US 20080101147A1

(19) United States

(12) Patent Application Publication

(10) Pub. No.: US 2008/0101147 A1

(43) Pub. Date: May 1, 2008

Amidi

(54) CLOCK AND POWER FAULT DETECTION FOR MEMORY MODULES

(76) Inventor: **Hossein Amidi**, Lake Forest, CA (US)

Correspondence Address:
PERKINS COIE LLP
P.O. BOX 2168
MENLO PARK, CA 94026

(21) Appl. No.: 11/552,949

(22) Filed: Oct. 25, 2006

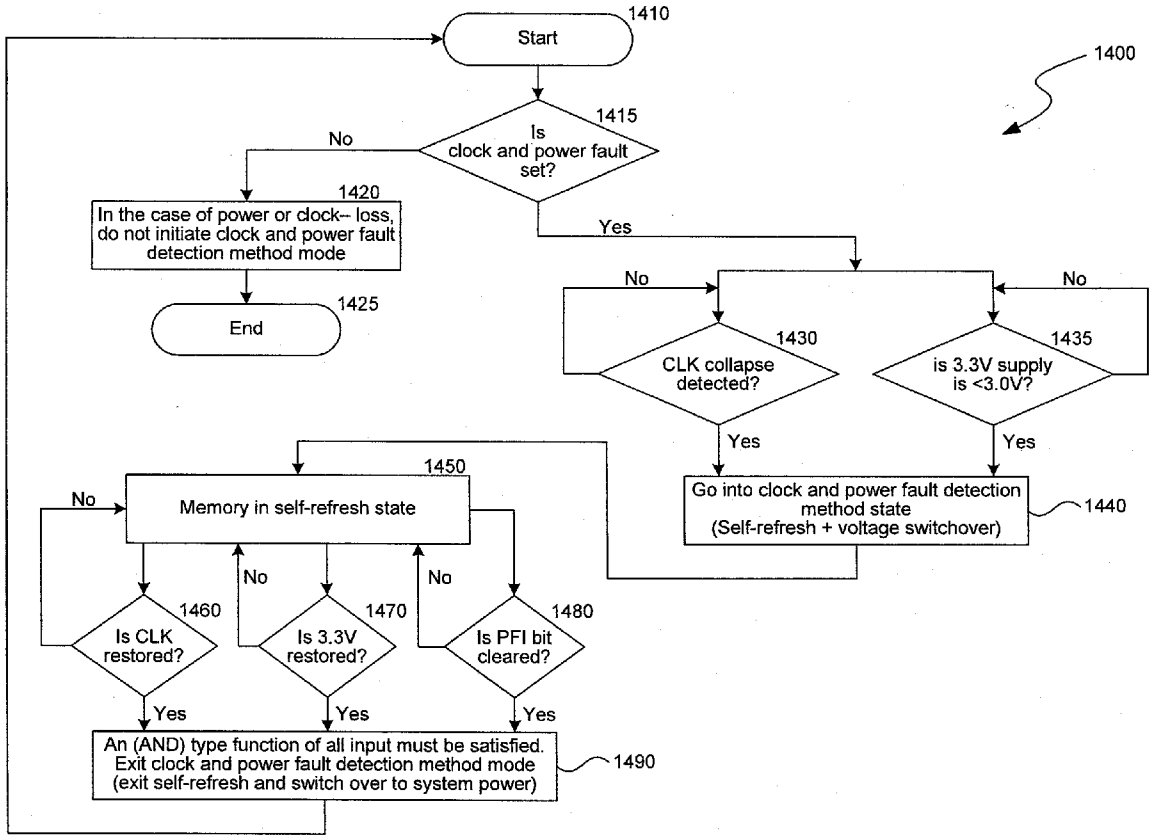
Publication Classification

(51) Int. Cl.
*G11C 5/14* (2006.01)

(52) U.S. Cl. ..... 365/229

(57) ABSTRACT

A system, method and apparatus for clock and power fault detection for a memory module is provided. In one embodiment, a system is provided. The system includes a voltage detection circuit and a clock detection circuit. The system further includes a controller coupled to the voltage detection circuit and the clock detection circuit. The system also includes a memory control state machine coupled to the controller. The system includes volatile memory coupled to the memory control state machine. The system further includes a battery and battery regulation circuitry coupled to the controller and the memory control state machine. The battery, battery regulation circuitry, volatile memory, memory control state machine, controller, clock detection circuit and voltage detection circuit are all collectively included in a unitary memory module.



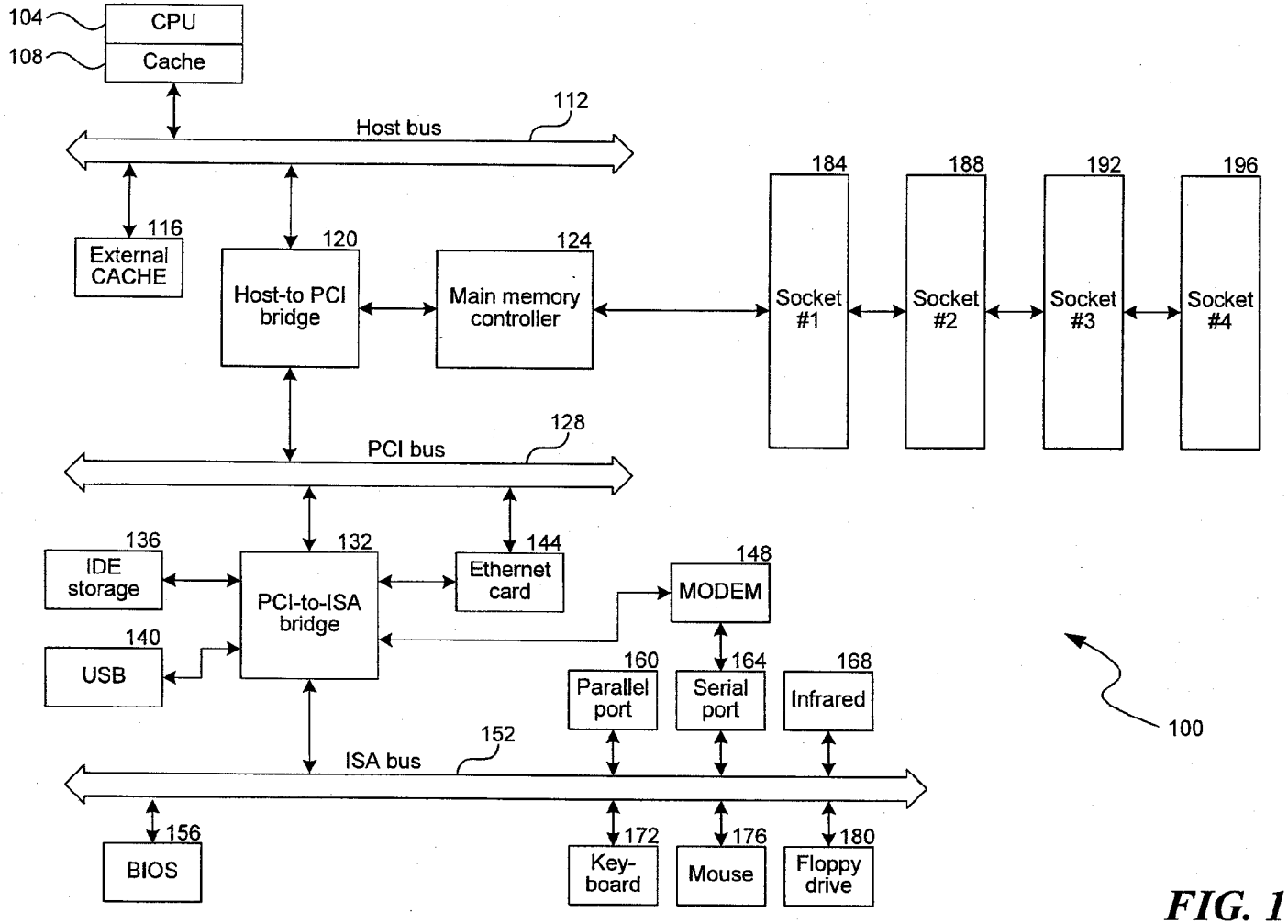

104
CPU
108
Cache
Host bus
112
116
External CACHE
120
Host-to PCI bridge
124
Main memory controller
184
Socket #1
188
Socket #2
192
Socket #3
196
Socket #4
PCI bus
128
136
IDE storage
132
PCI-to-ISA bridge
144
Ethernet card
148
MODEM
140
USB
160
Parallel port
164
Serial port
168
Infrared
100
ISA bus
152
156
BIOS
172
Key-board
176
Mouse
180
Floppy drive

*FIG. 1*

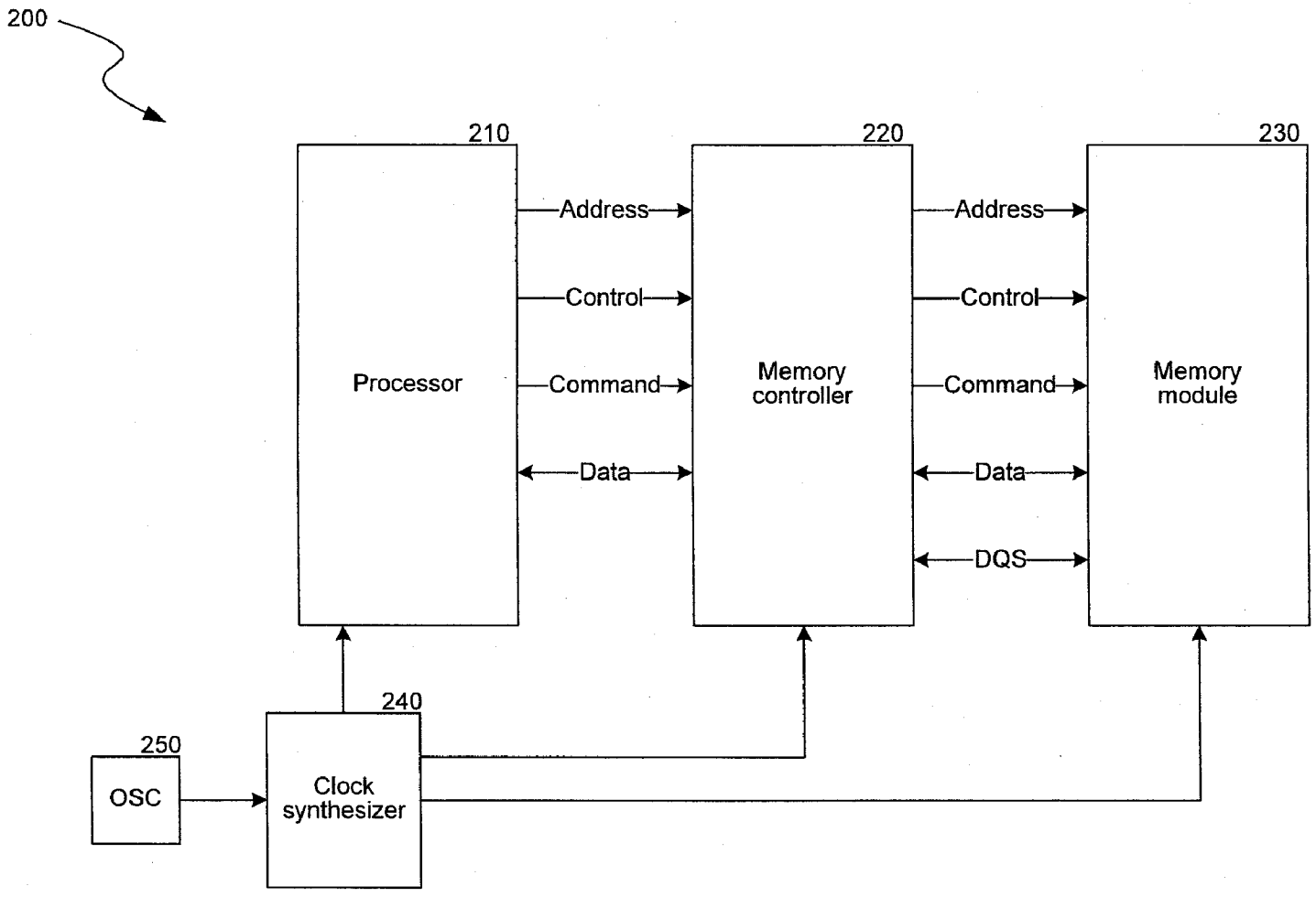

200
210
220
230
Processor
Memory controller
Memory module
Address
Control
Command
Data
Address
Control
Command
Data
DQS
240
250
OSC
Clock synthesizer

