US006070214A

# United States Patent [19]

**Ahern**

[11] **Patent Number: 6,070,214**

[45] **Date of Patent: *May 30, 2000**

[54] **SERIALLY LINKED BUS BRIDGE FOR EXPANDING ACCESS OVER A FIRST BUS TO A SECOND BUS**

[75] Inventor: **Frank Ahern**, Scottsdale, Ariz.

[73] Assignee: **Mobility Electronics, Inc.**, Scottsdale, Ariz.

[ * ] Notice: This patent is subject to a terminal disclaimer.

[21] Appl. No.: **09/130,058**

[22] Filed: **Aug. 6, 1998**

[51] **Int. Cl.**[7] ..... **G06F 13/40**; G06F 13/00; G06F 13/42

[52] **U.S. Cl.** ..... **710/128**; 710/129; 710/106

[58] **Field of Search** ..... 710/126–130, 710/100–102, 105–106, 1, 2

[56] **References Cited**

U.S. PATENT DOCUMENTS

| | | | |
|---|---|---|---|
| 3,800,097 | 3/1974 | Maruscak et al. | 179/98 |
| 4,413,319 | 11/1983 | Schultz et al. | 364/200 |
| 4,504,927 | 3/1985 | Callan | 364/900 |
| 4,787,029 | 11/1988 | Khan | 364/200 |
| 4,882,702 | 11/1989 | Struger et al. | 364/900 |
| 4,954,949 | 9/1990 | Rubin | 364/200 |
| 4,959,833 | 9/1990 | Mercola et al. | 371/32 |
| 4,961,140 | 10/1990 | Pechanek et al. | 364/200 |
| 5,006,981 | 4/1991 | Beltz et al. | 364/200 |
| 5,038,320 | 8/1991 | Heath et al. | 364/900 |
| 5,111,423 | 5/1992 | Kopec, Jr. et al. | 395/500 |
| 5,191,657 | 3/1993 | Ludwig et al. | 395/325 |
| 5,237,690 | 8/1993 | Bealkowski et al. | 395/700 |
| 5,274,711 | 12/1993 | Rutledge et al. | 381/47 |
| 5,325,491 | 6/1994 | Fasig | 395/325 |
| 5,335,326 | 8/1994 | Nguyen et al. . | |
| 5,335,329 | 8/1994 | Cox et al. | 395/325 |
| 5,430,847 | 7/1995 | Bradley et al. | 395/325 |
| 5,446,869 | 8/1995 | Padgett et al. | 395/500 |
| 5,457,785 | 10/1995 | Kikinis et al. | 710/128 |
| 5,475,818 | 12/1995 | Molyneaux et al. | 709/208 |
| 5,483,020 | 1/1996 | Hardie et al. | 174/36 |
| 5,495,569 | 2/1996 | Kotzur | 395/181 |
| 5,497,498 | 3/1996 | Taylor | 395/800 |
| 5,507,002 | 4/1996 | Heil | 395/828 |
| 5,517,623 | 5/1996 | Farrell et al. | 395/281 |
| 5,524,252 | 6/1996 | Desai et al. | 395/800 |
| 5,530,895 | 6/1996 | Enstrom | 395/829 |
| 5,540,597 | 7/1996 | Budman et al. | 439/77 |
| 5,542,055 | 7/1996 | Amini et al. | 395/281 |
| 5,548,730 | 8/1996 | Young et al. | 395/280 |
| 5,555,510 | 9/1996 | Verseput et al. | 364/514 R |

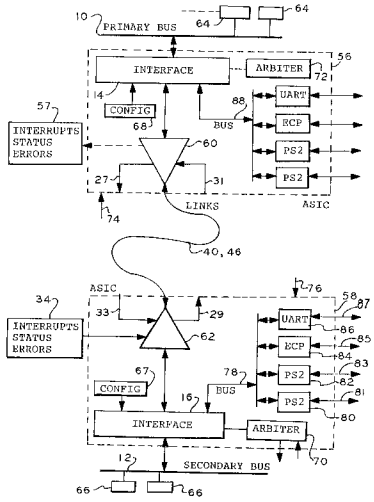
(List continued on next page.)

*Primary Examiner*—Ayaz R. Sheikh
*Assistant Examiner*—Sumati Lefkowitz
*Attorney, Agent, or Firm*—Thomas L. Adams, Esq.

[57] **ABSTRACT**

A bridge accessible by a host processor can expand access over a first bus to a second bus. The first bus and the second bus are each adapted to separately connect to respective ones of a plurality of bus-compatible devices. Allowable ones of the devices include memory devices and input/output devices. The bridge has a link, together with a first and a second interface. The first interface is coupled between the first bus and the link. The second interface is coupled between the second bus and the link. The first interface and the second interface are operable to (a) send bus-related information serially through the link in a format different from that of the first bus and the second bus, (b) approve an initial exchange between the first bus and the second bus in response to pending transactions having a characteristic signifying a destination across the bridge, (c) exchange bus-related information between the first bus and the second bus according to a predetermined hierarchy giving the first bus a higher level than the second bus, and (d) allow the host processor, communicating through the first bus, to individually address different selectable ones of the bus-compatible devices on the second bus, including memory devices and input/output devices that may be present: (i) using on the first bus substantially the same type of addressing as is used to access devices the first bus, and (ii) without first employing a second, intervening one of the bus-compatible devices on the second bus.

**81 Claims, 4 Drawing Sheets**



U.S. PATENT DOCUMENTS

| | | | |
|---|---|---|---|
| 5,561,806 | 10/1996 | Fitchett et al. | 709/250 |
| 5,572,525 | 11/1996 | Shen et al. | 370/85.1 |
| 5,572,688 | 11/1996 | Sytwu | 395/309 |
| 5,579,489 | 11/1996 | Dornier et al. | 710/101 |
| 5,586,265 | 12/1996 | Beukema | 709/217 |
| 5,590,377 | 12/1996 | Smith | 395/842 |
| 5,611,053 | 3/1997 | Wu et al. | 395/280 |
| 5,655,142 | 8/1997 | Gephardt et al. | 395/800.32 |
| 5,671,421 | 9/1997 | Kardach et al. | 710/260 |
| 5,694,556 | 12/1997 | Neal et al. | 395/308 |
| 5,696,911 | 12/1997 | Fredriksson | 710/106 |
| 5,696,949 | 12/1997 | Young | 395/551 |
| 5,701,483 | 12/1997 | Pun | 395/681 |
| 5,724,529 | 3/1998 | Smith et al. | 395/309 |
| 5,748,921 | 5/1998 | Lambrecht et al. | 710/128 |
| 5,764,924 | 6/1998 | Hong | 710/101 |
| 5,774,681 | 6/1998 | Kunishige | 710/128 |
| 5,793,995 | 8/1998 | Riley et al. | 710/128 |
| 5,799,207 | 8/1998 | Wang et al. | 710/38 |
| 5,809,262 | 9/1998 | Potter | 710/129 |
| 5,819,053 | 10/1998 | Goodrum et al. | 710/126 |
| 6,003,105 | 12/1999 | Vicard et al. | 710/129 |

