 530. Terminal 550 is typically one of many, if not thousands, of consoles in communication with computer 500 through one or more I/O unit 530. In particular, console unit 550 is shown as having included therein a device for reading medium of one or more types such as CD-ROM 600 shown in Figure 5. Media 600, an example of which is shown in Figure 4, comprises any convenient device including, but not limited to, magnetic media, optical

LzLabs GmbH. Ex. 1004-403

storage devices and chips such as flash memory devices or so-called thumb drives. Disk 600 also represents a more generic distribution medium in the form of electrical signals used to transmit data bits which represent codes for the instructions discussed herein. While such transmitted signals may be ephemeral in nature they still, nonetheless constitute a physical medium carrying the coded instruction bits and are intended for permanent capture at the signal's destination or destinations.

[0036] The typical emulation environment in which the present invention is employed is illustrated in Figure 5. Emulators such as 320 except as input instruction streams 305, representing machine or assembly language instructions which are designed to operate on source machine 300. Emulator 320 employees memory 315 in target machine 310 to produce a stream of instructions which are capable of executing on target machine 310. While Figure 5 particularly shows operation within an emulation environment, it is also noted that the present invention contemplates a situation in which emulator 320 operates essentially as an interpreter in which the instructions are not only translated to the new architecture but in which they are also executed at essentially the same time.

[0037] While the invention has been described in detail herein in accordance with certain preferred embodiments thereof, many modifications and changes therein may be effected by those skilled in the art. Accordingly, it is intended by the appended claims to cover all such modifications and changes as fall within the true spirit and scope of the invention.

LzLabs GmbH. Ex. 1004-404

Claims

What is claimed is:

1. A method for emulating computer instructions from a source machine to produce sequences of instructions on a target machine, said method comprising:

generating a sequence of target machine instructions which together operate to directly calculate target machine condition codes from carry, sign and overflow codes without including branch instructions from the target machine.

2. The method of claim 1 in which said calculation of target machine condition codes is carried out in a sequence of non-branching instructions that manipulate condition code settings produced within the target machine itself to produce the same condition code in a target machine location.

3. The method of claim 2 in which said manipulation employs temporary locations for storing intermediate results.

4. The method of claim 1 further including the step of executing said generated sequence of target machine instructions.

5. The method of claim 4 in which said executing occurs at substantially the same time as said generating.

LzLabs GmbH. Ex. 1004-405

6. The method of claim 1 in which said generating comprises the steps of:

mimicking an instruction present in a source instruction stream in a manner which sets one or more target CPU flag bits and which places a result in a first storage location;

executing an instruction which uses said result to set at least one bit in second storage location so as to distinguish one or more case results;

using said result to reset itself to a shorter bit configuration which serves to distinguish at least one of said case results;

executing an instruction which employs said first storage location to produce a result which distinguishes a different set of case results; and

executing an instruction which uses the results in said first and second storage locations to provide an indication in which at least three cases are distinguished, said cases representing condition codes of said source machine.

LzLabs GmbH. Ex. 1004-406

7. A computer program product comprising a machine readable medium including instructions encoded therein for generating a sequence of target machine instructions which together operate to directly calculate target machine condition codes from carry, sign and overflow codes without including branch instructions from the target machine.

LzLabs GmbH. Ex. 1004-407

8. A data processing system including a memory for stored program execution by said system, said memory having code therein for generating a sequence of target machine instructions which together operate to directly calculate target machine condition codes from carry, sign and overflow codes without including branch instructions from the target machine.

* * * * *

LzLabs GmbH. Ex. 1004-408

METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS

Abstract of the Disclosure

Emulation of source machine instructions is provided in which target machine CPU condition codes are employed to produce emulated condition code settings without the use, encoding or generation of branching instructions.