*FIG. 2*

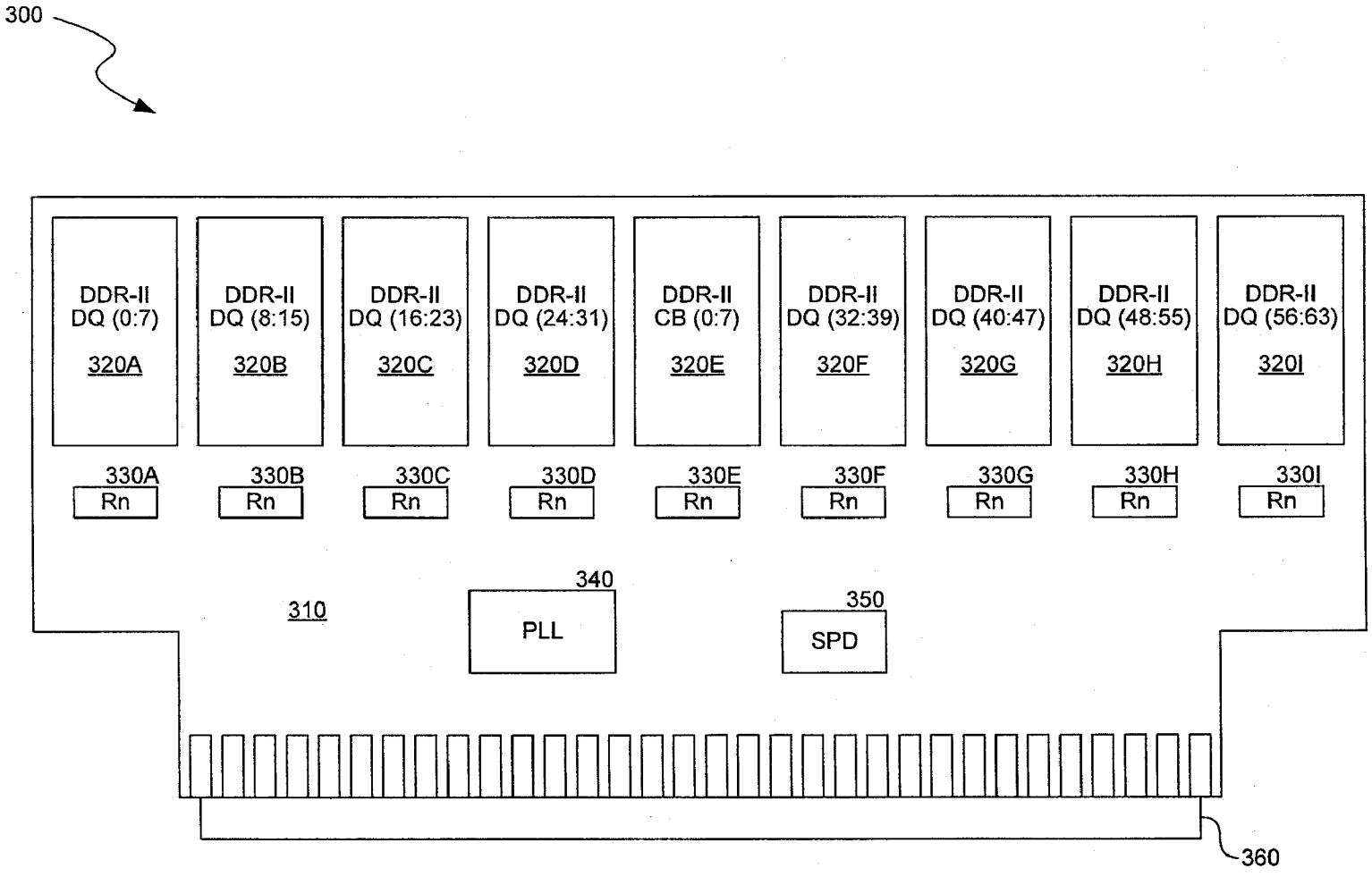

300
DDR-II DQ (0:7) 320A
DDR-II DQ (8:15) 320B
DDR-II DQ (16:23) 320C
DDR-II DQ (24:31) 320D
DDR-II CB (0:7) 320E
DDR-II DQ (32:39) 320F
DDR-II DQ (40:47) 320G
DDR-II DQ (48:55) 320H
DDR-II DQ (56:63) 320I
330A Rn
330B Rn
330C Rn
330D Rn
330E Rn
330F Rn
330G Rn
330H Rn
330I Rn
310
340 PLL
350 SPD
360

*FIG. 3*

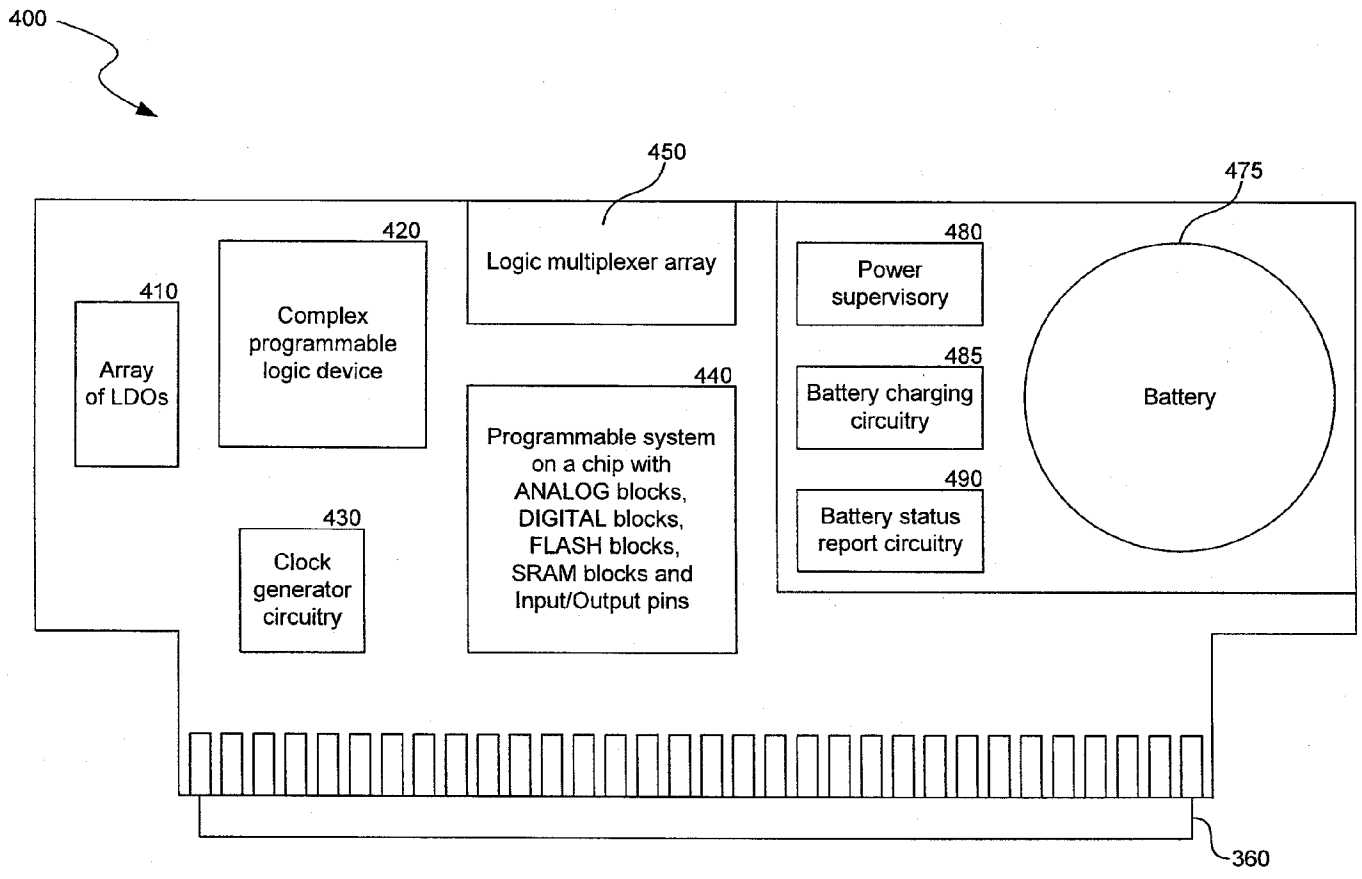

400
450
475
420
Logic multiplexer array
480
Power supervisory
410
Complex programmable logic device
485
Battery charging circuitry
440
Battery
Array of LDOs
Programmable system on a chip with ANALOG blocks, DIGITAL blocks, FLASH blocks, SRAM blocks and Input/Output pins
490
Battery status report circuitry
430
Clock generator circuitry
360

*FIG. 4*

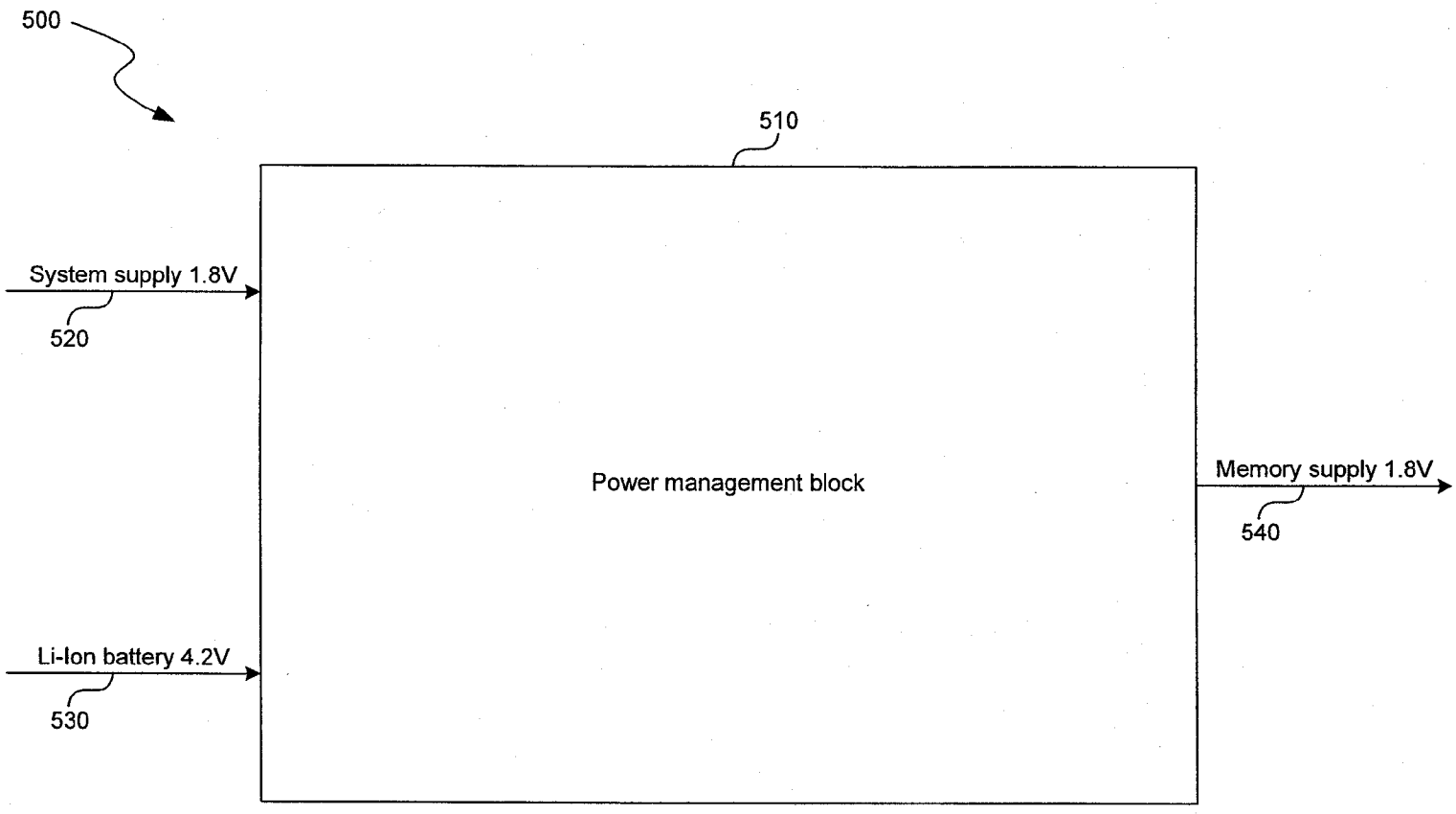

500
510
System supply 1.8V
520
Power management block
Memory supply 1.8V
540
Li-Ion battery 4.2V
530