FIG. 1

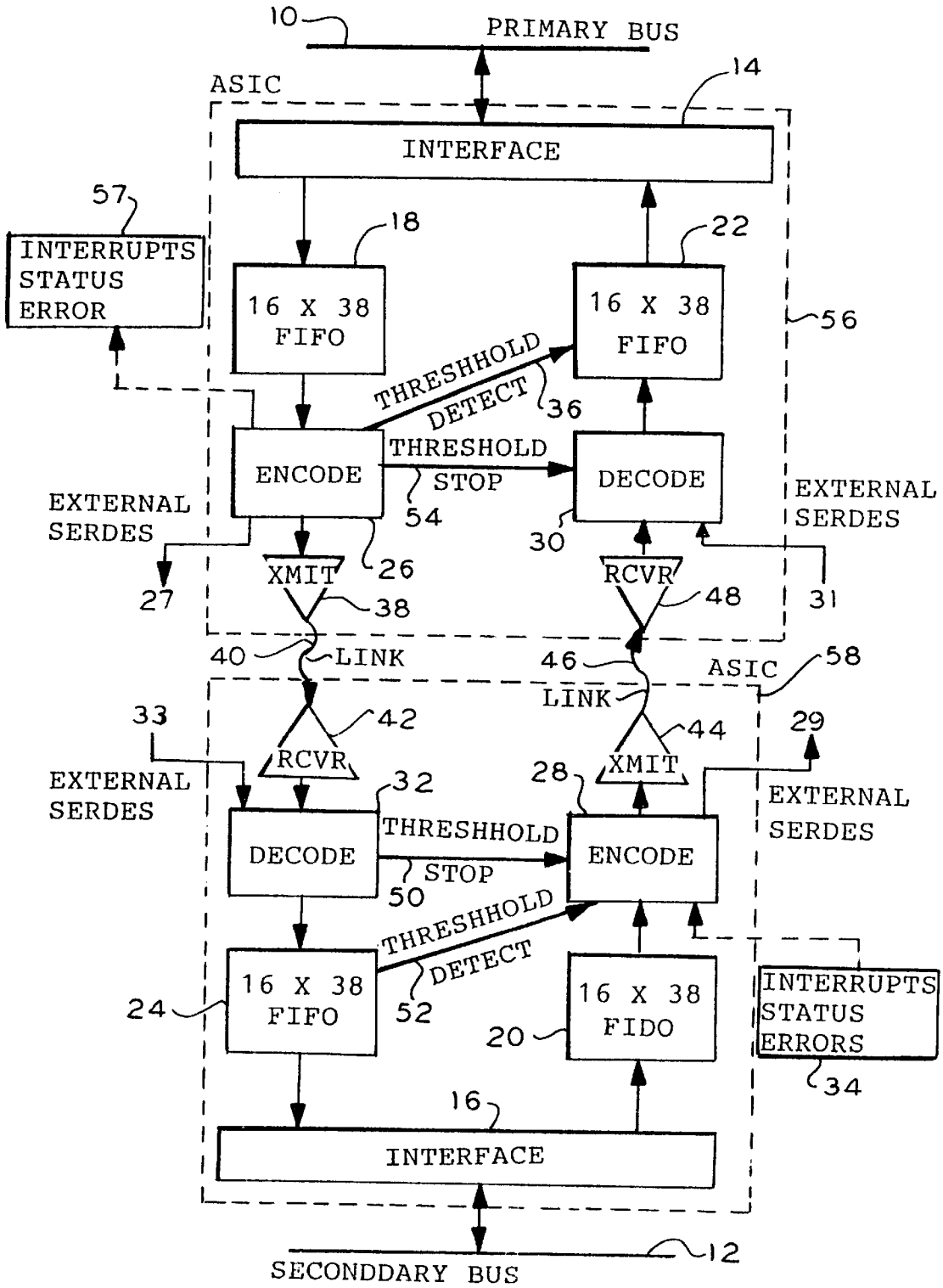
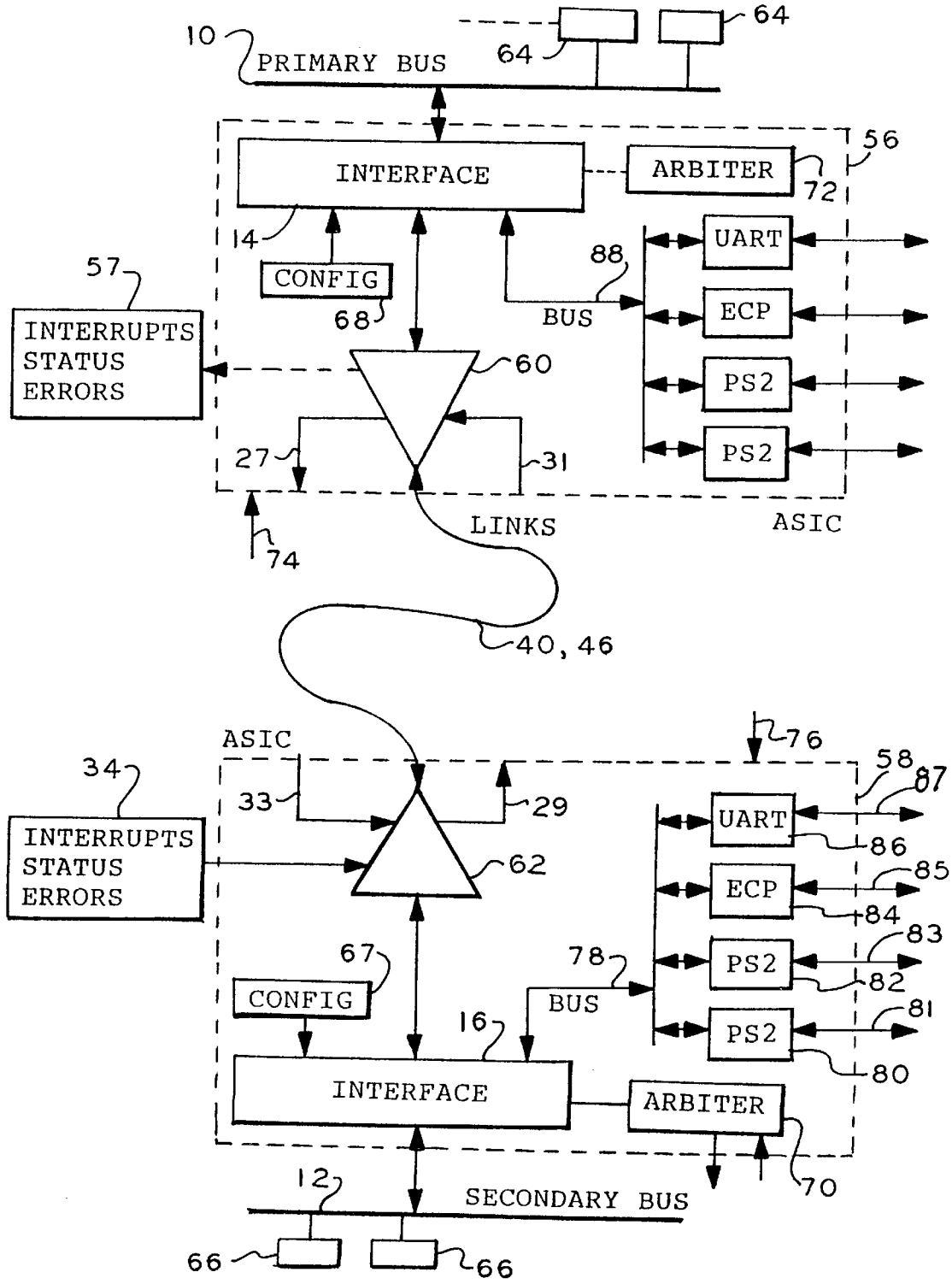

10
PRIMARY BUS
ASIC
14
INTERFACE
57
INTERRUPTS
STATUS
ERROR
18
16 X 38
FIFO
22
16 X 38
FIFO
56
THRESHHOLD
DETECT
36
ENCODE
THRESHOLD
STOP
54
DECODE
EXTERNAL
SERDES
EXTERNAL
SERDES
27
XMIT
26
30
RCVR
31
38
48
40
LINK
46
ASIC
58
LINK
33
42
44
29
RCVR
XMIT
EXTERNAL
SERDES
32
28
EXTERNAL
SERDES
DECODE
THRESHHOLD
STOP
50
ENCODE
THRESHHOLD
DETECT
24
16 X 38
FIFO
52
16 X 38
FIDO
20
INTERRUPTS
STATUS
ERRORS
34
16
INTERFACE
12
SECONDDARY BUS

FIG. 2


10
PRIMARY BUS
64
64
56
INTERFACE
ARBITER
72
14
CONFIG
68
57
INTERRUPTS
STATUS
ERRORS
88
BUS
UART
ECP
PS2
PS2
60
27
31
LINKS
ASIC
74
40, 46
ASIC
76
34
33
29
58
87
INTERRUPTS
STATUS
ERRORS
62
UART
86
85
ECP
84
83
67
78
PS2
82
CONFIG
BUS
81
16
PS2
80
INTERFACE
ARBITER
70
12
SECONDARY BUS
66
66