LzLabs GmbH. Ex. 1004-409

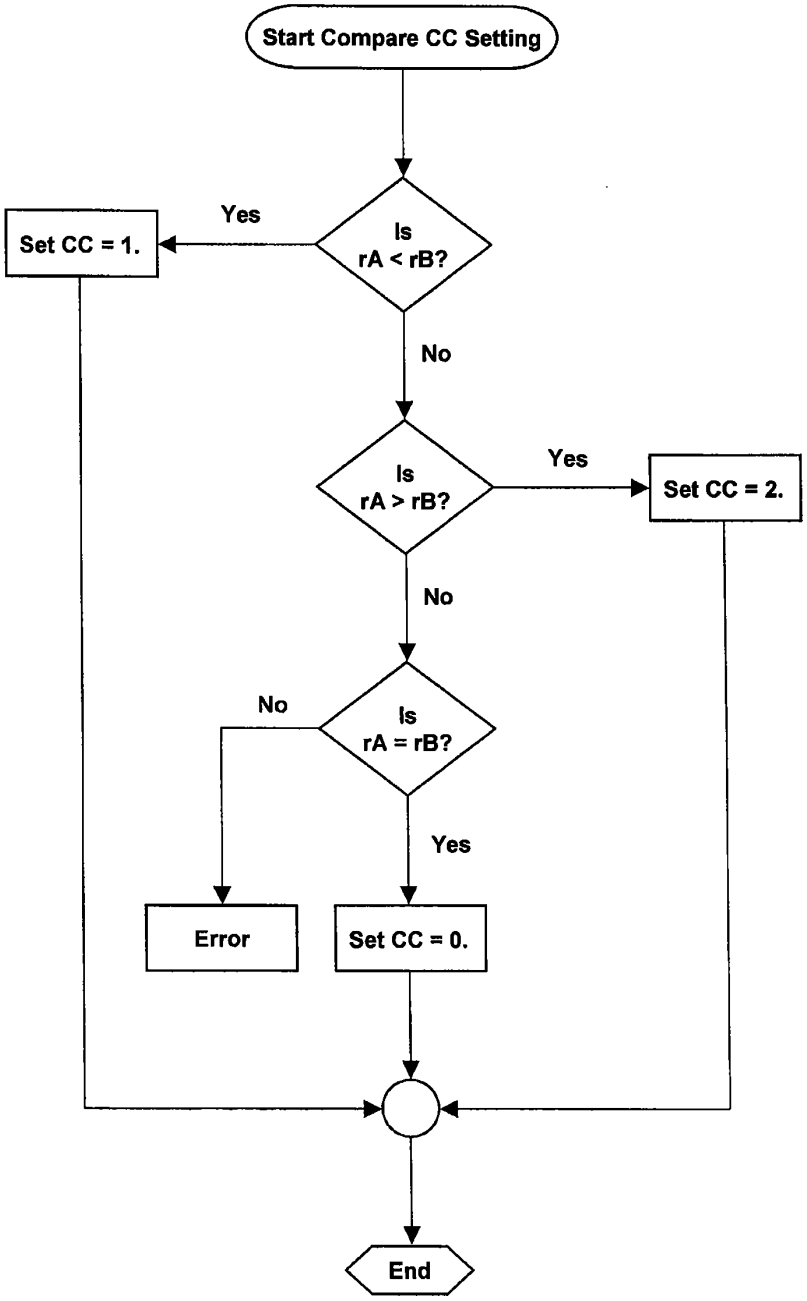

Start Compare CC Setting
Is rA < rB?
Yes
Set CC = 1.
No
Is rA > rB?
Yes
Set CC = 2.
No
Is rA = rB?
No
Yes
Error
Set CC = 0.
End

Figure 1

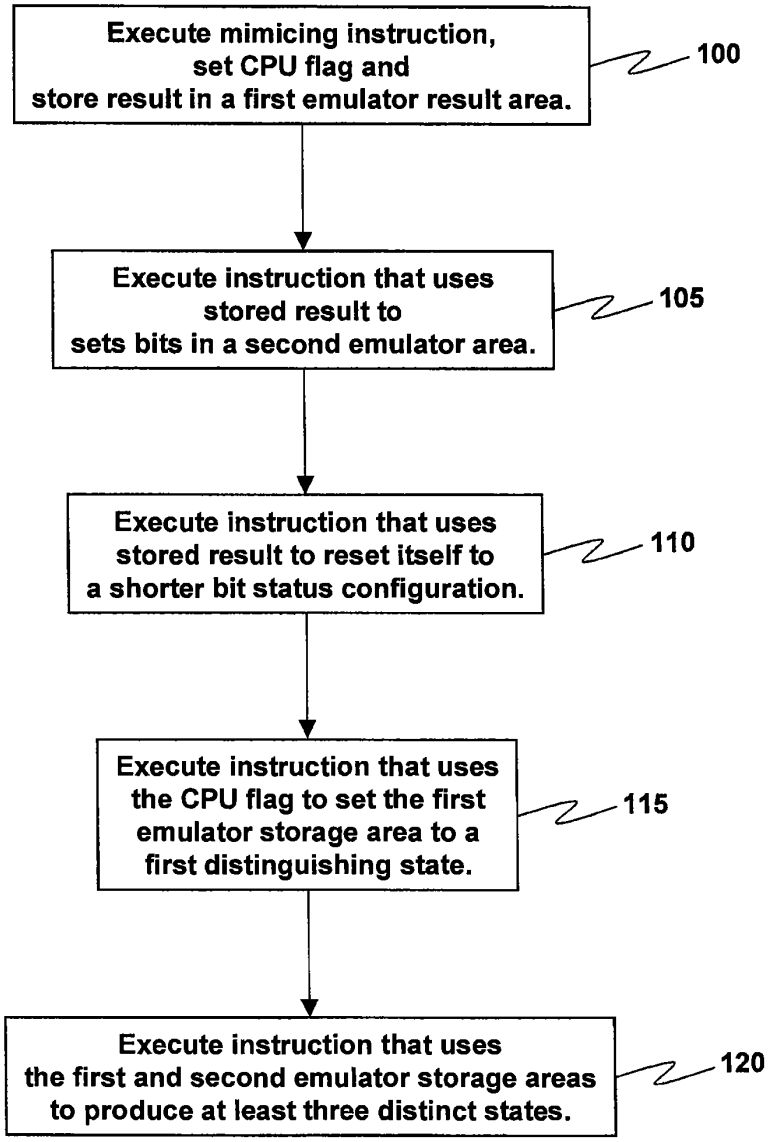

Execute mimicing instruction, set CPU flag and store result in a first emulator result area.
100
Execute instruction that uses stored result to sets bits in a second emulator area.
105
Execute instruction that uses stored result to reset itself to a shorter bit status configuration.
110
Execute instruction that uses the CPU flag to set the first emulator storage area to a first distinguishing state.
115
Execute instruction that uses the first and second emulator storage areas to produce at least three distinct states.
120