*FIG. 5*

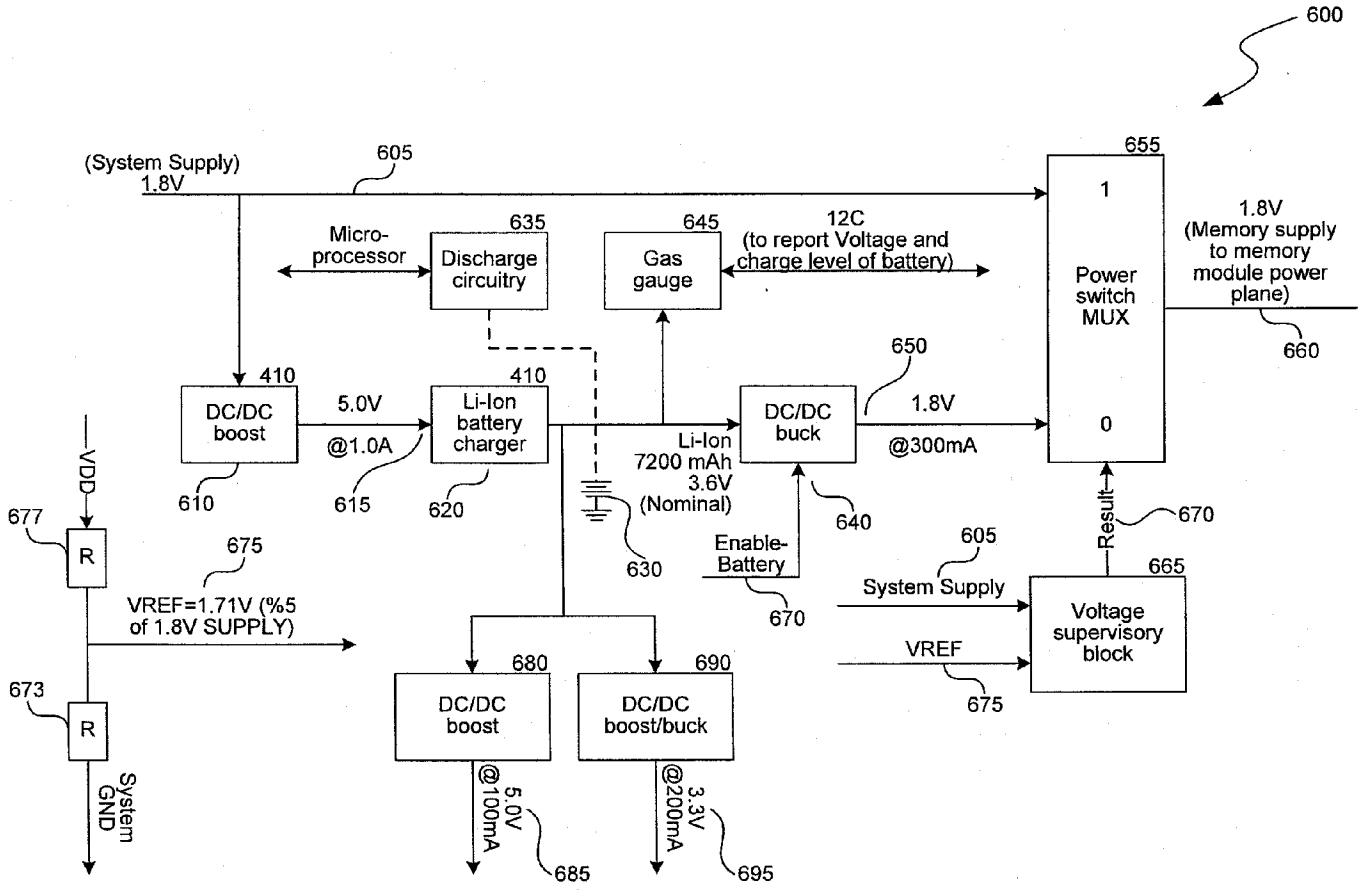

600
(System Supply)
1.8V
605
Micro-
processor
635
Discharge
circuitry
645
Gas
gauge
12C
(to report Voltage and
charge level of battery)
655
1
Power
switch
MUX
0
1.8V
(Memory supply
to memory
module power
plane)
660
410
DC/DC
boost
610
5.0V
@1.0A
615
410
Li-Ion
battery
charger
620
Li-Ion
7200 mAh
3.6V
(Nominal)
630
650
DC/DC
buck
640
1.8V
@300mA
Enable-
Battery
670
Result
670
605
System Supply
665
Voltage
supervisory
block
VREF
675
VDD
677
R
675
VREF=1.71V (%5
of 1.8V SUPPLY)
673
R
System
GND
680
DC/DC
boost
5.0V
@100mA
685
690
DC/DC
boost/buck
3.3V
@200mA
695

*FIG. 6*

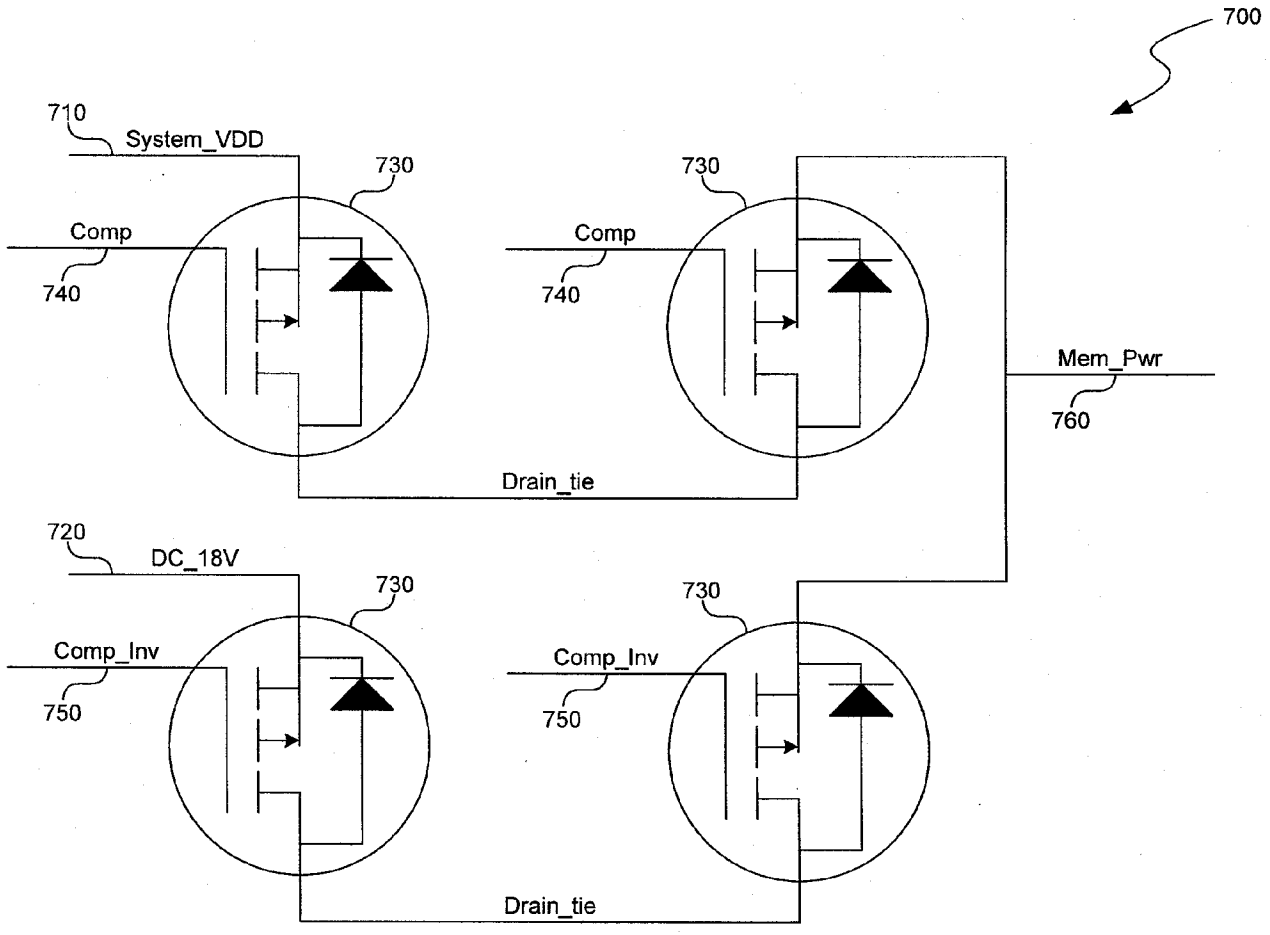

700
710
System_VDD
730
Comp
740
730
Comp
740
Mem_Pwr
760
Drain_tie
720
DC_18V
730
Comp_Inv
750
730
Comp_Inv
750
Drain_tie

*FIG. 7*

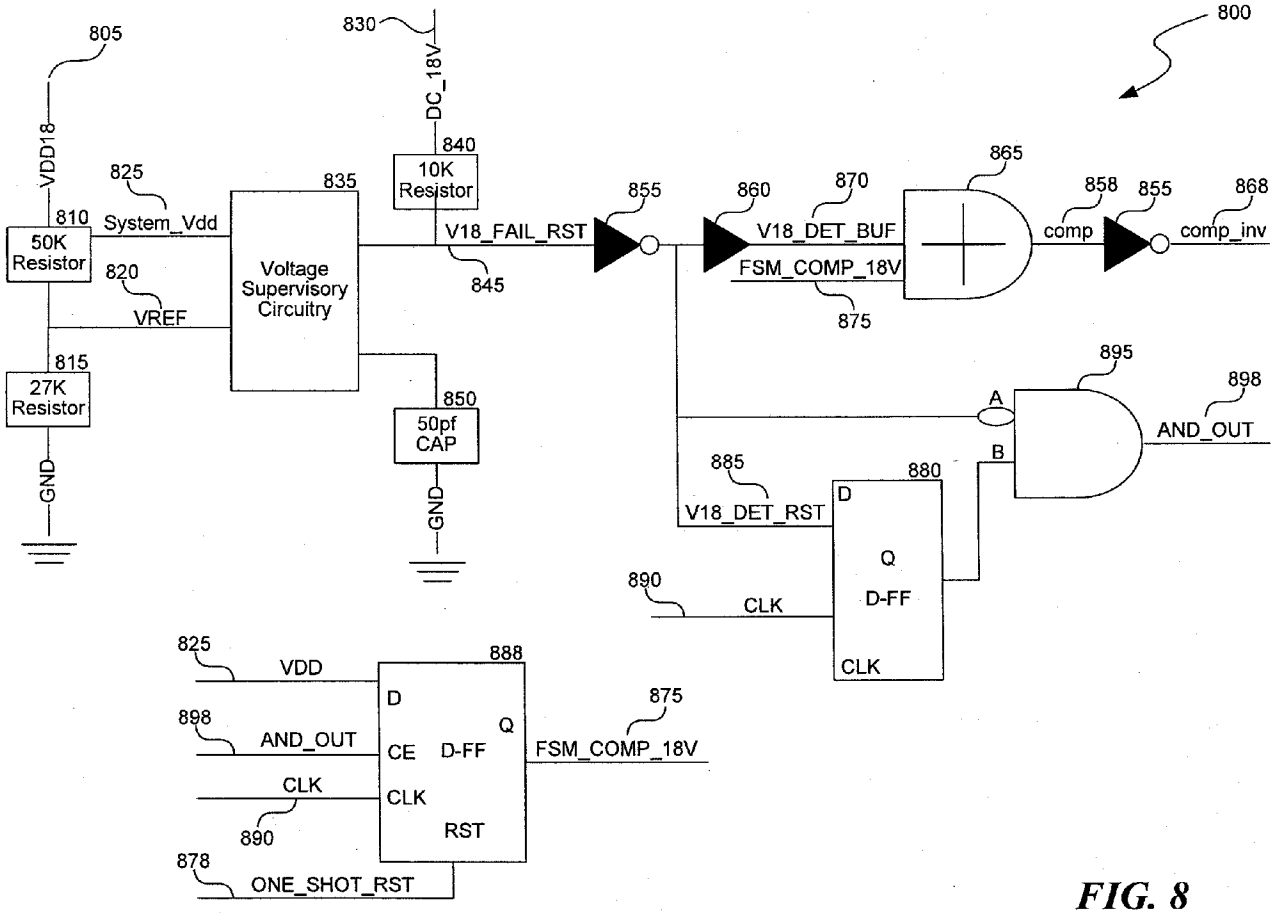

800
805
VDD18
810
50K Resistor
825
System_Vdd
820
VREF
815
27K Resistor
GND
830
DC_18V
840
10K Resistor
835
Voltage Supervisory Circuitry
845
V18_FAIL_RST
850
50pf CAP
GND
855
860
870
V18_DET_BUF
FSM_COMP_18V
875
865
858
comp
855
868
comp_inv
895
A
B
898
AND_OUT
885
V18_DET_RST
880
D
Q
D-FF
CLK
890
CLK
825
VDD
888
D
Q
898
AND_OUT
CE
D-FF
875
FSM_COMP_18V
CLK
CLK
890
RST
878
ONE_SHOT_RST