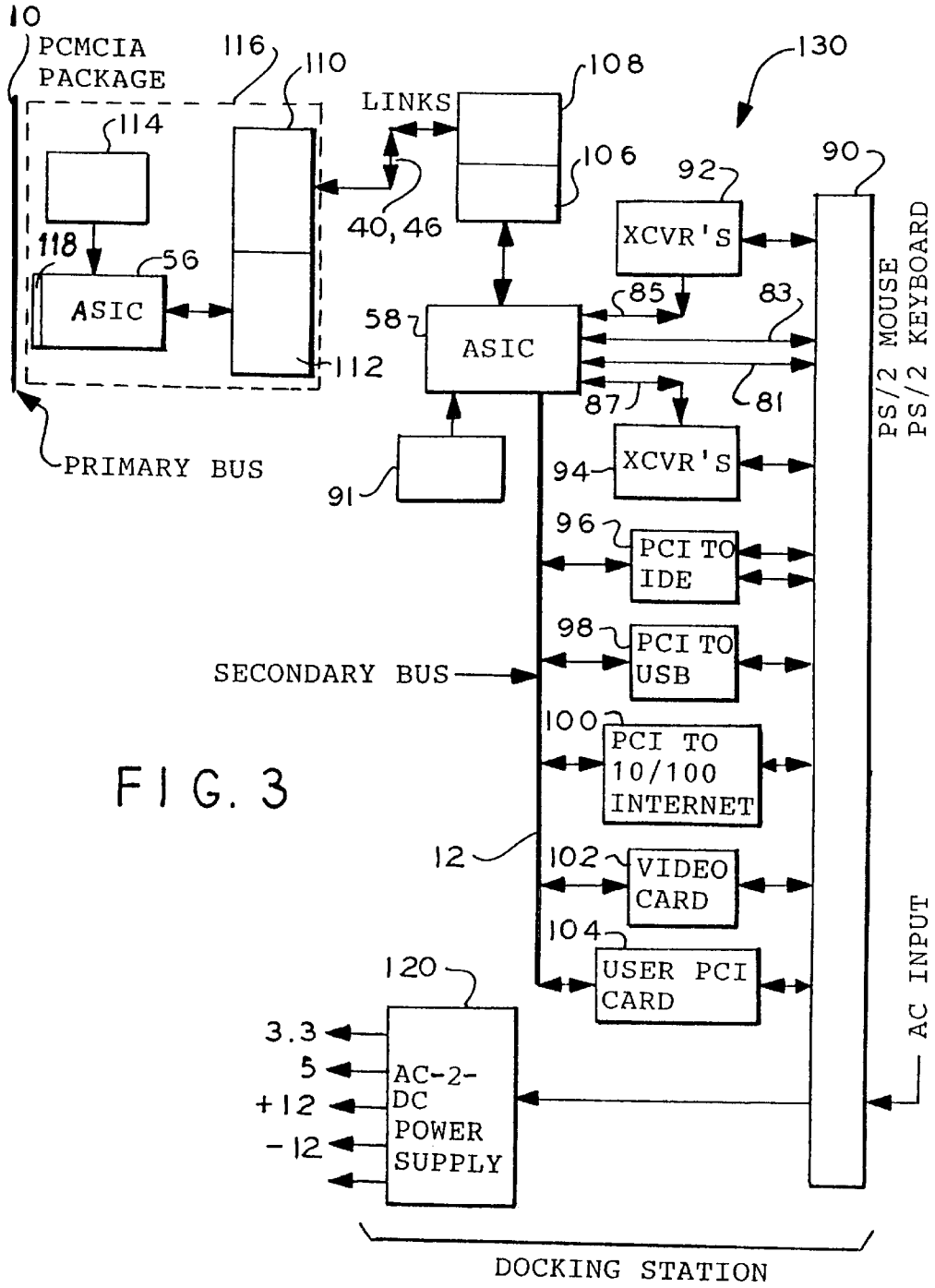

10
PCMCIA PACKAGE
116
110
114
118
56
ASIC
112
PRIMARY BUS
LINKS
108
106
40,46
130
92
XCVR'S
90
PS/2 MOUSE
PS/2 KEYBOARD
58
ASIC
85
83
87
81
91
94
XCVR'S
96
PCI TO IDE
98
PCI TO USB
SECONDARY BUS
100
PCI TO 10/100 INTERNET
12
102
VIDEO CARD
104
USER PCI CARD
120
3.3
5
+12
-12
AC-2-DC POWER SUPPLY
AC INPUT
DOCKING STATION

FIG. 3

FIG. 5

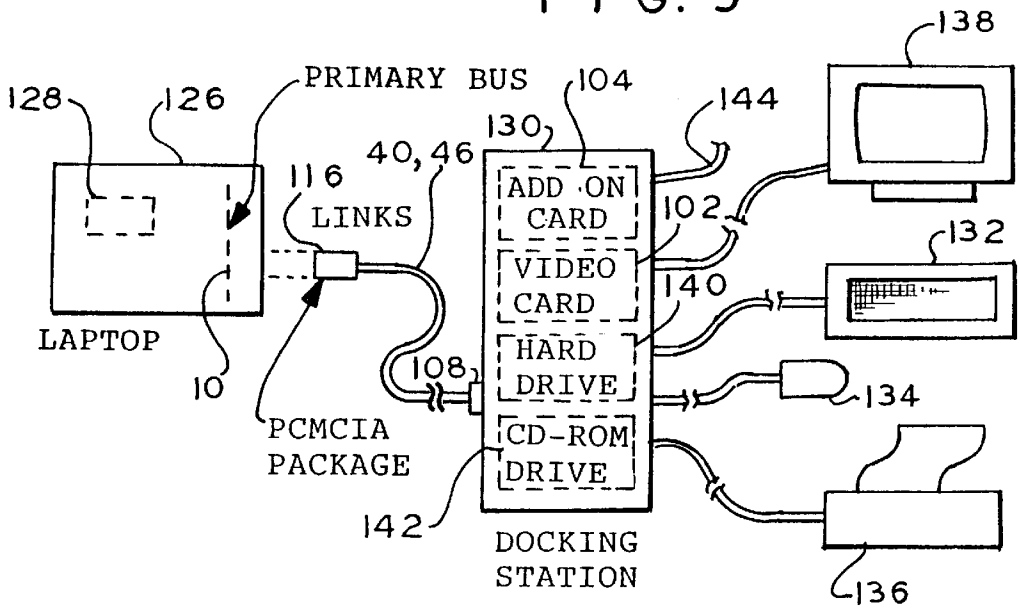



FIG. 6

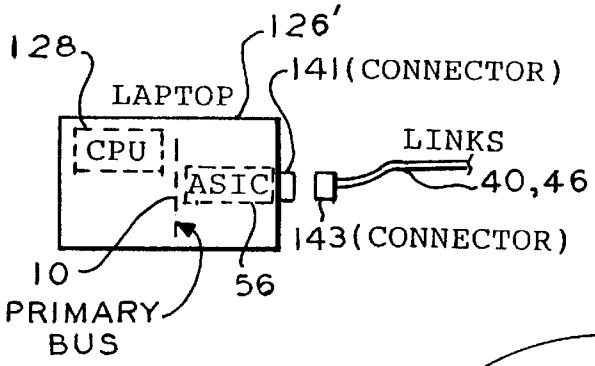
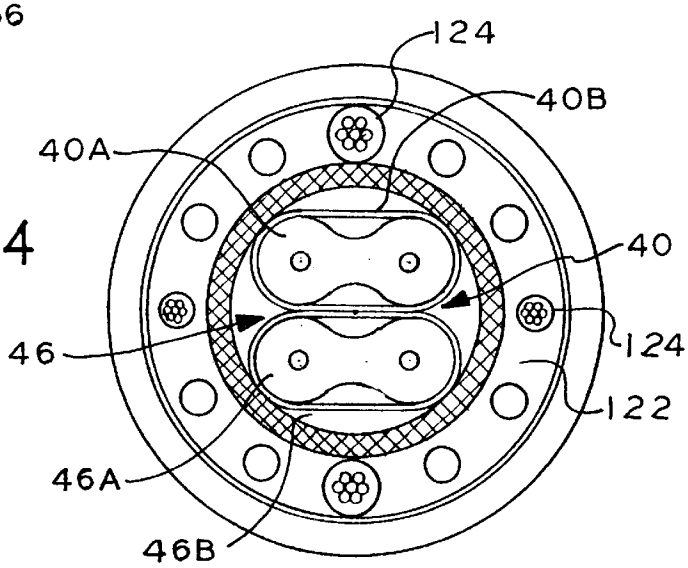



FIG. 4



# SERIALLY LINKED BUS BRIDGE FOR EXPANDING ACCESS OVER A FIRST BUS TO A SECOND BUS

## BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to data processing systems, and more particularly, to bridge systems including mechanisms for transferring information between buses.

2. Description of Related Art

Computers can use buses to transfer data between a host processor and various devices, such as memory devices and input/output devices. As used herein an "input/output" device is a device that either generates an input or receives an output (or does both). Thus "input/output" is used in the disjunctive. These buses may be arranged in a hierarchy with the host processor connected to a high level bus reserved for exchanging the data most urgently needed by the processor. Lower level buses may connect to devices having a lower priority.

Other reasons exist for providing separate buses. Placing an excessive number of devices on one bus produces high loading. Such loading makes a bus difficult to drive because of the power needed and the delays caused by signaling so many devices. Also, some devices on a bus may periodically act as a master and request control over a bus in order to communicate with a slave device. By segregating some devices on a separate bus, master devices can communicate with other devices on the lower level bus without tying up the bus used by the host processor or other masters.

The PCI bus standard is specified by the PCI Special Interest Group of Hillsboro, Oreg. The PCI bus features a 32-bit wide, multiplexed address-data (AD) bus portion, and can be expanded to a 64-bit wide AD bus portion. Maintaining a high data throughput rate (e.g., a 33 MHZ clock rate) on the PCI bus leads to a fixed limitation on the number of electrical AC and DC loads on the bus. Speed considerations also limit the physical length of the bus and the capacitance that can be placed on the bus by the loads, while future PCI bus rates (e.g., 66 MHZ) will exacerbate the electrical load and capacitance concerns. Failure to observe these load restrictions can cause propagation delays and unsynchronized operation between bus devices.

To circumvent these loading restrictions, the PCI bus standard specifies a bridge to allow a primary PCI bus to communicate with a secondary PCI bus through such a bridge. Additional loads may be placed on the secondary bus without increasing the loading on the primary bus. For bridges of various types see U.S. Pat. Nos. 5,548,730 and 5,694,556.

The PCI bridge observes a hierarchy that allows an initiator or bus master on either bus to complete a transaction with a target on the other bus. As used herein, hierarchy refers to a system for which the concept of a higher or lower level has meaning. For example, a PCI bus system is hierarchical on several scores. An ordering of levels is observed in that a high level host processor normally communicates from a higher level bus through a bridge to a lower level bus. An ordering of levels is also observed in that buses at equal levels do not communicate directly but through bridges interconnected by a higher level bus. Also, an ordering of levels is observed in that data is filtered by their addresses before being allowed to pass through a bridge, based on the levels involved. Other hierarchical systems exist that may observe an ordering of levels by



using one or more of the foregoing concepts, or by using different concepts.