Figure 2

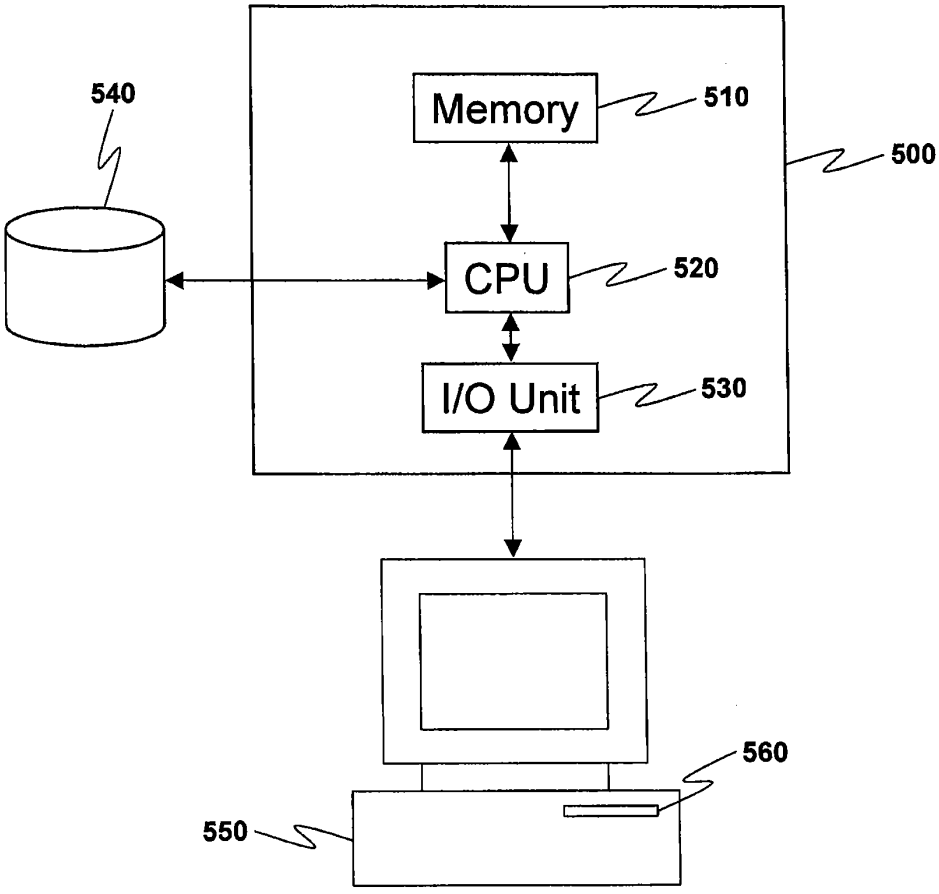


Figure 3

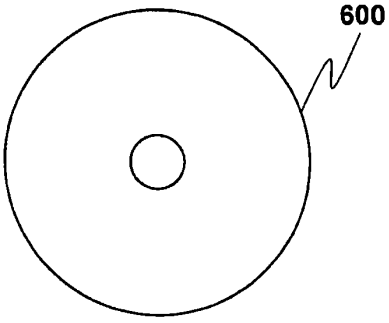


Figure 4

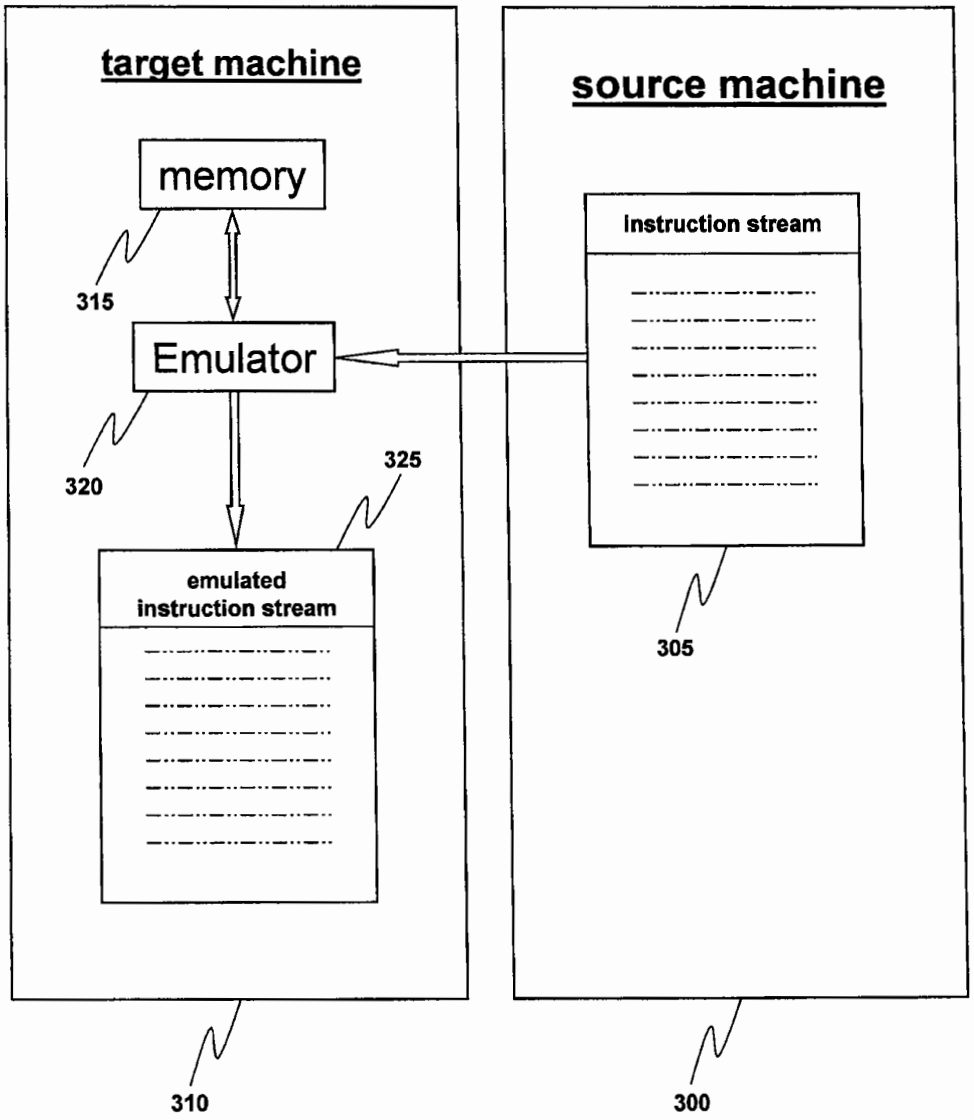

target machine
source machine
memory
315
Emulator
320
325
emulated
instruction stream
Instruction stream
305
310
300

Figure 5

INFORMATION DISCLOSURE CITATION (*USE SEVERAL SHEETS IF NECESSARY*)	ATTY DOCKET NO. **POU920060152US1**	SERIAL NO. **UNASSIGNED**
	APPLICANT(S) **COPELAND ET AL.**	
	FILING DATE **HEREWITH**	GROUP **UNASSIGNED**

U.S. PATENT DOCUMENTS

*EXAMINER INITIAL		DOCUMENT NUMBER	DATE	NAME	CLASS	SUBCLASS	FILING DATE IF APPROPRIATE
	AA	2003/0093649 A1	05/15/2003	Hilton	712	41	

FOREIGN PATENT DOCUMENTS

		DOCUMENT NUMBER	DATE	COUNTRY	CLASS	SUBCLASS	TRANSLATION		
							NO	YES	ABSTRACT

OTHER DOCUMENTS (*Including Author, Title, Date, Pertinent Pages, Etc.*)

EXAMINER	DATE CONSIDERED

EXAMINER: Initial here if reference considered, whether or not citation is in conformance with MPEP 609; Draw line through citation if not in conformance and not considered. **Include copy of this form with next communication to applicant.**

LzLabs GmbH. Ex. 1004-414

Express Mail Label No.

Docket No.
POU920060152US1

Declaration and Power of Attorney For Patent Application

English Language Declaration

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name,

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled

METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS

the specification of which

(check one)

☒ is attached hereto.

☐ was filed on _____ as United States Application No. or PCT International Application Number _____

and was amended on _____

(if applicable)

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information which is material to patentability as defined in 37 CFR 1.56, including for continuation-in-part applications, material information which became available between the filing date of the prior application and the national or PCT international filing date of the continuation-in-part application.