*FIG. 8*

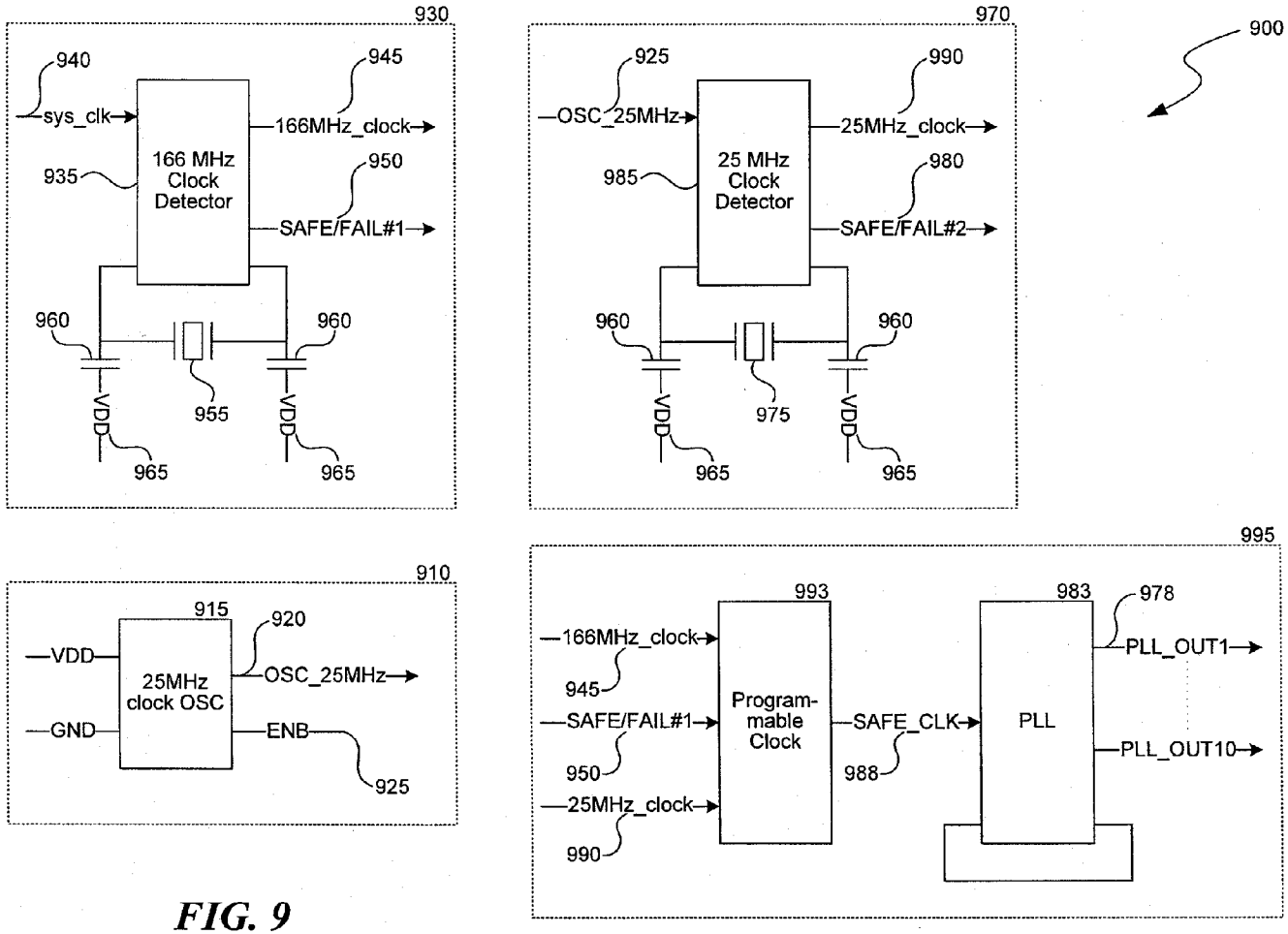

900
930
940
sys_clk
945
166MHz_clock
935
166 MHz Clock Detector
950
SAFE/FAIL#1
960
960
955
VDD
VDD
965
965
970
925
OSC_25MHz
990
25MHz_clock
985
25 MHz Clock Detector
980
SAFE/FAIL#2
960
960
975
VDD
VDD
965
965
910
915
920
VDD
25MHz clock OSC
OSC_25MHz
GND
ENB
925
995
993
983
978
166MHz_clock
945
Program-mable Clock
PLL
PLL_OUT1
SAFE/FAIL#1
950
SAFE_CLK
988
PLL_OUT10
25MHz_clock
990

*FIG. 9*

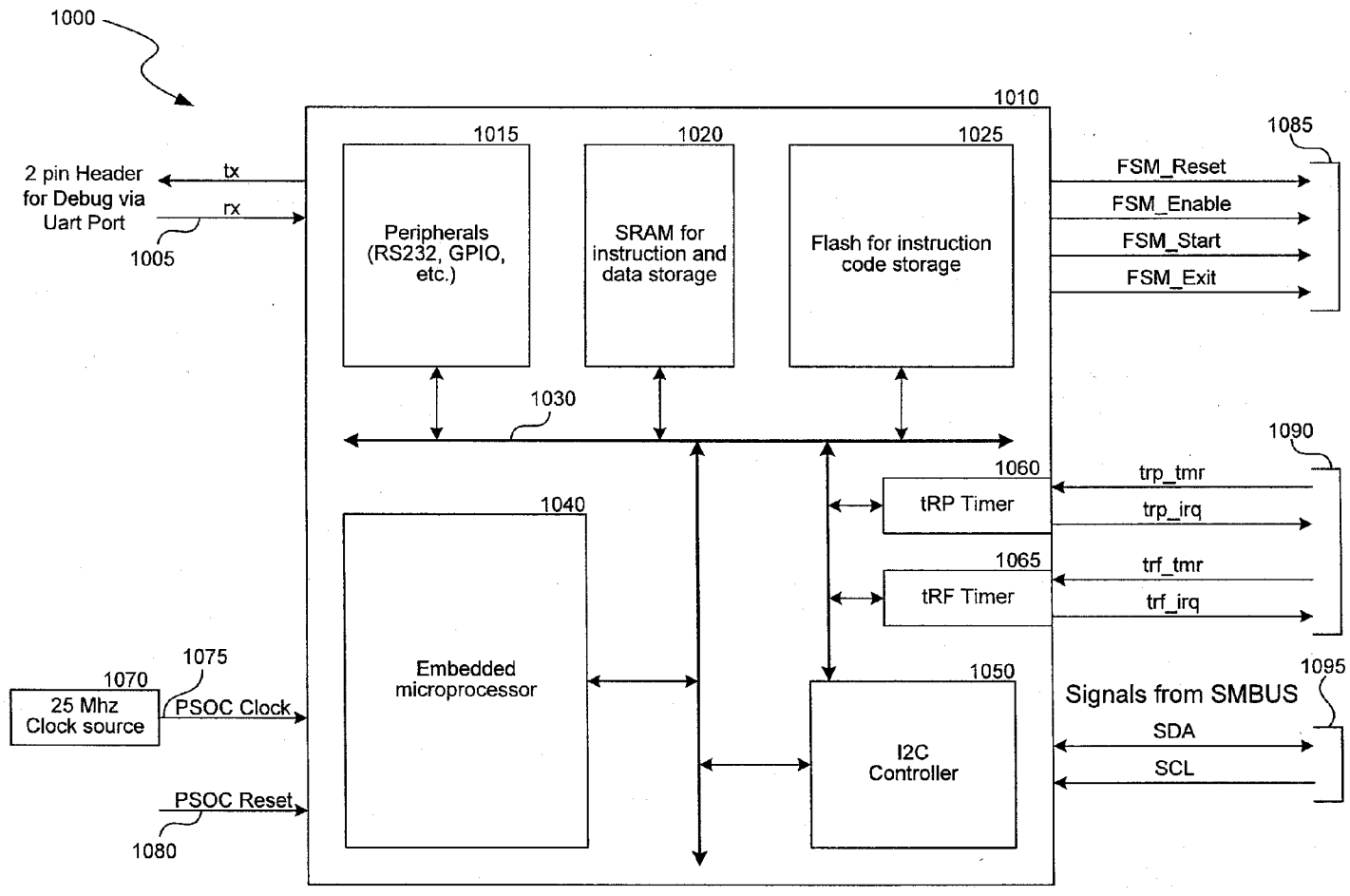

1000
1010
1015
1020
1025
2 pin Header for Debug via Uart Port
tx
rx
1005
Peripherals (RS232, GPIO, etc.)
SRAM for instruction and data storage
Flash for instruction code storage
1085
FSM_Reset
FSM_Enable
FSM_Start
FSM_Exit
1030
1090
1060
tRP Timer
trp_tmr
trp_irq
1040
1065
tRF Timer
trf_tmr
trf_irq
1075
1070
25 Mhz Clock source
PSOC Clock
Embedded microprocessor
1050
1095
Signals from SMBUS
SDA
SCL
I2C Controller
PSOC Reset
1080