Some personal computers have slots for add-on cards, which allow the card to connect to a peripheral bus in the computer. Because a user often needs additional slots, expansion cards have been designed that will connect between the peripheral bus and an external unit that offers additional slots for add-on cards. For systems for expanding a bus, see U.S. Pat. Nos. 5,006,981; 5,191,657; and 5,335,329. See also U.S. Pat. No. 5,524,252.

For portable computers, special considerations arise when the user wishes to connect additional peripheral devices. Often a user will bring a portable computer to a desktop and connect through a docking station or port replicator to a keyboard, monitor, printer or the like. A user may also wish to connect to a network through a network interface card in the docking station. At times, a user may need additional devices such as hard drives or CD-ROM drives. While technically possible to a limited extent, extending a bus from a portable computer through a cable is difficult because of the large number of wires needed and because of latencies caused by a cable of any significant length.

In U.S. Pat. No. 5,696,949 a host chassis has a PCI to PCI bridge that connects through a cabled bus to another PCI to PCI bridge in an expansion chassis. This system is relatively complicated since two independent bridges communicate over a cabled bus. This cabled bus includes essentially all of the lines normally found in a PCI bus. This approach employs a delay technique to deal with clock latencies associated with the cabled bus. A clock signal generated on the expansion side of the cabled bus: (a) is sent across the cabled bus, but experiences a delay commensurate with the cable length; and (b) is delayed an equivalent amount on the expansion side of the cabled bus by a delay line there, before being used on the expansion side. Such a design complicates the system and limits it to a tuned cable of a pre-designed length, making it difficult to accommodate work spaces with various physical layouts.

U.S. Pat. No. 5,590,377 shows a primary PCI bus in a portable computer being connected to a PCI to PCI bridge in a docking station. When docked, the primary and secondary buses are physically very close. A cable is not used to allow separation between the docking station and the portable computer. With this arrangement, there is no interface circuitry between the primary PCI bus and the docking station. See also U.S. Pat. No. 5,724,529.

U.S. Pat. No. 5,540,597 suggests avoiding additional PCMCIA connectors when connecting a peripheral device to a PC card slot in a portable computer, but does not otherwise disclose any relevant bridging techniques.

U.S. Pat. No. 4,882,702 and show a programmable controller for controlling industrial machines and processes. The system exchanges data serially with a variety of input/output modules. One of these modules may be replaced with an expansion module that can serially communicate with several groups of additional input/output modules. This system is not bridge-like in that the manner of communicating with the expansion module is different than the manner of communicating with the input/output modules. For the expansion module the system changes to a block transfer mode where a group of status bytes are transferred for all the expansion devices. This system is also limited to input/output transactions and does not support a variety of addressable memory transactions. See also U.S. Pat. Nos. 4,413,319; and 4,504,927.

In U.S. Pat. No. 5,572,525 another bus designed for instrumentation (IEEE 488 General Purpose Instrumentation

Bus) connects to an extender that breaks the bus information into packets that are sent serially through a transmission cable to another extender. This other extender reconstructs the serial packets into parallel data that is applied to a second instrumentation bus. This extender is an intelligent system operating through a message interpretation layer and several other layers before reaching the parallel to serial conversion layer. Thus this system is unlike a bridge. This system is also limited in the type of transactions that it can perform. See also U.S. Pat. No. 4,959,833.

U.S. Pat. No. 5,325,491 shows a system for interfacing a local bus to a cable with a large number of wires for interfacing with remote peripherals. See also U.S. Pat. Nos. 3,800,097; 4,787,029; 4,961,140; and 5,430,847.

The Small Computer System Interface (SCSI) defines bus standards for a variety of peripheral devices. This SCSI bus is part of an intelligent system that responds to high-level commands. Consequently, SCSI systems require software drivers to enable hardware to communicate to the SCSI bus. This fairly complicated system is quite different from bridges such as bridges as specified under the PCI standard. A variety of other complex techniques and protocols exist for transferring data, including Ethernet, Token Ring, TCP/IP, ISDN, FDDI, HIPPI, ATM, Fibre Channel, etc., but these bear little relation to bridge technology.

See also U.S. Pat. Nos. 4,954,949; 5,038,320; 5,111,423; 5,446,869; 5,495,569; 5,497,498; 5,507,002; 5,517,623; 5,530,895; 5,542,055; 5,555,510; 5,572,688; and 5,611,053.

Accordingly, there is a need for an improved system for transferring information between buses.

## SUMMARY OF THE INVENTION

In accordance with the illustrative embodiments demonstrating features and advantages of the present invention, there is provided a bridge accessible by a host processor for expanding access over a first bus to a second bus. The first bus and the second bus are each adapted to separately connect to respective ones of a plurality of bus-compatible devices. Allowable ones of the devices include memory devices and input/output devices. The bridge has a link, together with a first and a second interface means. The first interface means is coupled between the first bus and the link. The second interface means is coupled between the second bus and the link. The first interface means and the second interface means are operable to (a) send bus-related information serially through the link in a format different from that of the first bus and the second bus, (b) approve an initial exchange between the first bus and the second bus in response to a pending transaction having a characteristic signifying a destination across the bridge, and (c) allow the host processor, communicating through the first bus, to individually address different selectable ones of the bus-compatible devices on the second bus, including memory devices and input/output devices that may be present: (i) using on the first bus substantially the same type of addressing as is used to access devices the first bus, and (ii) without first employing a second, intervening one of the bus-compatible devices on the second bus.

In accordance with another aspect of the invention a bridge accessible by a host processor can expand access over a first bus to a second bus. The first bus and the second bus each are adapted to separately connect to respective ones of a plurality of bus-compatible devices. Allowable ones of the devices include memory devices and input/output devices. The bridge has a link, together with a first and a second interface means. The first interface means is coupled between the first bus and the link. The second interface means is coupled between the second bus and the link. The first interface means and the second interface means are operable to (a) send bus-related information serially through the link in a format different from that of the first bus and the second bus, (b) exchange bus-related information between the first bus and the second bus according to a predetermined hierarchy giving the first bus a higher level than the second bus, and (c) allow the host processor, communicating through the first bus, to individually address different selectable ones of the bus-compatible devices on the second bus, including memory devices and input/output devices that may be present: (i) using on the first bus substantially the same type of addressing as is used to access devices the first bus, and (ii) without first employing a second, intervening one of the bus-compatible devices on the second bus.

By employing apparatus and methods of the foregoing type, an improved system is achieved for transferring information between buses. In one preferred embodiment, two buses communicate over a duplex link formed with a pair of simplex links, each employing twisted pair or twin axial lines (depending on the desired speed and the anticipated transmission distance). Information from the buses are first loaded onto FIFO (first-in first-out) registers before being serialized into frames for transmission over the link. Received frames are deserialized and loaded into FIFO registers before being placed onto the destination bus. Preferably, interrupts, error signals, and status signals are sent along the link.

In this preferred embodiment, address and data are taken from a bus one transaction at a time, together with four bits that act either as control or byte enable signals. Two or more additional bits may be added to tag each transaction as either: an addressing cycle; acknowledgment of a non-posted write; data burst; end of data burst (or single cycle). If these transactions are posted writes they can be rapidly stored in a FIFO register before being encoded into a number of frames that are sent serially over a link. When pre-fetched reads are allowed, the FIFO register can store pre-fetched data in case the initiator requests it. For single cycle writes or other transactions that must await a response, the bridge can immediately signal the initiator to wait, even before the request is passed to the target.

In a preferred embodiment, one or more of the buses follows the PCI or PCMCIA bus standard (although other bus standards can be used instead). The preferred apparatus then operates as a bridge with a configuration register that is loaded with information specified under the PCI standard. The apparatus can transfer information between buses depending upon whether the pending addresses fall within a range embraced by the configuration registers. This scheme works with devices on the other side of the bridge, which can be given unique base addresses to avoid addressing conflicts.

In one highly preferred embodiment, the apparatus may be formed as two separate application-specific integrated circuits (ASIC) joined by a cable. Preferably, these two integrated circuits have the same structure, but can act in two different modes in response to a control signal applied to one of its pins. Working with hierarchical buses (primary and secondary buses) these integrated circuits will be placed in a mode appropriate for its associated bus. The ASIC associated with the secondary bus preferably has an arbiter that can grant masters control of the secondary bus. This preferred ASIC can also supply a number of ports to support a mouse and keyboard, as well as parallel and serial ports.

When used with a portable computer, one of the ASIC's can be assembled with a connector in a package designed to

fit into a PC card slot following the PCMCIA standard. This ASIC can connect through a cable to the other ASIC, which can be located in a docking station. Accordingly, the apparatus can act as a bridge between a CardBus and a PCI bus located in a docking station. Since the preferred ASIC can also provide a port for a mouse and keyboard, this design is especially useful for a docking station. Also, the secondary PCI bus implemented by the ASIC can connect to a video card or to a video processing circuit on the main dock circuit board in order to drive a monitor.

In some embodiments, one ASIC will be mounted in the portable computer by the original equipment manufacturer (OEM). This portable computer will have a special connector dedicated to the cable that connects to the docking station with the mating ASIC. For such embodiments, the existence within the preferred ASIC of ports for various devices can be highly advantageous. An OEM can use this already existing feature of the ASIC and thereby eliminate circuitry that would otherwise have been needed to implement such ports.