I hereby claim foreign priority benefits under 35 U.S.C. 119(a)-(d) or (f), or 365(b) of any foreign application(s) for patent, or plant breeder's rights certificate(s), or 365(a) of any PCT International application which designated at least one country other than the United States of America, listed below and have also identified below, by checking the box, any foreign application for patent, inventor's or plant breeder's rights certificate(s), or any PCT international application having a filing date before that of the application on which priority is claimed.

Prior Foreign Application(s)			Priority Claimed
_____ (Number)	_____ (Country)	_____ (Day/Month/Year Filed)	☐
_____ (Number)	_____ (Country)	_____ (Day/Month/Year Filed)	☐
_____ (Number)	_____ (Country)	_____ (Day/Month/Year Filed)	☐

LzLabs GmbH. Ex. 1004-415

I hereby claim the benefit under 35 U.S.C. Section 119(e) of any United States provisional application(s) listed below:

(Application Serial No.)	(Filing Date)
(Application Serial No.)	(Filing Date)
(Application Serial No.)	(Filing Date)

I hereby claim the benefit under 35 U. S. C. Section 120 of any United States application(s), or Section 365(c) of any PCT International application designating the United States, listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States or PCT International application in the manner provided by the first paragraph of 35 U.S.C. Section 112, I acknowledge the duty to disclose to the United States Patent and Trademark Office all information known to me to be material to patentability as defined in Title 37, C. F. R., Section 1.56 which became available between the filing date of the prior application and the national or PCT International filing date of this application:

(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)
(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)
(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)
(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)
(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)
(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

LzLabs GmbH. Ex. 1004-416

POWER OF ATTORNEY: As a named inventor, I hereby appoint the following attorney(s) and/or agent(s) to prosecute this application and transact all business in the Patent and Trademark Office connected therewith. *(list name and registration number)*