*FIG. 10*

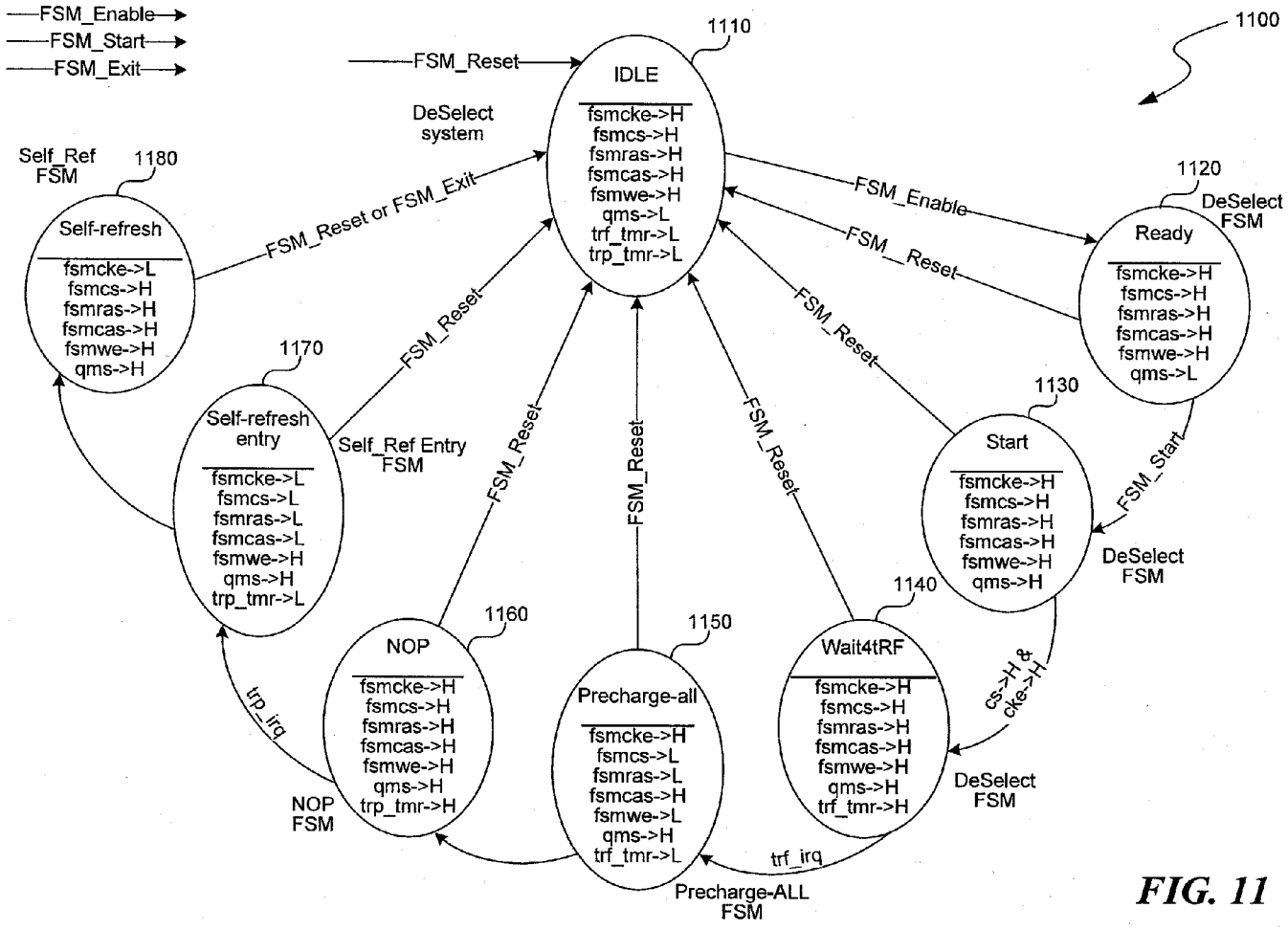

FSM_Enable
FSM_Start
FSM_Exit
FSM_Reset
1100
1110
IDLE
fsmcke->H
fsmcs->H
fsmras->H
fsmcas->H
fsmwe->H
qms->L
trf_tmr->L
trp_tmr->L
DeSelect system
Self_Ref FSM
1180
Self-refresh
fsmcke->L
fsmcs->H
fsmras->H
fsmcas->H
fsmwe->H
qms->H
FSM_Reset or FSM_Exit
FSM_Enable
1120
DeSelect FSM
Ready
fsmcke->H
fsmcs->H
fsmras->H
fsmcas->H
fsmwe->H
qms->L
FSM_Reset
1170
Self-refresh entry
fsmcke->L
fsmcs->L
fsmras->L
fsmcas->L
fsmwe->H
qms->H
trp_tmr->L
Self_Ref Entry FSM
FSM_Reset
1130
Start
fsmcke->H
fsmcs->H
fsmras->H
fsmcas->H
fsmwe->H
qms->H
FSM_Start
DeSelect FSM
1160
NOP
fsmcke->H
fsmcs->H
fsmras->H
fsmcas->H
fsmwe->H
qms->H
trp_tmr->H
trp_irq
NOP FSM
1150
Precharge-all
fsmcke->H
fsmcs->L
fsmras->L
fsmcas->H
fsmwe->L
qms->H
trf_tmr->L
Precharge-ALL FSM
1140
Wait4tRF
fsmcke->H
fsmcs->H
fsmras->H
fsmcas->H
fsmwe->H
qms->H
trf_tmr->H
trf_irq
cs->H & cke->H
DeSelect FSM

*FIG. 11*

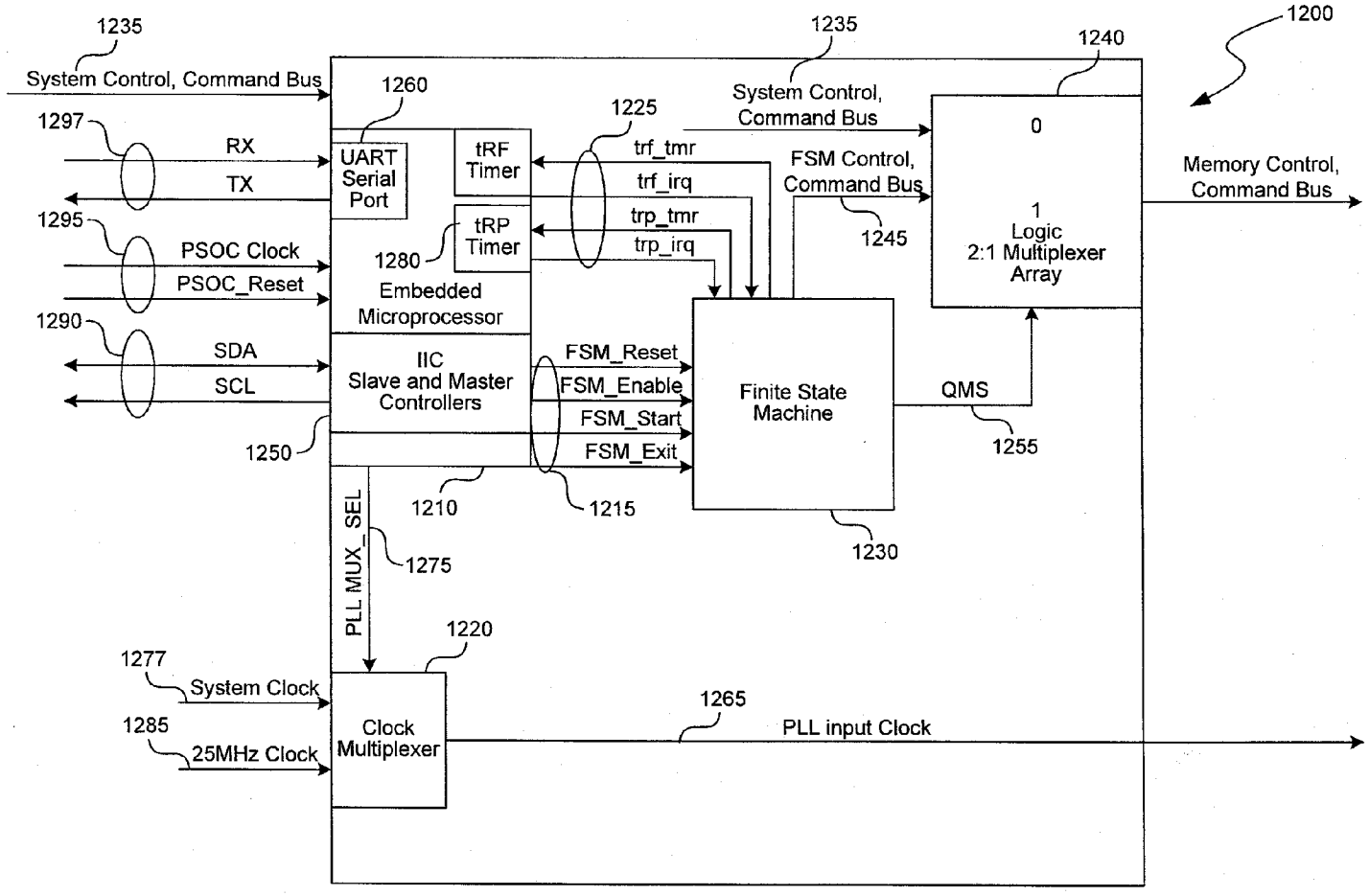

1200
1235
System Control, Command Bus
1297
RX
TX
1295
PSOC Clock
PSOC_Reset
1290
SDA
SCL
1250
1260
UART Serial Port
tRF Timer
tRP Timer
1280
Embedded Microprocessor
IIC Slave and Master Controllers
1210
1215
1225
trf_tmr
trf_irq
trp_tmr
trp_irq
FSM_Reset
FSM_Enable
FSM_Start
FSM_Exit
1235
System Control, Command Bus
FSM Control, Command Bus
1245
Finite State Machine
1230
QMS
1255
1240
0
1
Logic 2:1 Multiplexer Array
Memory Control, Command Bus
PLL MUX_SEL
1275
1220
1277
System Clock
1285
25MHz Clock
Clock Multiplexer
1265
PLL input Clock

*FIG. 12*

1300

| Function | CKE Previous cycle | CKE Current cycle | CS | RAS | CAS | WE | BA0 BA1 | A[13:11] | A10 | A[9:0] | Note |
|---|---|---|---|---|---|---|---|---|---|---|---|
| (Extended) mode register set | H | H | L | L | L | L | BA | OP code | | | |
| Auto-refresh | H | H | L | L | L | H | X | X | X | X | |
| Self-refresh entry | H | L | L | L | L | H | X | X | X | X | |
| Self-Refresh Exit | L | H | H | X | X | X | X | X | X | X | |
| | | | L | H | H | H | | | | | |
| Single bank precharge | H | H | L | L | H | L | BA | X | L | X | |
| Precharge all banks | H | H | L | L | H | L | X | X | H | X | |
| Bank activate | H | H | L | L | H | H | BA | Row Address | | | |
| Write | H | H | L | H | L | L | BA | Column | L | Column | |
| Write with auto-precharge | H | H | L | H | L | L | BA | Column | H | Column | |
| Read | H | H | L | H | L | H | BA | Column | L | Column | |
| Read with auto-precharge | H | H | L | H | L | H | BA | Column | H | Column | |
| No operation | H | X | L | H | H | H | X | X | X | X | |
| Device deselect | H | X | H | X | X | X | X | X | X | X | |
| Power down entry | H | L | H | X | X | X | X | X | X | X | |
| | | | L | H | H | H | X | X | X | X | |
| Power down exit | L | H | H | X | X | X | | | | | |
| | | | L | H | H | H | X | X | X | X | |

1) The state of ODT does not affect the states described in this table. The ODT function is not available during Self Refresh.
2) "X" means "H or L (but a defined logic level)".
3) Operation that is not specified is illegal and after such an event, in order to guarantee proper operation, the DRAM must be powered down and then restarted through the specified initialization sequence before normal operation can continue.
4) All DDR2 SDRAM commands are defined by states of CS, WE, RAS, CAS, and CKE at the rising edge of the clock.
5) Bank addresses BA[1:0] determine which bank is to be operated upon. For (E)MRS BA[1:0] selects an (Extended) Mode Register.
6) $V_{REF}$ must be maintained during Self Refresh operation.
7) Self Refresh Exit is asynchronous.
8) Burst reads or writes at BL=4 cannot be terminated.
9) The Power Down Mode does not perform any refresh operations.