## BRIEF DESCRIPTION OF THE DRAWINGS

The above brief description as well as other objects, features and advantages of the present invention will be more fully appreciated by reference to the following detailed description of presently preferred but nonetheless illustrative embodiments in accordance with the present invention when taken in conjunction with the accompanying drawings, wherein:

FIG. **1** is a schematic block diagram showing a bridge split by a link within the bridge, in accordance with principles of the present invention;

FIG. **2** is a schematic block diagram showing a bridge in accordance with principles of the present invention using the link of FIG. **1**;

FIG. **3** is a schematic block diagram showing the bridge of FIG. **2** used in a docking system in accordance with principles of the present invention;

FIG. **4** is a cross-sectional view of the cable of FIG. **3**;

FIG. **5** is a schematic illustration of the bridge of FIG. **3** shown connected to a portable computer and a variety of peripheral devices; and

FIG. **6** shows a docking station similar to that of FIG. **5** but with the portable computer modified to contain an application-specific integrated circuit designed to support a link to the docking station.

## DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

Referring to FIG. **1**, a bridge is shown connecting between a first bus **10** and a second bus **12** (also referred to as primary bus **10** and secondary bus **12**). These buses may be PCI or PCMCIA 32-bit buses, although other types of buses are contemplated and the present disclosure is not restricted to any specific type of bus. Buses of this type will normally have address and data lines. In some cases, such as with the PCI bus, address and data are multiplexed onto the same lines. In addition, these buses will have signaling lines for allowing devices on the bus to negotiate transactions. For the PCI standard, these signaling lines will include four lines that are used either for control or byte enabling (C/BE[3:0]). Others signaling lines under the PCI standard exist for gaining control over the bus, for handshaking, and the like (e.g., FRAME#, TRDY#, IRDY#, STOP#, DEVSEL#, etc.)

Buses **10** and **12** are shown connecting to a first interface means **14** and second interface means **16**, respectively (also referred to as interfaces **14** and **16**). Bus information selected for transmission by interfaces **14** and **16** are loaded into registers **18** and **20**, respectively. Incoming bus information that interfaces **14** and **16** select for submission to the buses are taken from registers **22** and **24**, respectively. In one embodiment, registers **18–24** are each 16×38 FIFO registers, although different types of registers having different dimensions may be used in alternate embodiments.

In this embodiment, registers **18–24** are at least 38 bits wide. Thirty six of those bits are reserved for the 4 control bits (C/BE#[3:0]) and the 32 address/data bits (AD[31:0]) used under the PCI bus standard. The remaining two bits can be used to send additional tags for identifying the nature of the transaction associated therewith. Other bits may be needed to fully characterize every contemplated transaction. Transactions can be tagged as: addressing cycle; acknowledgment of a non-posted write; data burst; end of data burst (or single cycle). Thus outgoing write transactions can be tagged as a single cycle transaction or as part of a burst. Outgoing read requests can also be tagged as part of a burst with a sequence of byte enable codes (C/BE) for each successive read cycle of the burst. It will be appreciated that other coding schemes using a different number of bits can be used in other embodiments.

The balance of the structure illustrated in FIG. **1** is a link designed to establish duplex communications between interfaces **14** and **16** through registers **18–24**. For example, encoder **28** can accept the oldest 38 bits from register **20** and parse it into five bytes (40 bits). The extra two bits of the last byte are encoded to signify the interrupts, status signals and error signals that may be supplied from block **34**.

Each of these five bytes is converted into a 10 bit frame that can carry the information of each byte, as well as information useful for regulating the link. For example, these frames can carry comma markers, idle markers, or flow control signals, in a well-known fashion. A transceiver system working with bytes that were encoded into such 10 bit frames is sold commercially by Hewlett Packard as model number HDMP-1636 or -1646. Frames produced by encoder **28** are forwarded through transmitter **44** along simplex link **46** to receiver **48**, which supplies the serial information to decoder **30**. Likewise, encoder **26** forwards serial information through transmitter **38** along simplex link **40** to receiver **42**, which supplies the serial information to decoder **32**.

Flow control may be necessary should FIFO registers **22** or **24** be in danger of overflowing. For example, if FIFO register **22** is almost full, it supplies a threshold detect signal **36** to encoder **26**, which forwards this information through link **40** to decoder **32**. In response, decoder **32** issues a threshold stop signal **50** to encoder **28**, which then stops forwarding serial information, thereby preventing an overflow in FIFO register **22**. In a similar fashion, a potential overflow in FIFO register **24** causes a threshold detect signal **52** to flow through encoder **28** and link **46** to cause decoder **30** to issue a threshold stop signal **54**, to stop encoder **26** from sending more frames of information. In some embodiments, the system will examine the received information to determine if it contains transmission errors or has been corrupted in some fashion. In such event the system can request a retransmission of the corrupted information and thereby ensure a highly reliable link.

In this embodiment, elements **14**, **18**, **22**, **26**, **30**, **38** and **48** are part of a single, application specific integrated circuit (ASIC) **56**. Elements **16**, **20**, **24**, **28**, **32**, **42** and **44** are also part of an ASIC **58**. As described further hereinafter, first

ASIC **56** and second ASIC **58** have an identical structure but can be operated in different modes. It will be appreciated that other embodiments may not use ASIC's but may use instead alternate circuitry, such as a programable logic device, or the like. As shown herein, ASIC **56** is operating in a mode designed to service primary bus **10**, and (for reasons to be described presently) will be sending outputs to block **57**. In contrast block **34** of ASIC **58** will receive inputs from block **34**.

Encoders **26** and **28** have optional parallel outputs **27** and **29**, respectively, for applications requiring such information. Also for such applications, decoders **30** and **32** have parallel inputs **31** and **33**, respectively. These optional inputs and outputs may be connected to an external transceiver chip, such as the previously mentioned device offered by Hewlett Packard as model number HDMP-1636 or -1646. These devices will still allow the system to transmit serial information, but by means of an external transceiver chip. This allows the user of the ASIC's **56** and **58** more control over the methods of transmission over the link.

Referring to FIG. **2**, previously mentioned ASIC's **56** and **58** are shown in further detail. The previously mentioned encoders, decoders, transmitters, receivers, and FIFO registers are combined into blocks **60** and **62**, which are interconnected by a duplex cable formed of previously mentioned simplex links **40** and **46**. Previously mentioned interface **14** is shown connected to primary bus **10**, which is also connected to a number of bus-compatible devices **64**. Similarly, previously mentioned interface **16** is shown connected to secondary bus **12**, which is also connected to a number of bus-compatible devices **66**. Devices **64** and **66** may be PCI-compliant devices and may operate as memory devices or input/output devices.

Interface **14** a shown connected to a first register means **68**, which acts as a configuration register in compliance with the PCI standard. Since this system will act as a bridge, configuration registers **68** will have the information normally associated with a bridge. Also, configuration registers **68** will contain a base register and limit register to indicate a range or predetermined schedule of addresses for devices that can be found on the secondary bus **12**. Under the PCI standard, devices on a PCI bus will themselves each have a base register, which allows mapping of the memory space and/or I/O space. Consequently, the base and limit registers in configuration registers **68** can accommodate the mapping that is being performed by individual PCI devices. The information on configuration registers **68** are mirrored on second configuration register **67** (also referred to as a second configuration means). This makes the configuration information readily available to the interfaces on both sides of the link.

In this embodiment, ASIC **58** has an arbiter **70**. Arbiters are known devices that accept requests from masters on secondary bus **12** for control of the bus. The arbiter has a fair algorithm that grants the request of one of the contending masters by issuing it a grant signal. In this hierarchical scheme, secondary bus **12** requires bus arbitration, but primary bus **10** will provide its own arbitration. Accordingly, ASIC **56** is placed in a mode where arbiter **72** is disabled. The modes of ASIC's **56** and **58** are set by control signals applied to control pins **74** and **76**, respectively. Because of this mode selection, the signal directions associated with blocks **57** and **34** will be reversed.

In this embodiment, ASIC **58** is in a mode that implements a third bus **78**. Bus **78** may follow the PCI standard, but is more conveniently implemented in a different standard. Bus **78** connects to a number of devices that act as a port means. For example, devices **80** and **82** can implement PS/2 ports that can connect to either a mouse or a keyboard. Device **84** implements an ECP/EPP parallel port for driving a printer or other device. Device **86** implements a conventional serial port. Devices **80**, **82**, **84** and **86** are shown with input/output lines **81**, **83**, **85** and **87**, respectively. Devices **80–86** may be addressed on bus **10** as if they were PCI devices on bus **12**. Also in this embodiment, a bus **88** is shown in ASIC **56**, with the same devices as shown on bus **78** to enable an OEM to implement these ports without the need for separate input/output circuits.

Referring to FIG. **3**, previously mentioned ASIC **58** is shown in a docking station **130** connected to an oscillator **91** for establishing a remote and internal clock. ASIC **58** has its lines **81** and **83** connected through a connection assembly **90** for connection to a keyboard and mouse, respectively. Serial lines **85** and parallel lines **87** are shown connected to transceivers **92** and **94**, respectively, which then also connect to connection assembly **90** for connection to various parallel and serial peripherals, such as printers and modems.