All practitioners associated with Customer No. 46369

Send Correspondence to: **Lawrence D. Cutter, Esq. at Customer No. 46369**

Direct Telephone Calls to: *(name and telephone number)*

Lawrence D. Cutter, Esq. (518) 452-5600

Full name of sole or first inventor **Reid T. Copeland**
Sole or first inventor's signature Date January 26/2007
Residence **64 Nipigon Avenue, Toronto, Ontario, M2M 2W1, Canada**
Citizenship **Canada**
Post Office Address **64 Nipigon Avenue, Toronto, Ontario, M2M 2W1, Canada**

Full name of second inventor, if any **Patrick R. Doyle**
Second inventor's signature Date
Residence **88-50 Green Belt Drive, Toronto, Ontario, M3C 1M4, Canada**
Citizenship **Canada**
Post Office Address **88-50 Green Belt Drive, Toronto, Ontario, M3C 1M4, Canada**

LzLabs GmbH. Ex. 1004-417

Full name of third inventor, if any
Charles B. Hall

Third inventor's signature — Date

Residence
2307 Lancing Avenue SW, Calgary, Alberta, T3E 5P5, Canada

Citizenship
Canada

Post Office Address
2307 Lancing Avenue SW, Calgary, Alberta, T3E 5P5, Canada

Full name of fourth inventor, if any
Andrew Johnson

Fourth inventor's signature — Date

Residence
Brunswick House, High Street, Twyford, Winchester, Hampshire, SO21 1RF, Great Britain

Citizenship
United Kingdom

Post Office Address
Brunswick House, High Street, Twyford, Winchester, Hampshire, SO21,1RF, Great Britain

Full name of fifth inventor, if any
Ali I. Sheikh

Fifth inventor's signature — Date
Jan 26, 2007

Residence
131-11 Sherwood Avenue, Toronto, Ontario, M4P 2A6, Canada

Citizenship
Pakistan

Post Office Address
131-11 Sherwood Avenue, Toronto, Ontario, M4P 2A6, Canada

Full name of sixth inventor, if any

Sixth inventor's signature — Date

Residence

Citizenship

Post Office Address

LzLabs GmbH. Ex. 1004-418

Express Mail Label No.

Docket No.
POU920060152US1

Declaration and Power of Attorney For Patent Application

English Language Declaration

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name,

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled

METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS

the specification of which

(check one)

☒ is attached hereto.

☐ was filed on _____ as United States Application No. or PCT International Application Number _____

and was amended on _____
(if applicable)

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information which is material to patentability as defined in 37 CFR 1.56, including for continuation-in-part applications, material information which became available between the filing date of the prior application and the national or PCT international filing date of the continuation-in-part application.

I hereby claim foreign priority benefits under 35 U.S.C. 119(a)-(d) or (f), or 365(b) of any foreign application(s) for patent, or plant breeder's rights certificate(s), or 365(a) of any PCT International application which designated at least one country other than the United States of America, listed below and have also identified below, by checking the box, any foreign application for patent, inventor's or plant breeder's rights certificate(s), or any PCT international application having a filing date before that of the application on which priority is claimed.

Prior Foreign Application(s)			Priority Claimed
(Number)	(Country)	(Day/Month/Year Filed)	☐
(Number)	(Country)	(Day/Month/Year Filed)	☐
(Number)	(Country)	(Day/Month/Year Filed)	☐

LzLabs GmbH. Ex. 1004-419

I hereby claim the benefit under 35 U.S.C. Section 119(e) of any United States provisional application(s) listed below:

(Application Serial No.)	(Filing Date)
(Application Serial No.)	(Filing Date)
(Application Serial No.)	(Filing Date)

I hereby claim the benefit under 35 U. S. C. Section 120 of any United States application(s), or Section 365(c) of any PCT International application designating the United States, listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States or PCT International application in the manner provided by the first paragraph of 35 U.S.C. Section 112, I acknowledge the duty to disclose to the United States Patent and Trademark Office all information known to me to be material to patentability as defined in Title 37, C. F. R., Section 1.56 which became available between the filing date of the prior application and the national or PCT International filing date of this application:

(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)
(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)
(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)
(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)
(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)
(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

LzLabs GmbH. Ex. 1004-420

POWER OF ATTORNEY: As a named inventor, I hereby appoint the following attorney(s) and/or agent(s) to prosecute this application and transact all business in the Patent and Trademark Office connected therewith. *(list name and registration number)*

All practitioners associated with Customer No. 46369

Send Correspondence to: **Lawrence D. Cutter, Esq. at Customer No. 46369**

Direct Telephone Calls to: *(name and telephone number)*

Lawrence D. Cutter, Esq. (518) 452-5600

Full name of sole or first inventor **Reid T. Copeland**
Sole or first inventor's signature Date
Residence **64 Nipigon Avenue, Toronto, Ontario, M2M 2W1, Canada**
Citizenship **Canada**
Post Office Address **64 Nipigon Avenue, Toronto, Ontario, M2M 2W1, Canada**

Full name of second inventor, if any **Patrick R. Doyle**
Second inventor's signature Date Jan 29 2007
Residence **88-50 Green Belt Drive, Toronto, Ontario, M3C 1M4, Canada**
Citizenship **Canada**
Post Office Address **88-50 Green Belt Drive, Toronto, Ontario, M3C 1M4, Canada**