*FIG. 13*

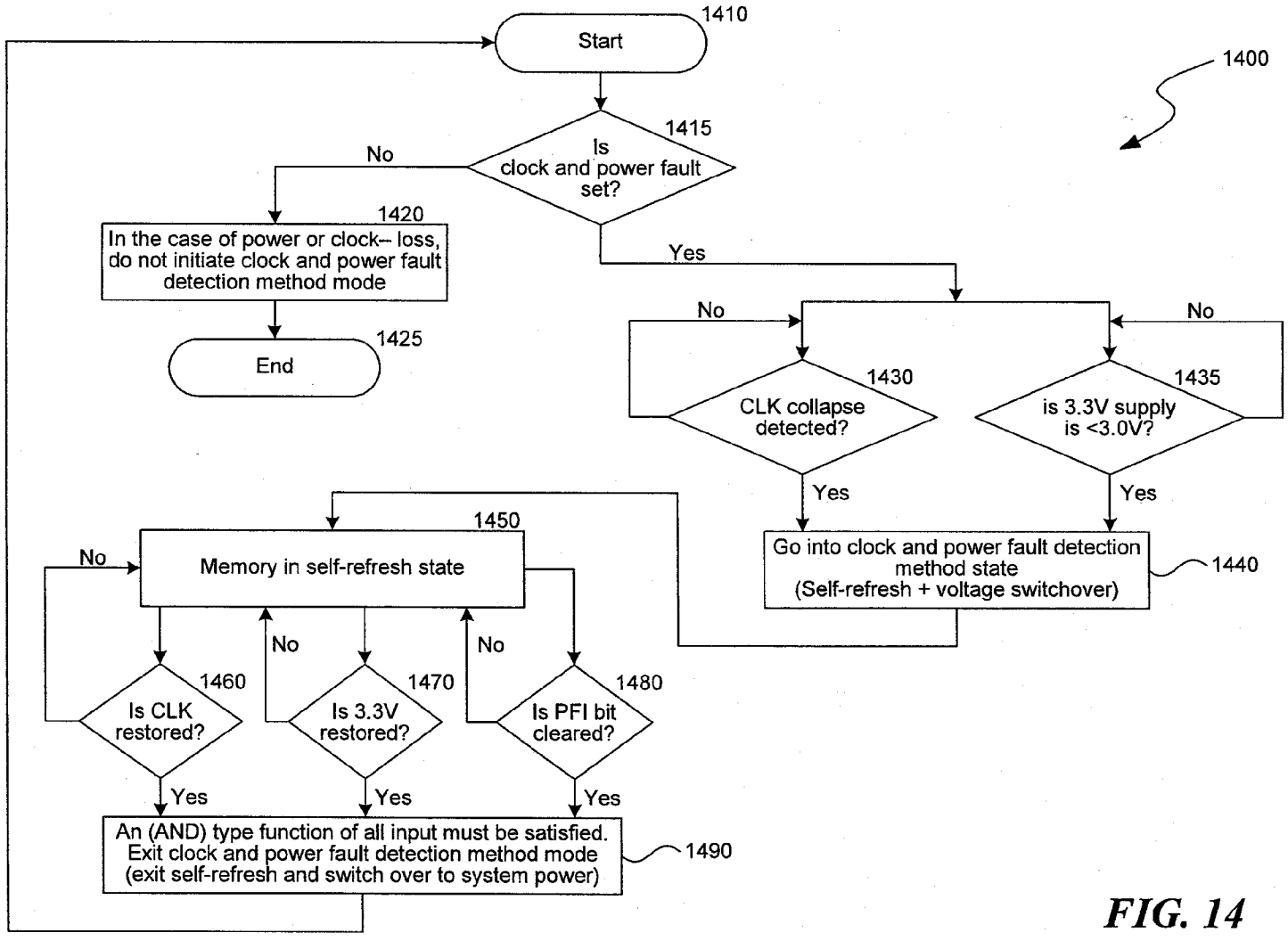

1400
1410
Start
1415
Is clock and power fault set?
No
Yes
1420
In the case of power or clock– loss, do not initiate clock and power fault detection method mode
1425
End
No
No
1430
CLK collapse detected?
1435
is 3.3V supply is <3.0V?
Yes
Yes
1440
Go into clock and power fault detection method state (Self-refresh + voltage switchover)
1450
Memory in self-refresh state
No
No
No
1460
Is CLK restored?
1470
Is 3.3V restored?
1480
Is PFI bit cleared?
Yes
Yes
Yes
1490
An (AND) type function of all input must be satisfied. Exit clock and power fault detection method mode (exit self-refresh and switch over to system power)

*FIG. 14*

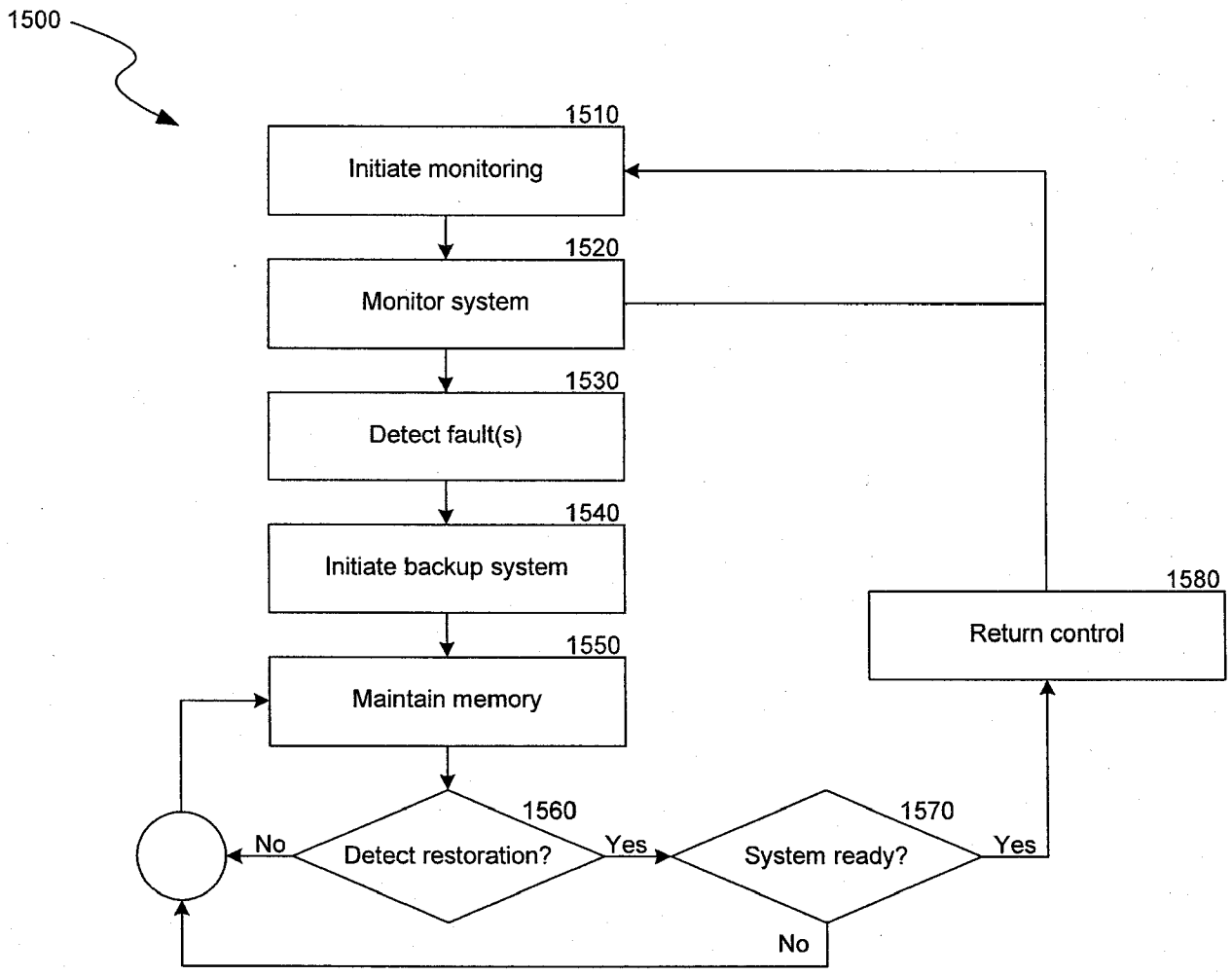

1500
1510
Initiate monitoring
1520
Monitor system
1530
Detect fault(s)
1540
Initiate backup system
1550
Maintain memory
1560
Detect restoration?
No
Yes
1570
System ready?
Yes
No
1580
Return control

*FIG. 15*

# CLOCK AND POWER FAULT DETECTION FOR MEMORY MODULES

## BACKGROUND

**[0001]** Computer systems operate in part using volatile memory. Memory modules with Random Access Memory typically will not retain any data when power is not supplied. Such memory modules require power to maintain values which are stored in memory cells, and may also require periodic refresh of contents of memory cells. This differs from non-volatile memory such as various forms of Read Only Memory or other memory such as magnetic or optical memory. However, whereas non-volatile memory tends to have long-term storage capacity, it also tends to be slower, with read-only memory of various types often copied into volatile random access memory during operation of computers and similar machines.

**[0002]** As systems become more mission critical, the possibility of irreplaceable data being stored in volatile memory increases. Similarly, failure analysis can be much simpler if information about the state of a system is available after a failure occurs. Moreover, some data may be useful for restarting a system after a failure, even though that information is not otherwise vital for external purposes. Also, some data may simply be desirable for retention purposes, but may also be most useful in volatile memory.

**[0003]** Thus, it may be useful to provide an option for keeping data in volatile memory even when a surrounding system loses power. Moreover, it may also be useful to keep data in volatile memory when a surrounding system suffers some form of an error which causes a clock to malfunction even though power is still supplied. Likewise, it may be useful to provide volatile memory which has non-volatile characteristics in short- or medium-term time periods.

## BRIEF DESCRIPTION OF THE DRAWINGS

**[0004]** The present invention is illustrated by way of example in the accompanying drawings. The drawings should be understood as illustrative rather than limiting.

**[0005]** FIG. **1** illustrates an embodiment of a computer.

**[0006]** FIG. **2** illustrates an embodiment of a memory interface in a computer.

**[0007]** FIG. **3** illustrates an embodiment of a top side of an unbuffered clocked memory module.

**[0008]** FIG. **4** illustrates an embodiment of a back or bottom side of an unbuffered clocked memory module.

**[0009]** FIG. **5** illustrates an embodiment of a power management block.

**[0010]** FIG. **6** illustrates an embodiment of the internals of a power management block.

**[0011]** FIG. **7** illustrates an embodiment of a power switch multiplexer.

**[0012]** FIG. **8** illustrates an embodiment of a voltage supervisory system.

**[0013]** FIG. **9** illustrates an embodiment of clock circuitry.

**[0014]** FIG. **10** illustrates an embodiment of a processor system on a chip.

**[0015]** FIG. **11** illustrates an embodiment of a state machine.

**[0016]** FIG. **12** illustrates an embodiment of a memory control system.

**[0017]** FIG. **13** illustrates an embodiment of a truth table.

**[0018]** FIG. **14** illustrates an embodiment of a process of controlling power supply and clock signals to a memory system.

**[0019]** FIG. **15** illustrates an alternate embodiment of a process for monitoring clock and power for a memory module.

## DETAILED DESCRIPTION

**[0020]** A system, method and apparatus is provided for a clock and power fault detection for a memory module. The specific embodiments described in this document represent examples or embodiments of the present invention, and are illustrative in nature rather than restrictive.

**[0021]** Clock and power fault detection for a memory module may be provided in a variety of ways. For example, one may provide a system with a controller which detects voltage levels and clock signals, a state machine for operating a memory in backup mode, and a battery and supporting circuitry for supplying backup power. Similarly, one may provide a process which operates to detect voltage and clock signals, initiate backup operations, maintain memory (through refresh, for example), and detect a recovery status. Providing such a system or process within a memory module can be very helpful, as it avoids the need for system circuitry in a computer system or similar device which can maintain a memory module from outside the module. Moreover, such a system or process may be tuned to the specific memory module, instead of requiring overhead to deal with many different types of memory modules, for example.

**[0022]** In one embodiment, a system is provided. The system includes a voltage detection circuit and a clock detection circuit. The system further includes a controller coupled to the voltage detection circuit and the clock detection circuit. The system also includes a memory control state machine coupled to the controller. The system includes volatile memory coupled to the memory control state machine. The system further includes a battery and battery regulation circuitry coupled to the controller and the memory control state machine. The battery, battery regulation circuitry, volatile memory, memory control state machine, controller, clock detection circuit and voltage detection circuit are all collectively included in a unitary memory module.

**[0023]** In the following description, for purposes of explanation, numerous specific details are set forth in order to provide a thorough understanding of the invention. It will be apparent, however, to one skilled in the art that the invention can be practiced without these specific details. In other instances, structures and devices are shown in block diagram form in order to avoid obscuring the invention.

**[0024]** Reference in the specification to "one embodiment" or "an embodiment" means that a particular feature, structure, or characteristic described in connection with the embodiment is included in at least one embodiment of the invention. The appearances of the phrase "in one embodiment" in various places in the specification are not necessarily all referring to the same embodiment, nor are separate or alternative embodiments mutually exclusive of other embodiments. Features and aspects of various embodiments may be integrated into other embodiments, and embodiments illustrated in this document may be implemented without all of the features or aspects illustrated or described.