ASIC **58** is also shown connected to previously mentioned secondary bus **12**. Bus **12** is shown connected to an adapter card **96** to allow the PCI bus **12** to communicate with an IDE device such as a hard drive, backup tape drive, CD-ROM drive, etc. Another adapter card **98** is shown for allowing communications from bus **12** to a universal serial port (USB). A network interface card **100** will allow communications through bus **12** to various networks operating under the Ethernet standard, Token Ring standard, etc. Video adapter card **102** (also referred to as a video means) allows the user to operate another monitor. Add-on card **104** may be one of a variety of cards selected by the user to perform a useful function. While this embodiment shows various functions being implemented by add-on cards, other embodiments may implement one or more of these function on a common circuit board in the dock (e.g., all functions excluding perhaps the IDE adapter card).

ASIC **58** communicates through receiver/transmitter **106**, which provides a physical interface through a terminal connector **108** to cable **40**, **46**. Connector **108** may be a **20** pin connector capable of carrying high speed signals with EMI shielding (for example a low force helix connector of the type offered by Molex Incorporated), although other connector types may used instead. The opposite end of cable **40**, **46** connects through a gigabit, terminal connector **110** to physical interface **112**, which acts as a receiver/transmitter. Interface **112** is shown connected to previously mentioned first ASIC **56**, which is also shown connected to an oscillator **114** to establish a local clock signal. This specific design contemplates using an external transmitter/receiver (external SERDES of lines **27**, **29**, **31**, and **33** of FIG. **1**), although other embodiments can eliminate these external devices in favor of the internal devices in ASIC's **56** and **58**.

This embodiment is adapted to cooperate with a portable computer having a PCMCIA 32-bit bus **10**, although other types of computers can be serviced. Accordingly, ASIC **56** is shown in a package **116** having an outline complying with the PCMCIA standard and allowing package **116** to fit into a slot in a portable computer. Therefore, ASIC **56** has a connector **118** for connection to bus **10**. Cable **40**, **46** will typically be permanently connected to package **116**, but a detachable connector may be used in other embodiments, where a user wishes to leave package **116** inside the portable computer.

Power supply **120** is shown producing a variety of supply voltages used to power various components. In some

embodiments, one of these supply lines can be connected directly to the portable computer to charge its battery.

Referring to FIG. **4**, the previously mentioned simplex links **40** and **46** are shown as twin axial lines **40**A and **46**A, wrapped with individual shields **40**B and **46**B. A single shield **122** encircles the lines **40** and **46**. Four parallel wires **124** are shown (although a greater number may be used in other embodiments) mounted around the periphery of shields **122** for various purposes. These wires **124** may carry power management signals, dock control signals or other signals that may be useful in an interface between a docking station and a portable computer. While twin axial lines offer high performance, twisted pairs or other transmission media may be used in other embodiments where the transmission distance is not as great and where the bit transfer speed need not be as high. While a hard wire connection is illustrated, in other embodiments a wireless or other type of connection can be employed instead.

Referring to FIG. **5**, previously mentioned package **116** is shown in position to be connected to a PCMCIA slot in portable computer **126**. Computer **126** is shown having primary bus **10** and a host processor **128**. Package **116** is shown connected through cable **40**, **46** to previously mentioned connector **108** on docking station **130**. Previously mentioned docking station **130** is shown connecting through PS/2 ports to keyboard **132** and mouse **134**. A printer **136** is shown connected to a parallel port in docking station **130**. Previously mentioned video means **102** is shown connected to a monitor **138**. Docking station **130** is also shown with an internal hard drive **140** connecting to the adapter card previously mentioned. A CD-ROM drive **142** is also shown mounted in docking station **130** and connects to the secondary bus through an appropriate adapter card (not shown). Previously mentioned add-on card **104** is shown with its own cable **144**.

Referring to FIG. **6**, a modified portable computer **126'** is again shown with a host processor **128** and primary bus **10**. In this embodiment however, portable computer **126'** contains previously mentioned ASIC **56**. Thus there is no circuitry required (other than perhaps drivers) between ASIC **56** and cable **40**,**46**. In this case, the laptop end of cable **40**, **46** has a connector **142** similar to the one on the opposite end of the cable (connector **108** of FIG. **5**). Connector **143** is designed to mate with connector **141**and support the high-speed link. As before, connectors **141** and **143** can also carry various power management signals, and other signals associated with a docking system.

An important advantage of this arrangement is the fact that ASIC **56** contains circuitry for providing ports, such as a serial port, a parallel port, PS/2 ports for a mouse and keyboard, and the like. Since portable computer **126'** would ordinarily provide such ports, ASIC **56** simplifies the design of the portable computer. This advantage is in addition to the advantage of having a single ASIC design (that is, ASIC's **56** and **58** are structured identically), which single design is capable of operating at either the portable computer or the docking station, thereby simplifying the ASIC design and reducing stocking requirements, etc.

To facilitate an understanding of the principles associated with the foregoing apparatus, its operation will be briefly described. This operation will be described in connection with the docking system of FIGS. **3** and **5** (which generally relates to FIG. **2**), although operation would be similar for other types of arrangements. For the docking system, a connection is established by plugging package **116** (FIG. **5**) into portable computer **126**. This establishes a link between the primary bus **10** and ASIC **56** (FIG. **3**).

At this time an initiator (the host processor or a master) having access to primary bus **10** may assert control of the bus. An initiator will normally send a request signal to an internal arbiter (not shown) that will eventually grant control to this initiator. In any event, the initiator asserting control over primary bus **10** will exchange the appropriate handshaking signals and drive an address onto the bus **10**. Control signals simultaneously applied to the signaling lines of bus **10** will indicate whether the transaction is a read, write, or other type of transaction.

Interface **14** (FIG. **2**) will examine the pending address and determine whether it represents a transaction with devices on the other side of the bridge (that is, secondary bus **12**) or with the bridge itself. Configuration register **68** has already been loaded in the usual manner with information that indicates a range of addresses defining the jurisdiction of the interface **14**.

Assuming a write transaction is pending on bus **10**, interface **14** will transfer 32 address bits together with four control bits (PCI standard) to FIFO register **18** (FIG. **1**). Encoder **26** will add at least two additional bits tagging this information as an addressing cycle. The information is then broken into frames that can carry flow control and other signals before being transmitted serially over link **40**.

Without waiting, interface **14** will proceed to a data cycle and accept up to 32 bits of data from bus **10** together with four byte enable bits. As before, this information will be tagged, supplemented with additional information and broken into frames for serial transmission over link **40**. This transmitted information will be tagged to indicate whether it is part of a burst or a single cycle.

Upon receipt, decoder **32** restores the frames into the original 38 bit format and loads the last two described cycles onto the stack of register **24**. Interface **16** eventually notices the first cycle as an addressing cycle in a write request. Interface **16** then negotiates control over bus **12** in the usual fashion and applies the address to bus **12**. A device on bus **12** will respond to the write request by performing the usual handshaking.

Next, interface **16** will drive the write data stacked on register **24** into bus **12**. If this transaction is a burst, interface **16** will continue to drive data onto bus **12** by fetching it from register **24**. If however this transaction is a single cycle write, interface **16** will close the transaction on bus **12** and load an acknowledgment into register **20**. Since this acknowledgment need not carry data or address information, a unique code may be placed into register **20**, so that encoder **28** can appropriately tag this line before parsing it into frames for transmission over link **46**. Upon receipt, decoder **30** will produce a unique code that is loaded into register **22** and eventually forwarded to interface **14**, which sends an acknowledgment to the device on bus **10** that the write has succeeded.

If the initiator instead sets its control bits during the address cycle to indicate a read request, interface **14** would also accept this cycle, if it has jurisdiction. Interface **14** will also signal the initiator on bus **10** that it is not ready to return data (e.g., a retry signal, which may be the stop signal as defined under the PCI standard). The initiator can still start (but not finish) a data cycle by driving its signaling lines on bus **10** with byte enable information. Using the same technique, the address information, followed by the byte enable information, will be accepted by interface **14** and loaded with tags into register **18**. These two lines of information will be then encoded and transmitted serially over link **40**. Upon receipt, this information will be loaded into

the stack of register **24**. Eventually, interface **16** will notice the first item as a read request and drive this address information onto secondary bus **12**. A device on bus **12** will respond and perform the appropriate handshaking. Interface **16** will then forward the next item of information from register **24** containing the byte enables, onto bus **12** so the target device can respond with the requested data. This responsive data is loaded by interface **16** into register **20**. If pre-fetching is indicated, interface **16** will initiate a number of successive read cycles to accumulate data in register **20** from sequential addresses that may or may not be requested by the initiator.

As before, this data is tagged, broken into frames and sent serially over link **46** to be decoded and loaded into register **22**. The transmitted data can include pre-fetched data that will be accumulated in register **22**. Interface **14** transfers the first item of returning data onto primary bus **10**, and allows the initiator to proceed to another read cycle if desired. If another read cycle is conducted as part of a burst transaction, the requested data will already be present in register **22** for immediate delivery by interface **14** to bus **10**. If these pre-fetched data are not requested for the next cycle, then they are discarded.