LzLabs GmbH. Ex. 1004-421

Full name of third inventor, if any
Charles B. Hall

Third inventor's signature Date

Residence
2307 Lancing Avenue SW, Calgary, Alberta, T3E 5P5, Canada

Citizenship
Canada

Post Office Address
2307 Lancing Avenue SW, Calgary, Alberta, T3E 5P5, Canada

Full name of fourth inventor, if any
Andrew Johnson

Fourth inventor's signature Date

Residence
Brunswick House, High Street, Twyford, Winchester, Hampshire, SO21 1RF, Great Britain

Citizenship
United Kingdom

Post Office Address
Brunswick House, High Street, Twyford, Winchester, Hampshire, SO21,1RF, Great Britain

Full name of fifth inventor, if any
Ali I. Sheikh

Fifth inventor's signature Date

Residence
131-11 Sherwood Avenue, Toronto, Ontario, M4P 2A6, Canada

Citizenship
Pakistan

Post Office Address
131-11 Sherwood Avenue, Toronto, Ontario, M4P 2A6, Canada

Full name of sixth inventor, if any

Sixth inventor's signature Date

Residence

Citizenship

Post Office Address

LzLabs GmbH. Ex. 1004-422

Express Mail Label No.

Docket No.
POU920060152US1

Declaration and Power of Attorney For Patent Application

English Language Declaration

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name,

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled

METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS

the specification of which

(check one)

☒ is attached hereto.

☐ was filed on _____ as United States Application No. or PCT International Application Number _____

and was amended on _____

(if applicable)

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information which is material to patentability as defined in 37 CFR 1.56, including for continuation-in-part applications, material information which became available between the filing date of the prior application and the national or PCT international filing date of the continuation-in-part application.

I hereby claim foreign priority benefits under 35 U.S.C. 119(a)-(d) or (f), or 365(b) of any foreign application(s) for patent, or plant breeder's rights certificate(s), or 365(a) of any PCT International application which designated at least one country other than the United States of America, listed below and have also identified below, by checking the box, any foreign application for patent, inventor's or plant breeder's rights certificate(s), or any PCT international application having a filing date before that of the application on which priority is claimed.

Prior Foreign Application(s)			Priority Claimed
_____ (Number)	_____ (Country)	_____ (Day/Month/Year Filed)	☐
_____ (Number)	_____ (Country)	_____ (Day/Month/Year Filed)	☐
_____ (Number)	_____ (Country)	_____ (Day/Month/Year Filed)	☐

LzLabs GmbH. Ex. 1004-423

I hereby claim the benefit under 35 U.S.C. Section 119(e) of any United States provisional application(s) listed below:

(Application Serial No.)	(Filing Date)
(Application Serial No.)	(Filing Date)
(Application Serial No.)	(Filing Date)

I hereby claim the benefit under 35 U. S. C. Section 120 of any United States application(s), or Section 365(c) of any PCT International application designating the United States, listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States or PCT International application in the manner provided by the first paragraph of 35 U.S.C. Section 112, I acknowledge the duty to disclose to the United States Patent and Trademark Office all information known to me to be material to patentability as defined in Title 37, C. F. R., Section 1.56 which became available between the filing date of the prior application and the national or PCT International filing date of this application:

(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)
(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)
(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)
(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)
(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)
(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

LzLabs GmbH. Ex. 1004-424

POWER OF ATTORNEY: As a named inventor, I hereby appoint the following attorney(s) and/or agent(s) to prosecute this application and transact all business in the Patent and Trademark Office connected therewith. *(list name and registration number)*

All practitioners associated with Customer No. 46369

Send Correspondence to: **Lawrence D. Cutter, Esq. at Customer No. 46369**

Direct Telephone Calls to: *(name and telephone number)*
Lawrence D. Cutter, Esq. (518) 452-5600

Full name of sole or first inventor **Reid T. Copeland**
Sole or first inventor's signature Date
Residence **64 Nipigon Avenue, Toronto, Ontario, M2M 2W1, Canada**
Citizenship **Canada**
Post Office Address **64 Nipigon Avenue, Toronto, Ontario, M2M 2W1, Canada**

Full name of second inventor, if any **Patrick R. Doyle**
Second inventor's signature Date
Residence **88-50 Green Belt Drive, Toronto, Ontario, M3C 1M4, Canada**
Citizenship **Canada**
Post Office Address **88-50 Green Belt Drive, Toronto, Ontario, M3C 1M4, Canada**

LzLabs GmbH. Ex. 1004-425

Full name of third inventor, if any
Charles B. Hall

Third inventor's signature — Date
2007/01/26

Residence
2307 Lancing Avenue SW, Calgary, Alberta, T3E 5P5, Canada

Citizenship
Canada

Post Office Address
2307 Lancing Avenue SW, Calgary, Alberta, T3E 5P5, Canada

Full name of fourth inventor, if any
Andrew Johnson

Fourth inventor's signature — Date

Residence
Brunswick House, High Street, Twyford, Winchester, Hampshire, SO21 1RF, Great Britain

Citizenship
United Kingdom

Post Office Address
Brunswick House, High Street, Twyford, Winchester, Hampshire, SO21,1RF, Great Britain

Full name of fifth inventor, if any
Ali I. Sheikh

Fifth inventor's signature — Date

Residence
131-11 Sherwood Avenue, Toronto, Ontario, M4P 2A6, Canada

Citizenship
Pakistan

Post Office Address
131-11 Sherwood Avenue, Toronto, Ontario, M4P 2A6, Canada

Full name of sixth inventor, if any

Sixth inventor's signature — Date

Residence

Citizenship

Post Office Address

LzLabs GmbH. Ex. 1004-426

Express Mail Label No.