**[0025]** FIG. **1** illustrates an embodiment of a computer. Computer **100** is a machine which includes some of the

elements typically found in various types of computers, such as desktop or laptop computers, personal digital assistants, or server computers, for example. Not shown are some common subsystems, such as a graphics accelerator or other video subsystem, for example.

**[0026]** Computer **100** includes a processor **104** (such as a central processing unit) with a cache **108** coupled thereto. Host bus **112** provides for an interface between processor **104** and components such as external cache **116**, host-to-PCI bridge **120** and other similar components. External cache **116** may provide for additional caching resources, such as a level two cache, for example. Host-to-PCI bridge **120** may provide a bridge to a PCI bus **128**, and may also provide a datapath to a memory controller **124**, for example. Thus, bridge **120** may serve as a memory hub, for example. Memory controller **124** controls access to memory modules in sockets **184**, **188**, **192** and **196**.

**[0027]** PCI bus **128** provides an interface with still more components of the computer **100**. Coupled to PCI bus **128** are PCI-to-ISA bridge **132** and Ethernet card **144**. Bridge **132** provides a bus bridge to ISA bus **152**, and provides a datapath to components such as IDE storage subsystem **136** and USB interface **140**, along with Ethernet card **144** and modem **148**. ISA bus **152** provides a datapath to BIOS **156**, parallel port **160**, serial port **164**, infrared port **168**, keyboard **172**, mouse **176** and floppy drive **180**. As one may expect, other components may be included and many of the components illustrated (such as the floppy disk **180** for example) may be omitted from some embodiments.

**[0028]** Memory sockets such as sockets **184**, **188**, **192** and **196** may be populated with various types of memory modules. FIG. **2** illustrates an embodiment of a memory interface in a computer. The memory interface **200** provides for communication between a processor **210** and a memory module **230**. Processor **210** provides control signals and (potentially) data to memory controller **220**. Controller **220** then provides control signals to memory module **230**. Thus, controller **220** handles the sometimes complicated process of signaling module **230**, providing address, control and command signals at the right times for the memory module **230**, and either providing or receiving data and DQS signals as necessary. Similar address, control and command signals are provided by the processor **210** to the controller **220**, but may not be provided with the timing required for module **230**. In synchronous systems, oscillator **250** and clock synthesizer **240** provide clock signals to the various components.

**[0029]** With a memory system in place, a memory module must be supplied as part of the system. FIG. **3** illustrates an embodiment of a top side of an unbuffered clocked memory module. Memory in general is typically SDRAM—synchronous dynamic random access memory, which is volatile. If a memory system gets interrupted too much, the data stored in SDRAM can be lost. Thus, memory module **300** may be used as part of a system which maintains some of the data in SDRAM.

**[0030]** In one embodiment, memory module **300** includes a memory card (e.g. a printed circuit board), memory chips, series termination resistors, PLL (Phase-Lock-Loop) and SPD (Serial Presence Detect) components, and a card interface. Memory card **310** provides the base for the memory module **300**, and provides traces for conductivity between components. Mounted on memory card **310** are memory chips **320** (**320**A-I as illustrated) and resistors **330** (**330**A-I as illustrated). Also provided are a PLL **340** and an SPD module **350**. Furthermore, some of the traces or conductors of memory card **310** connect to card interface **360**, a set of printed conductors on an edge of the card which may mate with a slot or socket on a card to which memory module **300** is connected. Thus, the front side of memory card **310** basically includes the memory components.

**[0031]** The back side of memory card **310** may contain other components. FIG. **4** illustrates an embodiment of a back or bottom side of an unbuffered clocked memory module. The module is provided with clock and power fault detection capabilities. Card interface **360** is present on this side as well, providing connectivity. Also included is an array of LDOs **410**, providing power regulation, a CPLD (Complex Programmable Logic Device) **420**, providing logic such as a state machine, a battery module **470**, a logic signal multiplexer array **450**, providing switching capabilities for logic signals, a processor **440**, providing a processor system on a chip, and a clock generator module **430**, providing clock signals. Battery module **470** includes a battery **475**, power supervisory module **480**, battery charging circuitry **485** and battery status circuitry **490**. Where it seems apparent that further description is needed, embodiments of these components are illustrated and described below.

**[0032]** Part of a memory module is typically a power management block. FIG. **5** illustrates an embodiment of a power management block. Power management block **500** includes the actual power management module **510**, an incoming system supply **520**, an incoming battery supply **530**, and an outgoing memory power supply **540**. Power management block **500** may thus be used to attempt to ensure a stable power supply even in the face of disruptions in system supply **520**, for example.

**[0033]** While a power management block **500** may be implemented in a variety of ways, power management block **600** provides one example of such an implementation. FIG. **6** illustrates an embodiment of the internals of a power management block. System supply **605** provides a system power supply to a boost power converter **610** and a power switch multiplexer **655**.

**[0034]** Boost converter **610** provides a power output **615** which powers battery charger **620**—which in turn supplies battery **630**. If necessary, discharge circuitry **635** can discharge battery **630** on a command from a microprocessor. Moreover, gas gauge **645** can interpret battery output **625** to determine a rough charge status of battery **630**. Additionally, battery output **625** is provided to boost converter **680** to produce power supply **685** and boost/buck converter **690** to produce power supply **695**. Also, battery output **625** provides power to buck converter **640** (when enabled by battery enable signal **670**) which produces battery power **650** as an input to power switch multiplexer **655**. Thus, power switch multiplexer **655** may switch between battery power **650** and system supply **605** based on a signal **670**.

**[0035]** Signal **670** is controlled by voltage supervisory block **665**, which receives system supply **605** and a reference voltage **675**, and compares the two. If system supply **605** has a greater magnitude than reference voltage **675**, signal **670** causes power switch multiplexer **655** to choose system supply **605** as a source for memory supply **660**. If system supply **605** has a magnitude lower than reference voltage **675**, then signal **670** causes power switch multiplexer **655** to choose battery power **650** for memory supply

660. Reference voltage **675** is generated in one embodiment by a resistive voltage divider using resistors **677** and **673** in series between a voltage rail and ground.

**[0036]** Power switch multiplexer **655** may be implemented in a number of different ways. Generally, multiplexers are well known. However, multiplexing a power signal output may involve concerns not commonly found in other multiplexing situations. FIG. **7** illustrates an embodiment of a power switch multiplexer. Multiplexer **700** uses a set of power transistors to provide conduction paths for the two input power supply signals.

**[0037]** System supply **710** is coupled to a first transistor **730**, which in turn is coupled to a second transistor **730**, which in turn is coupled to memory power output **760**. The transistors **730** in the path from supply **710** to memory supply **760** are controlled by a comparison signal **740**. Additionally, the transistors **730** in this path are coupled at the drain side of the transistors **730**, so that the parasitic diodes formed by each transistor **730** in the path are opposed to each other—parasitic conduction for one transistor is blocked by a blocking path in the other transistor when the transistors are turned off. Thus, when the path between system supply **710** and memory supply **760** is shutoff, even parasitic conductance should be minimal or zero.

**[0038]** A similar conduction path is provided between battery supply **720** and memory supply **760**. Two transistors **730** are drain coupled in the path from battery supply **720** to memory supply **760**, and the transistors of this path are controlled by comparison inverse signal **750**. Signal **740** and signal **750** are complements of each other, so conduction should only occur along one path in multiplexer **700** at any give time. Additionally, the opposing parasitic diodes of the transistors **730** in each conduction path should essentially block parasitic current when a given conduction path is turned off. Note that transistors **730** are described as power Metal-Oxide Semiconductor Field-Effect Transistor—MOSFETs (e.g. transistors **730** have drains), but other power transistors may be appropriate, provided that opposing parasitic components can be incorporated.

**[0039]** Much of the functions involved in managing power supply are handled by the voltage supervisory block of various embodiments. FIG. **8** illustrates an embodiment of a voltage supervisory system. While various embodiments may be used, voltage supervisory block **800** represents one embodiment which may be useful for providing power to a memory module. Voltage supervisory block **800** compares a system voltage with a reference voltage, and generates output signals with various logic components to control other parts of the power supply circuitry.

**[0040]** Voltage supervisory circuitry **835** compares a system power signal **825** with a voltage reference signal **820** and provides a power fail signal **845**. Voltage reference signal **820** is produced from a voltage rail **805** and a resistive divider composed of a 50 kohm resistor **810** in series with a 27 kohm resistor **815** to ground. Presumably, voltage rail **805** is powered from a secure power supply such as a battery power supply. Voltage supervisory circuitry **835** also is coupled to ground through a capacitor **850**.

**[0041]** The power fail signal **845** is pulled up to power rail **830** through resistor **840**, and feeds into inverter **855**. This produces a voltage detection reset signal **885**, and feeds into a buffer **860** to produce a voltage detection buffered signal **870**. OR gate **865** evaluates signal **870** and a finite state machine signal **875** to produce comparison signal **858** and through an inverter **855** to produce inverted comparison signal **868**. Comparison signal **858** and inverted comparison signal **868** may be used as a logically paired set of signals to control a power multiplexer such as that of FIG. **7**.

**[0042]** Voltage detection reset signal **885** is fed into flip-flop **880** which is clocked by clock signal **890**. The output of flip-flop **880** is fed to AND gate **895** along with an inverted version of signal **885** to produce a pulse at output **898**. AND output **898** is used as the chip enable signal to flip-flop **888**, which is also clocked with clock signal **890** and uses a power supply signal **825** as an input to produce finite state machine signal **875**. One-Shot reset signal **878** resets flip-flop **888**, thereby clearing signal **875**.

**[0043]** While voltage supervisory circuitry provides signals to indicate what voltage signal should be supplied as a power source, clock circuitry may be needed in some systems to maintain synchronous operations, or just to drive periodic processes. FIG. **9** illustrates an embodiment of clock circuitry. Several different clock sub-circuits are illustrated. Circuit **910** uses a 25 MHz oscillator **915** which is enabled by enable **925** to produce a 25 MHz clock **920**. Circuit **930** uses oscillator **955**, coupled through capacitors **960** to voltage rail **965**, to produce an oscillating signal which is used by clock detector **935** in detecting system clock **940**, and thereby producing 166 MHz clock **945**, along with a safe/fail signal **950**. Similarly, circuit **970** uses oscillator **975**, coupled through bypass capacitors **960** to power rail **965** with clock detector **985** to detect 25 MHz clock signal **925** and to produce 25 MHz clock **990** along with safe/fail signal **980**. A safe clock signal is produced by circuit **995**, using 166 MHz clock **945**, 25 MHz clock **990** and safe/fail signal **950** to determine the output (safe clock **988**) of programmable clock component **993**. Safe clock **988** provides an input to phase locked loop **983**, which produces PLL output signals **978** and uses a feedback loop to maintain a clock signal.

**[0044]** A processor system on a chip may be used to provide various functionality, in particular generation of control signals for a finite state machine, for example. FIG. **10** illustrates an embodiment of a processor system on a chip. Processor system on a chip **1010** may be implemented as a Cypress processor system on a chip, or through other digital signal processors or similar devices.

**[0045]** A debug port **1005** is provided, a reset input **1080** is provided, and a clock generator **1070** provides a clock **1075**. Internally, PSOC (Programmable System On a Chip) **1010** includes an embedded microprocessor module **1040**, an I2C controller module **1050**, peripheral interface **1015**, local SRAM **1020**, local FLASH ROM **1025**, and two timers (TRP timer **1060** and TRF timer **1065**), all of which are coupled through bus **1030**. Processor module **1040** may be programmed to generate finite state machine control signals **1085**, and to control timers **1060** and **1065** to generate timer signals **1090**. Similarly, an interface with an SMBUS may be provided through signals **1095**. Moreover, I2C controller **1050** allows for communication with a surrounding system. FLASH ROM **1025** may store code and SRAM **1020** may store local variables.

**[0046]** Control of a state machine allows for a state machine to be implemented and operated, such as in a complex programmable logic device (e.g. a CPLD available from Xilinx). FIG. **11** illustrates an embodiment of a state machine. State machine **1100** may be used to refresh SDRAM of a memory module when the SDRAM is not

otherwise the subject of operations in a normal system mode. Essentially, one may expect that SDRAM will be properly refreshed when system power is present and a system operates normally, but if SDRAM is to retain memory in a power fail situation, refresh operations must be implemented without benefit of other parts of a system, requiring that the memory module trigger SDRAM refresh operations itself.