Eventually the initiator will relinquish control of bus **10**. Next, an initiator on bus **12** may send a request for control of bus **12** to arbiter **70** (FIG. **2**). If arbiter **70** grants control, the initiator may make a read or write request by driving an address onto bus **12**. Interface **16** will respond if this address does not fall within the jurisdictional range of addresses specified in configuration register **67** (indicating the higher level bus **10** may have jurisdiction). In the same manner as before, but with a reversed flow over links **40**, **46**, interface **16** may accept address and data cycles and communicate them across link **40**, **46**. Before being granted bus **10**, interface **14** will send a request to an arbiter (not shown) associated with bus **10**.

In some instances, an initiator on primary bus **10** will wish to read from, or write to, port means **80**, **82**, **84**, or **86**. These four items are arranged to act as devices under the PCI standard. Interface **16** will therefore act as before, except that information will be routed not through bus **12**, but through bus **78**.

Other types of transactions may be performed, including reads and writes to the configuration registers **67** and **68** (FIG. **2**). Other types of transactions, as defined under the PCI standard (or other bus standards) may be performed as well.

Interrupt signals may be generated by the ports or other devices in ASIC **58**. Also external interrupts may be received as indicated by block **34**. As noted before, interrupt signals may be embedded in the code sent over link **46**.

Upon receipt, system **60** decodes the interrupts and forwards them on to block **57**, which may be simply one or more pins from ASIC **56** (implementing, for example, INTA of the PCI standard). This interrupt signal can either be sent over the bus **10** or to an interrupt controller that forwards interrupts to the host processor. System errors may be forwarded in a similar fashion to produce an output on a pin of ASIC **56** that can be routed directly to bus **10** or processed using dedicated hardware. The designer may wish to send individual status signals, which can be handled in a similar fashion along link **40**, **46**.

It is appreciated that various modifications may be implemented with respect to the above described, preferred embodiment. In other embodiments the illustrated ASIC's may be divided into several discrete packages using in some cases commercially available integrated circuits. Also, the media for the link may be wire, fiber-optics, infrared light, radio frequency signals, or other media. In addition, the primary and secondary buses may each have one or more devices, and these devices may be in one or more categories, including memory devices and input/output devices. Moreover, the devices may operate at a variety of clock speeds, bandwidths and data rates. Furthermore, transactions passing through the bridge may be accumulated as posted writes or as pre-fetched data, although some embodiments will not use such techniques. Also, the bridge described herein can be part of a hierarchy using a plurality of such bridges having their primary side connected to the same bus or to buses of an equivalent or different level. Additionally, the illustrated ports can be of a different number or type, or can be eliminated in some embodiments. Also, the illustrated arbiter can be eliminated for secondary buses that are not design to be occupied by a master. While a sequence of steps is described above, in other embodiments these steps may be increased or reduced in number, or performed in a different order, without departing from the scope of the present invention.

Obviously, many modifications and variations of the present invention are possible in light of the above teachings. It is therefore to be understood that within the scope of the appended claims, the invention may be practiced otherwise than as specifically described.

What is claimed is:

**1**. A bridge accessible by a host processor for expanding access over a first bus to a second bus, said first bus and said second bus each being adapted to separately connect to respective ones of a plurality of bus-compatible devices, allowable ones of said devices including memory devices and input/output devices, said bridge comprising:

a link;

a first interface adapted to couple between said first bus and said link; and

a second interface adapted to couple between said second bus and said link, said first interface and said second interface operating as a single bridge and being operable to (a) send outgoing, information serially through said link in a format different from that of said first bus and said second bus without waiting for an incoming acknowledgment over said link before inaugurating a transfer of said information over said link, (b) approve an initial exchange between said first bus and said second bus in response to pending bus transactions having a characteristic signifying a destination across said bridge, and (c) allow said host processor, communicating through said first bus, to individually address different selectable ones of the bus-compatible devices on said second bus, including memory devices and input/output devices that may be present: (i) using on said first bus substantially the same type of addressing as is used to access devices on said first bus, and (ii) without first employing a second, intervening one of the bus-compatible devices on said second bus.

**2**. A bridge according to claim **1** wherein said first interface and said second interface are operable to exchange information between said first bus and said second bus according to a predetermined hierarchy giving said first bus a higher level than said second bus.

**3**. A bridge according to claim **1** wherein said first bus and said second bus each have a plurality of signaling lines for enabling bus-compatible devices to negotiate bus communications, said first interface being operable in response to a pending transaction on said first bus to begin

processing said pending transaction and to apply a retry signal to at least one of said signaling lines of said first bus before the pending transaction on said first bus has been transmitted to and acknowledged by said second bus.

**4**. A bridge according to claim **3** wherein less than all of the information on the signaling lines of said first bus is transmitted by said first interface means over said link.

**5**. A bridge according to claim **3** wherein said first interface comprises:

a first pair of first in first out registers for stacking information for successive groups of information gathered from and destined for said first bus.

**6**. A bridge according to claim **5** wherein said second interface comprises:

a second pair of first in first out registers for stacking information for successive groups of information gathered from and destined for said second bus.

**7**. A bridge according to claim **1** wherein said first interface is selectively responsive to those addresses appearing on said first bus that are on a predetermined schedule of addresses corresponding to the bus-compatible devices accessible through said second bus, in order to avoid responding to adresses corresponding to other ones of the bus-compatible devices on said first bus.

**8**. A bridge according to claim **7** comprising:

a register for storing said predetermined schedule.

**9**. A bridge according to claim **7** wherein said first interface comprises:

a first register for storing said predetermined schedule, said second interface comprising:

a second register for storing said predetermined schedule.

**10**. A bridge according to claim **8** wherein said register is operable to establish with respect to said first bus a base address for one or more of the bus-compatible devices on said second bus.

**11**. A bridge according to claim **1** comprising:

a register for establishing with respect to said first bus a base address for one or more of the bus-compatible devices on said second bus.

**12**. A bridge according to claim **1** wherein said first interface and said second interface are operable to permit communication between bus-compatible devices on said second bus without routing through said first bus.

**13**. A bridge according to claim **12** comprising:

an arbiter with authority to grant the second bus and without authority to grant the first bus to either said second interface or one of the bus-compatible devices on said second bus.

**14**. A bridge according to claim **1** wherein said first interface and said second interface comprise:

a first and a second programmable logic device connected between said link and said first bus and said second bus, respectively.

**15**. A bridge according to claim **1** wherein said first interface and said second interface comprise:

a first and a second application-specific integrated circuit connected between said link and said first bus and said second bus, respectively.

**16**. A bridge according to claim **15** wherein said first and said second application-specific integrated circuit are identically structured and each have a control pin for receiving a control signal to establish operation in one of two modes.

**17**. A bridge according to claim **16** wherein said first and said second application-specific integrated circuit each comprise:

an arbiter, enabled only in said second application-specific integrated circuit, with authority to grant the second bus and without authority to grant the first bus to either said second interface or one of the bus-compatible devices on said second bus.

**18**. A bridge according to claim **15** wherein said first and said second application-specific integrated circuit each comprise:

a plurality of port coupled to said second interface for providing a plurality of ports for input/output.

**19**. A bridge according to claim **1** wherein said host processor is interrupt-driven, said second interface being operable to transmit through said link to said first interface interrupt signals destined to interrupt the host processor.

**20**. A bridge according to claim **19** wherein said host processor is responsive to error signals, said second interface being operable to transmit through said link to said first interface error signals destined to affect the host processor.

**21**. A bridge according to claim **1** wherein said first bus operates at a predetermined clock speed, said link being operable to propagate data between said first interface and said second interface at a bit transfer rate greater than said predetermined clock speed.

**22**. A bridge according to claim **21** wherein said link comprises:

a pair of simplex links for sending information in opposite directions.

**23**. A bridge according to claim **22** wherein said simplex links are driven for differential signal transfers.

**24**. A bridge according to claim **22** wherein each of said simplex links comprises a cable with twin axial lines.

**25**. A bridge according to claim **22** wherein each of said simplex links comprises a twisted pair.

**26**. A bridge according to claim **24** comprising:

a single shield around both of said simplex links; and

a separate pair of individual shields around each of said simplex links.

**27**. A bridge according to claim **1** wherein said second bus comprises a PCI bus.

**28**. A bridge according to claim **6** wherein said second interface is operable in response to a transaction from said link signifying an initial read request, to fetch and pre-fetch data from a competent one of the bus-compatible devices on said second bus for transmission back over said link in order to satisfy pending and anticipated transactions.

**29**. A bridge according to claim **28** wherein said first interface is operable in response to a transaction from said link signifying an initial read request, to fetch and pre-fetch data from a competent one of the bus-compatible devices on said first bus for transmission back over said link in order to satisfy pending and anticipated transactions.

**30**. A bridge according to claim **1** wherein said first interface and said second interface are operable to permit at least one of the bus-compatible devices on said second bus to address one or more of the bus-compatible devices on said first bus using on said second bus substantially the same type of addressing as is used to access devices on said second bus.

**31**. A bridge accessible by a host processor for expanding access over a first bus to a second bus, said first bus and said second bus each being adapted to separately connect to respective ones of a plurality of bus-compatible devices, allowable ones of said devices including memory devices and input/output devices, said bridge comprising:

a link;

a first interface adapted to couple between said first bus and said link; and

a second interface adapted to couple between said second bus and said link, said first interface and said second

interface operating as a single bridge and being operable to (a) send outgoing, information serially through said link in a format different from that of said first bus and said second bus, (b) exchange information between said first bus and said second bus according to a predetermined hierarchy giving said first bus a higher level than said second bus, and (c) allow said host processor, communicating through said first bus, to individually address different selectable ones of the bus-compatible devices on said second bus, including memory devices and input/output devices that may be present: (i) using on said first bus substantially the same type of addressing as is used to access devices on said first bus, (ii) without first employing a second, intervening one of the bus-compatible devices on said second bus, and (iii) without passing the information through an intervening hierarchical level.