Docket No.
POU920060152US1

Declaration and Power of Attorney For Patent Application

English Language Declaration

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name,

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled

METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS

the specification of which

(check one)

☒ is attached hereto.

☐ was filed on _____ as United States Application No. or PCT International Application Number _____

and was amended on _____
(if applicable)

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information which is material to patentability as defined in 37 CFR 1.56, including for continuation-in-part applications, material information which became available between the filing date of the prior application and the national or PCT international filing date of the continuation-in-part application.

I hereby claim foreign priority benefits under 35 U.S.C. 119(a)-(d) or (f), or 365(b) of any foreign application(s) for patent, or plant breeder's rights certificate(s), or 365(a) of any PCT International application which designated at least one country other than the United States of America, listed below and have also identified below, by checking the box, any foreign application for patent, inventor's or plant breeder's rights certificate(s), or any PCT international application having a filing date before that of the application on which priority is claimed.

Prior Foreign Application(s)			Priority Claimed
(Number)	(Country)	(Day/Month/Year Filed)	☐
(Number)	(Country)	(Day/Month/Year Filed)	☐
(Number)	(Country)	(Day/Month/Year Filed)	☐

LzLabs GmbH. Ex. 1004-427

I hereby claim the benefit under 35 U.S.C. Section 119(e) of any United States provisional application(s) listed below:

(Application Serial No.)	(Filing Date)
(Application Serial No.)	(Filing Date)
(Application Serial No.)	(Filing Date)

I hereby claim the benefit under 35 U. S. C. Section 120 of any United States application(s), or Section 365(c) of any PCT International application designating the United States, listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States or PCT International application in the manner provided by the first paragraph of 35 U.S.C. Section 112, I acknowledge the duty to disclose to the United States Patent and Trademark Office all information known to me to be material to patentability as defined in Title 37, C. F. R., Section 1.56 which became available between the filing date of the prior application and the national or PCT International filing date of this application:

(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)
(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)
(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)
(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)
(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)
(Application Serial No.)	(Filing Date)	(Status) (patented, pending, abandoned)

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

LzLabs GmbH. Ex. 1004-428

POWER OF ATTORNEY: As a named inventor, I hereby appoint the following attorney(s) and/or agent(s) to prosecute this application and transact all business in the Patent and Trademark Office connected therewith. *(list name and registration number)*

All practitioners associated with Customer No. 46369

Send Correspondence to: **Lawrence D. Cutter, Esq. at Customer No. 46369**

Direct Telephone Calls to: *(name and telephone number)*

Lawrence D. Cutter, Esq. (518) 452-5600

Full name of sole or first inventor
Reid T. Copeland

Sole or first inventor's signature Date

Residence
64 Nipigon Avenue, Toronto, Ontario, M2M 2W1, Canada

Citizenship
Canada

Post Office Address
64 Nipigon Avenue, Toronto, Ontario, M2M 2W1, Canada

Full name of second inventor, if any
Patrick R. Doyle

Second inventor's signature Date

Residence
88-50 Green Belt Drive, Toronto, Ontario, M3C 1M4, Canada

Citizenship
Canada

Post Office Address
88-50 Green Belt Drive, Toronto, Ontario, M3C 1M4, Canada

LzLabs GmbH. Ex. 1004-429

Full name of third inventor, if any
Charles B. Hall

Third inventor's signature Date

Residence
2307 Lancing Avenue SW, Calgary, Alberta, T3E 5P5, Canada

Citizenship
Canada

Post Office Address
2307 Lancing Avenue SW, Calgary, Alberta, T3E 5P5, Canada

Full name of fourth inventor, if any
Andrew Johnson

Fourth inventor's signature Andrew Johnson Date 24 January 2007

Residence
Brunswick House, High Street, Twyford, Winchester, Hampshire, SO21 1RF, Great Britain

Citizenship
United Kingdom

Post Office Address
Brunswick House, High Street, Twyford, Winchester, Hampshire, SO21 1RF, Great Britain

Full name of fifth inventor, if any
Ali I. Sheikh

Fifth inventor's signature Date

Residence
131-11 Sherwood Avenue, Toronto, Ontario, M4P 2A6, Canada

Citizenship
Pakistan

Post Office Address
131-11 Sherwood Avenue, Toronto, Ontario, M4P 2A6, Canada

Full name of sixth inventor, if any

Sixth inventor's signature Date

Residence

Citizenship

Post Office Address

LzLabs GmbH. Ex. 1004-430

Electronic Patent Application Fee Transmittal

Application Number:	
Filing Date:	
Title of Invention:	METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS
First Named Inventor/Applicant Name:	Reid T. Copeland
Filer:	Lawrence D. Cutter
Attorney Docket Number:	POU920060152US1