**[0047]** State machine **1100** is initialized with a reset at idle state **1110**. When a system powers on, for example, one may expect the reset signal to cause the state machine **1100** to move to idle state **1110**. This avoids unexpected operation due to transient signals, for example. When the system is armed, the enable signal moves state machine **1100** into ready state **1120**. From here, the system may await a start signal, at which point it moves to start state **1130**. From start state **1130**, the state machine **1100** advances to wait state **1140**, based on chip select and clock enable signals being asserted. At wait state **1140**, a trf timer (such as a timer of PSOC **1010**) is started, and a signal from the trf timer results in an advance to precharge state **1150**. From precharge state **1150**, the state machine **1100** advances automatically to another wait state **1160**, with a trp timer (such as another timer of PSOC **1010**) started. A signal from the trp timer advances the state machine **1100** to self-refresh entry state **1170**, with self-refresh initiated for the memory module. The state machine then advances automatically to self refresh state **1180**, and stays there automatically self-refreshing the memory module. A reset signal or an exit signal will move the state machine **1100** back to idle state **1110**, ending the self-refresh process, such as when a system sufficiently recovers power.

**[0048]** All of these components can be added up to a memory power control system—a system which controls a power supply and clock provided to a memory module, and thereby provides backup power and clocking when system power and system clocks fail. FIG. **12** illustrates an embodiment of a memory control system. System **1200** includes a PSOC, a finite state machine, a signal multiplexer and a clock multiplexer, and thereby receives and selects both clock and memory control and command bus signals.

**[0049]** PSOC **1210** is a processor system on a chip or similar processor, including a UART (Universal Asynchronous Receiver, Transmitter) serial port **1260**, timers **1270** and **1280**, and an IIC (I2C) controller **1250**. Debug signals **1297** may interface with UART **1260**, clock and reset signals **1295** are supplied to the PSOC, and system bus signals **1290** also interface with I2C controller **1250**. This allows PSOC **1210** to produce finite state machine signals **1215** and timer signals **1225**. These signals are used to control finite state machine **1230**, which produces signals that control refresh of memory of a memory module. This also produces signal logic multiplexer select signal **1255**, which selects an input for multiplexer **1240**. Multiplexer **1240** is a logic signal multiplexer, accepting as input a set of system control and command bus signals **1235** and sets of finite state machine control and command bus signals **1245** to produce memory control and command bus signals

**[0050]** During normal operation of the system with proper power and/or clock operation the system control and command bus input to signal logic multiplexer **1240** will be chosen by multiplexer select signal **1255**. During either power interruption or system clock fault cycles the finite state machine control and command bus signals will be supplied as memory control and command bus signals through the other inputs of signal logic multiplexer **1240**—and selected by multiplexer select signal **1255**. A separate multiplex select signal **1275** selects as input either a system clock **1277** or a 25 MHz clock **1285** as inputs to clock multiplexer **1220** which provides as output a PLL input clock signal **1265**. Clock multiplexer **1220** operates to not only select a clock output signal, but also to make sure that a switch from one clock signal to another clock signal does not cause edges to come so quickly as to simulate a clock rate higher than that specified for the system and memory module.

**[0051]** While a general approach to a clock and power fault detection system provides much insight, a truth table for a specific implementation may also be useful. FIG. **13** illustrates an embodiment of a truth table. Truth table **1300** represents the values of signals to memory components of a DDR2 SDRAM memory module as would be generated to control the clock and power fault detection system and the actual memory (for write protection and refresh purposes).

**[0052]** Whether the embodiments illustrated in the various figures are used, or alternative embodiments are used to provide clock and power fault detection for a memory module, various processes may be implemented to control such a clock and power fault detection system. FIG. **14** illustrates an embodiment of a process of controlling clock and power fault detection for a memory system. Process **1400** includes determining if clock and power fault detection is enabled, determining if power and clock signals are stable, providing clock and power fault detection, determining if the system power and clock signals have been restored, and operating the memory module normally. Process **1400** and other processes of this document are implemented as a set of modules, which may be process modules or operations, software modules with associated functions or effects, hardware modules designed to fulfill the process operations, or some combination of the various types of modules, for example. The modules of process **1400** and other processes described herein may be rearranged, such as in a parallel or serial fashion, and may be reordered, combined, or subdivided in various embodiments.

**[0053]** Process **1400** initiates at start module **1410**. At module **1415**, a determination is made as to whether a battery back bit or similar signal is set. If not, the system remains in a state at module **1420** where clock and power fault detection is not initiated if power or clock signals fail. This may be due to the system being manually powered down (a user turns it off) or the system not having enabled clock and power fault detection for whatever reason. At module **1425**, the process stops with power going off.

**[0054]** If the clock and power fault detection bit is set, or some other signal is set to enable clock and power fault detection, then the process checks at module **1430** and **1435** whether a clock signal for the memory module has collapsed (module **1430**) or whether the power supply is below a reference voltage (module **1435** and 3.0 V for a 3.3 V rail in one embodiment). If not, the process repeats the check—looping until something does happen. If one or both of the clock collapsing or the power supply diminishing occurs, then at module **1440**, the process moves the memory module into a clock and power fault detection state. This may include setting a power fail interrupt bit in a register, indicating that a power failure was detected (or a clock failure) and the memory module assumed control.

**[0055]** This involves providing battery power and providing a battery-powered clock signal as well. At module **1450**, the process also moves the memory module into a self-refresh state, where refresh is controlled by the memory module's on-board circuitry, rather than by system signals. The process then goes to checking whether the system has restored normal operation. At module **1460**, a check is made as to whether the clock signal is restored. At module **1470**, a check is made as to whether the power rail is restored. As illustrated, the 3.3 V power rail is checked, but other power rails, such as a 1.8 V power rail in some embodiments, may be checked instead. However, one may expect that a higher voltage power rail will droop before a lower voltage power rail in many system implementations.

**[0056]** Additionally, at module **1480**, the module checks whether a bit has been cleared indicating the system is ready to take back control of the memory module. In one embodiment, this is a power fail interrupt bit. Typically, each of modules **1460**, **1470** and **1480** must be satisfied—the power supply must be stable, the clock must be stable, and the power fail bit must be clear—before the memory module will release operation back to the system. This allows for screening of results caused by transient signals, for example. From module **1490**, the system resumes control, and the process returns to its initial state at module **1410**.

**[0057]** Another example of a similar process may provide further understanding of how one may provide power and clock monitoring for a memory module. FIG. **15** illustrates an alternate embodiment of a process for monitoring clock and power for a memory module. Process **1500** includes initiating monitoring, detecting a fault, initiating a backup system, maintaining memory, detecting potential restoration, checking whether a system is ready, and restoring control to the system.

**[0058]** In this embodiment, process **1500** initiates by initiating monitoring at module **1510**. This may involve detecting that a bit is set enabling monitoring, for example. Alternatively, it may involve detecting that startup has completed in a system (e.g. a set time after a voltage level is detected for power), and automatically initiating monitoring at that point. At module **1520**, the system is monitored, with power and clock signals checked for proper levels (power) or proper signals (clock). At module **1530**, a fault is detected, whether in the clock or power signals (or both).

**[0059]** The backup system is initiated at module **1540**, with the system taking over supply of the clock and power signals. Both signals are supplied to avoid confusion about what must be restored later—and because a fault in one signal is likely to lead to a fault in the other signal. Part of initiation may include asserting a bit in a register of the memory module indicating the module is in backup mode. With the backup system operating, the memory module is maintained at module **1550**—power is supplied from a battery and a clock signal is generated to allow the module to refresh memory.

**[0060]** A determination is then made as to whether a restored signal is detected for the clock and power signals at module **1560**. If either signal is not properly detected (fails to meet minimum criteria), the process moves back to module **1550** and maintains memory. If both signals are detected, the process then determines at module **1570** whether the system has signaled that it is ready to handle power and clock signaling. One signal that may be used is clearing the asserted bit of the memory module which was asserted to indicate backup mode at module **1540**. Thus, the bit may be asserted for backup mode by the module and cleared for restoration by the system. If the system has not signaled that it is ready to handle power and clock signals, then the process moves back to module **1550**. If the system has signaled that it is ready to handle power and clock signals, then control is returned to the system at module **1580**, and the process returns to module **1510** (or potentially to module **1520**).

**[0061]** One skilled in the art will appreciate that although specific examples and embodiments of the system and methods have been described for purposes of illustration, various modifications can be made without deviating from the present invention. For example, embodiments of the present invention may be applied to many different types of databases, systems and application programs. Moreover, features of one embodiment may be incorporated into other embodiments, even where those features are not described together in a single embodiment within the present document.

What is claimed is:

**1**. A system, comprising:

a voltage detection circuit;

a clock detection circuit;

a controller coupled to the voltage detection circuit and the clock detection circuit;

a memory control state machine coupled to the controller;

volatile memory coupled to the memory control state machine;

a battery;

and

battery regulation circuitry coupled to the controller and the memory control state machine;

and wherein the battery, battery regulation circuitry, volatile memory, memory control state machine, controller, clock detection circuit and voltage detection circuit are all collectively included in a unitary memory module.

**2**. The system of claim **1**, further comprising:

a power multiplexer coupled to receive a power supply of the battery and a power supply of a surrounding system and to provide a power supply to the volatile memory.

**3**. The system of claim **2**, wherein:

the power multiplexer includes:

a series of two drain coupled power MOSFETs coupled between the power supply of the battery and the power supply of the volatile memory;

and

a series of two drain coupled power MOSFETs coupled between the power supply of the surrounding system and the power supply of the volatile memory.

**4**. The system of claim **1**, wherein:

the controller is implemented as a processor system on a chip.

**5**. The system of claim **1**, wherein:

the memory control state machine is implemented as a complex programmable logic device.

**6**. The system of claim **1**, wherein:

the volatile memory is DDR2 SDRAM.

**7**. The system of claim **1**, wherein:

the volatile memory is DDR SDRAM.

**8**. The system of claim **1**, wherein:
the battery regulation circuitry includes clock generation circuitry.
**9**. The system of claim **1**, wherein:
the battery regulation circuitry includes battery recharge circuitry.
**10**. The system of claim **1**, wherein:
the battery regulation circuitry includes battery monitor circuitry.
**11**. A system, comprising:
a processor;
a bus coupled to the processor;
a memory control hub coupled to the bus;
a memory module coupled to the memory control hub;
and wherein the memory module includes:
a voltage detection circuit;
a clock detection circuit;
a controller coupled to the voltage detection circuit and the clock detection circuit;
a memory control state machine coupled to the controller;
volatile memory coupled to the memory control state machine;
a battery;
and
battery regulation circuitry coupled to the battery, the controller and the memory control state machine.
**12**. The system of claim **11**, further comprising:
means for interfacing with peripheral components of the system.
**13**. The system of claim **11**, further comprising:
a power multiplexer coupled to receive a power supply of the battery and a power supply of a surrounding system and to provide a power supply to the volatile memory;
a clock multiplexer coupled to receive a memory module-integrated battery backup clock and a system clock and to provide a memory module clock;
and wherein:
the controller is implemented as a processor system on a chip;
and
the memory control state machine is implemented as a complex programmable logic device.
**14**. A method, comprising:
detecting an unacceptable variation in a voltage or clock signal to a memory module;
initiating backup operations within the memory module;
and
maintaining memory of the memory module.
**15**. The method of claim **14**, further comprising:
detecting a recovery status of a system coupled to the memory module.
**16**. The method of claim **15**, further comprising:
returning control of the memory module to the system coupled to the memory module.
**17**. The method of claim **14**, wherein:
maintaining memory of the memory module includes setting memory components into a self-refresh state.
**18**. The method of claim **14**, wherein:
initiating backup operations within the memory module includes:
switching to a battery power supply integrated with the memory module;
and
switching to a clock signal produced by a component of the memory module.
**19**. The method of claim **15**, wherein:
detecting a recovery status of a system coupled to the memory module includes:
detecting an acceptable clock signal from the system;
and
detecting an acceptable power signal from the system.
**20**. The method of claim **19**, wherein:
detecting a recovery status of a system coupled to the memory module further includes:
detecting clearance of a predetermined bit of the memory module.

* * * * *