**32**. A bridge according to claim **31** wherein said first bus and said second bus each have a plurality of signaling lines for enabling bus-compatible devices to negotiate bus communications, said first interface being operable in response to a pending transaction on said first bus to begin processing said pending transaction and to apply a retry signal to at least one of said signaling lines of said first bus before the pending transaction on said first bus has been transmitted to and acknowledged by said second bus.

**33**. A bridge according to claim **32** wherein less than all of the information on the signaling lines of said first bus is transmitted by said first interface over said link.

**34**. A bridge according to claim **32** wherein said first interface comprises:

a first pair of first in first out registers for stacking information for successive groups of information gathered from and destined for said first bus.

**35**. A bridge according to claim **34** wherein said second interface comprises:

a second pair of first in first out registers for stacking information for successive groups of information gathered from and destined for said second bus.

**36**. A bridge according to claim **31** wherein said first interface is selectively responsive to those addresses appearing on said first bus that are on a predetermined schedule of addresses corresponding to the bus-compatible devices accessible through said second bus, in order to avoid responding to addresses corresponding to other ones of the bus-compatible devices on said first bus.

**37**. A bridge according to claim **36** comprising:

a register for storing said predetermined schedule.

**38**. A bridge according to claim **36** wherein said first interface comprises:

a first register for storing said predetermined schedule, said second interface comprising:

a second register for storing said predetermined schedule.

**39**. A bridge according to claim **37** wherein said register is operable to establish with respect to said first bus a base address for one or more of the bus-compatible devices on said second bus.

**40**. A bridge according to claim **31** comprising:

a register for establishing with respect to said first bus a base address for one or more of the bus-compatible devices on said second bus.

**41**. A bridge according to claim **31** wherein said first interface and said second interface are operable to permit communication between bus-compatible devices on said second bus without routing through said first bus.

**42**. A bridge according to claim **41** comprising:

an arbiter with authority to grant the second bus and without authority to grant the first bus to either said second interface or one of the bus-compatible devices on said second bus.

**43**. A bridge according to claim **31** wherein said first interface and said second interface comprise:

a first and a second programmable logic device connected between said link and said first bus and said second bus, respectively.

**44**. A bridge according to claim **31** wherein said first interface and said second interface comprise:

a first and a second application-specific integrated circuit connected between said link and said first bus and said second bus, respectively.

**45**. A bridge according to claim **43** wherein said first and said second application-specific integrated circuit are identically structured and each have a control pin for receiving a control signal to establish operation in one of two modes.

**46**. A bridge according to claim **45** wherein said first and said second application-specific integrated circuit each comprise:

an arbiter, enabled only in said second application-specific integrated circuit, with authority to grant the second bus and without authority to grant the first bus to either said second interface or one of the bus-compatible devices on said second bus.

**47**. A bridge according to claim **43** wherein said first and said second application-specific integrated circuit each comprise:

a plurality of port coupled to said second interface for providing a plurality of ports for input/output.

**48**. A bridge according to claim **31** wherein said host processor is interrupt-driven, said second interface being operable to transmit through said link to said first interface interrupt signals destined to interrupt the host processor.

**49**. A bridge according to claim **48** wherein said host processor is responsive to error signals, said second interface being operable to transmit through said link to said first interface error signals destined to affect the host processor.

**50**. A bridge according to claim **31** wherein said first bus operates at a predetermined clock speed, said link being operable to propagate data between said first interface and said second interface at a bit transfer rate greater than said predetermined clock speed.

**51**. A bridge according to claim **50** wherein said link comprises:

a pair of simplex links for sending information in opposite directions.

**52**. A bridge according to claim **51** wherein said simplex links are driven for differential signal transfers.

**53**. A bridge according to claim **51** wherein each of said simplex links comprises a twisted pair.

**54**. A bridge according to claim **51** wherein each of said simplex links comprises a cable with twin axial lines.

**55**. A bridge according to claim **54** comprising:

a single shield around both of said simplex links; and

a separate pair of individual shields around each of said simplex links.

**56**. A bridge according to claim **31** wherein said second bus comprises a PCI bus.

**57**. A bridge according to claim **35** wherein said second interface is operable in response to a transaction from said link signifying an initial read request, to fetch and pre-fetch data from a competent one of the bus-compatible devices on said second bus for transmission back over said link in order to satisfy pending and anticipated transactions.

58. A bridge according to claim 57 wherein said first interface is operable in response to a transaction from said link signifying an initial read request, to fetch and pre-fetch data from a competent one of the bus-compatible devices on said first bus for transmission back over said link in order to satisfy pending and anticipated transactions.

59. A bridge according to claim 31 wherein said first interface and said second interface are operable to permit at least one of the bus-compatible devices on said second bus to address one or more of the bus-compatible devices on said first bus using on said second bus substantially the same type of addressing as is used to access devices on said second bus.

60. A bridge accessible by a processor for expanding access over a first bus to a second bus, said first bus and said second bus each being adapted to separately connect to respective ones of a plurality of bus-compatible devices, said bridge comprising:

a link;

a first interface coupled between said first bus and said link; and

a second interface adapted to couple between said second bus and said link, said first interface and said second interface operating as a single bridge and being operable to transfer information serially through said link in a format different from that of said first bus and said second bus without waiting for an incoming acknowledgment over said link before inaugurating a transfer of said information over said link.

61. A bridge according to claim 60 wherein said first interface and said second interface are operable to exchange information between said first bus and said second bus according to a predetermined hierarchy giving said first bus a higher level than said second bus.

62. A bridge according to claim 60 wherein said first bus and said second bus each have a plurality of signaling lines for enabling bus-compatible devices to negotiate bus communications, said first interface being operable in response to a pending transaction on said first bus to begin processing said pending transaction and to apply a retry signal to at least one of said signaling lines of said first bus before the pending transaction on said first bus has been transmitted to and acknowledged by said second bus.

63. A bridge according to claim 62 wherein less than all of the information on the signaling lines of said first bus is transmitted by said first interface over said link.

64. A bridge according to claim 60 wherein said first interface is selectively responsive to those addresses appearing on said first bus that are on a predetermined schedule of addresses corresponding to the bus-compatible devices accessible through said second bus, in order to avoid responding to addresses corresponding to other ones of the bus-compatible devices on said first bus.

65. A bridge according to claim 64 comprising:

a register for storing said predetermined schedule.

66. A bridge according to claim 64 wherein said first interface comprises:

a first register for storing said predetermined schedule, said second interface comprising:

a second register for storing said predetermined schedule.

67. A bridge according to claim 65 wherein said register is operable to establish with respect to said first bus a base address for one or more of the bus-compatible devices on said second bus.

68. A bridge according to claim 60 comprising:

a register for establishing with respect to said first bus a base address for one or more of the bus-compatible devices on said second bus.

69. A bridge according to claim 60 wherein said first interface and said secondary interface are operable to permit communication between bus-compatible devices on said second bus without routing through said first bus.

70. A bridge according to claim 60 wherein said first interface and said second interface comprise:

a first and a second programmable logic device connected between said link and said first bus and said second bus, respectively.

71. A bridge according to claim 60 wherein said first interface and said second interface comprise:

a first and a second application-specific integrated circuit connected between said link and said first bus and said second bus, respectively.

72. A bridge according to claim 71 wherein said first and said second application-specific integrated circuit are identically structured and each have a control pin for receiving a control signal to establish operation in one of two modes.

73. A bridge according to claim 71 wherein said first and said second application-specific integrated circuit each comprise:

a plurality of ports coupled to said second interface for providing input/output.

74. A bridge according to claim 60 wherein said processor is interrupt-driven, said second interface being operable to transmit through said link to said first interface interrupt signals destined to interrupt the processor.

75. A bridge according to claim 74 wherein said processor is responsive to error signals, said second interface being operable to transmit through said link to said first interface error signals destined to affect the processor.

76. A bridge according to claim 60 wherein said first bus operates at a predetermined clock speed, said link being operable to propagate data between said first interface and said second interface at a bit transfer rate greater than said predetermined clock speed.

77. A bridge according to claim 76 wherein said link comprises:

a pair of simplex links for sending information in opposite directions.

78. A bridge according to claim 77 wherein said simplex links are driven for differential signal transfers.

79. A bridge according to claim 60 wherein said second bus comprises a PCI bus.

80. A bridge according to claim 60 wherein said second interface is operable in response to a transaction from said link signifying an initial read request, to fetch and pre-fetch data from a competent one of the bus-compatible devices on said second bus for transmission back over said link in order to satisfy pending and anticipated transactions.

81. A bridge according to claim 60 wherein said first interface and said second interface are operable to permit at least one of the bus-compatible devices on said second bus to address one or more of the bus-compatible devices on said first bus using on said second bus substantially the same type of addressing as is used to access devices on said second bus.

* * * * *