Filed as Large Entity

Utility Filing Fees

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Basic Filing:				
Utility application filing	1011	1	300	300
Utility Search Fee	1111	1	500	500
Utility Examination Fee	1311	1	200	200
Pages:				
Claims:				
Miscellaneous-Filing:				
Petition:				
Patent-Appeals-and-Interference:				

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Post-Allowance-and-Post-Issuance:				
Extension-of-Time:				
Miscellaneous:				
		Total in USD ($)		**1000**

Electronic Acknowledgement Receipt	
EFS ID:	1475898
Application Number:	11668623
International Application Number:	
Confirmation Number:	4180
Title of Invention:	METHOD FOR EFFICIENTLY EMULATING COMPUTER ARCHITECTURE CONDITION CODE SETTINGS
First Named Inventor/Applicant Name:	Reid T. Copeland
Customer Number:	46369
Filer:	Lawrence D. Cutter
Filer Authorized By:	
Attorney Docket Number:	POU920060152US1
Receipt Date:	30-JAN-2007
Filing Date:	
Time Stamp:	13:33:51
Application Type:	Utility

Payment information:

Submitted with Payment	yes
Payment was successfully received in RAM	$1000
RAM confirmation Number	1876
Deposit Account	090463
The Director of the USPTO is hereby authorized to charge indicated fees and credit any overpayment as follows: Charge any Additional Fees required under 37 C.F.R. Section 1.16 and 1.17	

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)	Multi Part /.zip	Pages (if appl.)
1	Application Data Sheet	POU920060152US1_DataSheet.pdf	1175473	no	5
Warnings:					
Information:					
2		POU920060152US1_Application.pdf	669697	yes	18
	Multipart Description/PDF files in .zip description				
	Document Description		**Start**	**End**	
	Specification		1	13	
	Claims		14	17	
	Abstract		18	18	
Warnings:					
Information:					
3	Drawings	POU920060152US1_Drawings.pdf	45354	no	4
Warnings:					
Information:					
4	Information Disclosure Statement (IDS) Filed	POU920060152US1_IDC01.pdf	23676	no	1
Warnings:					
Information:					
This is not an USPTO supplied IDS fillable form					
5	Oath or Declaration filed	POU920060152US1_Declaration.pdf	624974	no	16
Warnings:					
Information:					
6	Fee Worksheet (PTO-06)	fee-info.pdf	8404	no	2
Warnings:					
Information:					
		Total Files Size (in bytes):	2547578		

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office
If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

PTO/SB/06 (12-04)
Approved for use through 7/31/2006. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

PATENT APPLICATION FEE DETERMINATION RECORD
Substitute for Form PTO-875

11/668,623

APPLICATION AS FILED – PART I

FOR	(Column 1) NUMBER FILED	(Column 2) NUMBER EXTRA	SMALL ENTITY RATE ($)	SMALL ENTITY FEE ($)	OR	OTHER THAN SMALL ENTITY RATE ($)	OTHER THAN SMALL ENTITY FEE ($)
BASIC FEE (37 CFR 1.16(a), (b), or (c))							300
SEARCH FEE (37 CFR 1.16(k), (i), or (m))							500
EXAMINATION FEE (37 CFR 1.16(o), (p), or (q))							200
TOTAL CLAIMS (37 CFR 1.16(i))	8 minus 20 =		X$25=			X$50	
INDEPENDENT CLAIMS (37 CFR 1.16(h))	3 minus 3 =		X$100=			X$200=	
APPLICATION SIZE FEE (37 CFR 1.16(s))	If the specification and drawings exceed 100 sheets of paper, the application size fee due is $250 ($125 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR						
MULTIPLE DEPENDENT CLAIM PRESENT (37 CFR 1.16(j))			N/A			N/A	
* If the difference in column 1 is less than zero, enter "0" in column 2.			TOTAL			TOTAL	1000

APPLICATION AS AMENDED – PART II

AMENDMENT A

	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA	SMALL ENTITY RATE ($)	SMALL ENTITY ADDI-TIONAL FEE ($)	OR	OTHER THAN SMALL ENTITY RATE ($)	OTHER THAN SMALL ENTITY ADDI-TIONAL FEE ($)
Total (37 CFR 1.16(i))	*	Minus	**	=	x =		OR	x =	
Independent (37 CFR 1.16(h))	*	Minus	***	=	x =		OR	x =	
Application Size Fee (37 CFR 1.16(s))									
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))					N/A		OR	N/A	
					TOTAL ADD'T FEE		OR	TOTAL ADD'T FEE	

AMENDMENT B

	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA	SMALL ENTITY RATE ($)	SMALL ENTITY ADDI-TIONAL FEE ($)	OR	OTHER THAN SMALL ENTITY RATE ($)	OTHER THAN SMALL ENTITY ADDI-TIONAL FEE ($)
Total (37 CFR 1.16(i))	*	Minus	**	=	x =		OR	x =	
Independent (37 CFR 1.16(h))	*	Minus	***	=	x =		OR	x =	
Application Size Fee (37 CFR 1.16(s))									
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))					N/A		OR	N/A	
					TOTAL ADD'T FEE		OR	TOTAL ADD'T FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.
** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20".
*** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3".
The "Highest Number Previously Paid For" (Total or Independent) is the highest number found in the appropriate box in column 1.

This collection of information is required by 37 CFR 1.16. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

LzLabs GmbH. Ex. 1004-436