

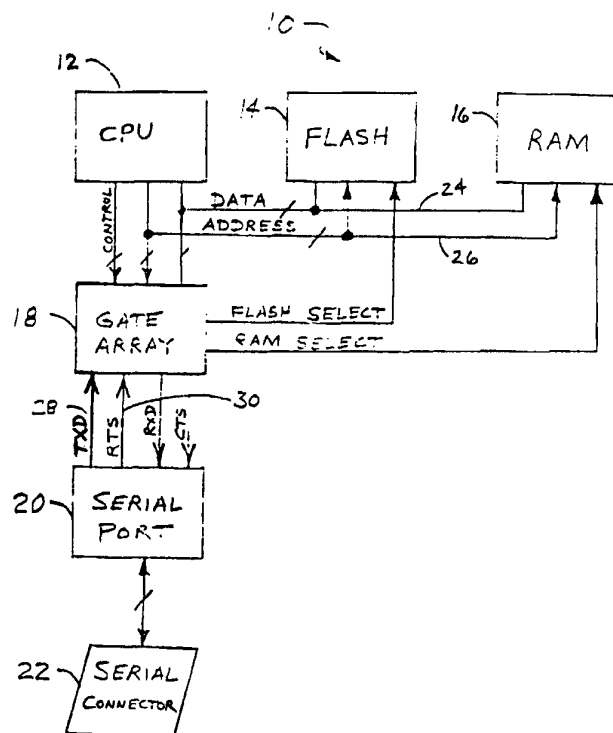


INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification ⁶ : G06F 9/445, 13/12	A1	(11) International Publication Number: WO 97/09673 (43) International Publication Date: 13 March 1997 (13.03.97)
(21) International Application Number: PCT/US96/14082 (22) International Filing Date: 3 September 1996 (03.09.96) (30) Priority Data: 60/003,000 7 September 1995 (07.09.95) US 08/566,877 4 December 1995 (04.12.95) US (71) Applicant: METRICOM, INC. [US/US]; 980 University Avenue, Los Gatos, CA 95030 (US). (72) Inventor: PAULSEN, David, L.; 708 Waltham Street, Mountain View, CA 94040 (US). (74) Agents: ALLEN, Kenneth, R. et al.; Townsend and Townsend and Crew L.L.P., 8th floor, Two Embarcadero Center, San Francisco, CA 94111-3834 (US).		(81) Designated States: CA, CN, JP, MX, European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE). Published <i>With international search report.</i> <i>Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.</i>

(54) Title: TRANSFERRING CODE TO COMPUTER WITHOUT ACCESSING RAM**(57) Abstract**

A device with a start-up mode after reset whereby boot-up instructions are by default always accepted directly from an external source. The device includes a gate array (18), connected directly via a serial port (20) to an input device (22), to receive program code for transfer directly as instructions to a CPU (12). The CPU is part of a computer controlled device (CCD) containing a microprocessor, RAM (16), and a flash memory device (14), the flash memory being unprogrammed when the CCD is fabricated. When the CCD first powers up, a special mode of operation ensues in which the gate array (18) directly monitors bits received and shifts them into a local register from which bytes are loaded into the CPU (12) as instructions. A boot sequence is thereby accepted to load RAM (16) of CPU (12) which thereafter can control transfer of additional executable code or load a permanent boot sequence into bulk erase (nonvolatile) flash memory (14).



FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AM	Armenia	GB	United Kingdom	MW	Malawi
AT	Austria	GE	Georgia	MX	Mexico
AU	Australia	GN	Guinea	NE	Niger
BB	Barbados	GR	Greece	NL	Netherlands
BE	Belgium	HU	Hungary	NO	Norway
BF	Burkina Faso	IE	Ireland	NZ	New Zealand
BG	Bulgaria	IT	Italy	PL	Poland
BJ	Benin	JP	Japan	PT	Portugal
BR	Brazil	KE	Kenya	RO	Romania
BY	Belarus	KG	Kyrgyzstan	RU	Russian Federation
CA	Canada	KP	Democratic People's Republic of Korea	SD	Sudan
CF	Central African Republic	KR	Republic of Korea	SE	Sweden
CG	Congo	KZ	Kazakhstan	SG	Singapore
CH	Switzerland	LI	Liechtenstein	SI	Slovenia
CI	Côte d'Ivoire	LK	Sri Lanka	SK	Slovakia
CM	Cameroon	LR	Liberia	SN	Senegal
CN	China	LT	Lithuania	SZ	Swaziland
CS	Czechoslovakia	LU	Luxembourg	TD	Chad
CZ	Czech Republic	LV	Latvia	TG	Togo
DE	Germany	MC	Monaco	TJ	Tajikistan
DK	Denmark	MD	Republic of Moldova	TT	Trinidad and Tobago
EE	Estonia	MG	Madagascar	UA	Ukraine
ES	Spain	ML	Mali	UG	Uganda
FI	Finland	MN	Mongolia	US	United States of America
FR	France	MR	Mauritania	UZ	Uzbekistan
GA	Gabon			VN	Viet Nam

5 TRANSFERRING CODE TO COMPUTER WITHOUT ACCESSING RAM

BACKGROUND OF THE INVENTION

10 This invention relates to techniques for starting computer systems and more particularly to a method for downloading executable code to a computer which is not provided with conventional boot code autonomous from the central processing unit.

15 All computing devices based on the model described by von Neumann require that a stored program be made available in memory for execution. Typically, such computing devices have a pre-loaded program (a "boot" program) in the form of executable instructions that contains sufficient logic to permit more extensive and higher-level programs to be loaded from external devices into the memory associated with the CPU. 20 In early (pre-1975) computers, this boot program could be loaded manually directly through front panel switches connecting directly to registers to "stuff" the appropriate data and control bits to be transferring into a volatile 25 memory associated with the CPU. However, modern computers lack a front panel, so this boot program is typically loaded into a pre-programmed read-only memory (ROM) device and automatically run upon application of power to the CPU. An example is the BIOS (Basic I/O System) Read Only Memory of an 30 IBM-type PC.

35 In embedded computing devices, in which a microprocessor, ROM, and RAM are combined to perform a dedicated function, there typically is no external storage device, such as a disk drive. In these systems, either the stored program is typically unchangeable, in which case it is combined with the boot function onto ROM. In these systems, updating the stored program requires opening the device and

changing a memory device, not an operation that a consumer would typically perform.

Alternatively, if a method is provided for a consumer to download a new stored program into the device, that typically requires that a boot program be resident inside the device and protected from consumer intervention.

The recent development of Flash EEPROM (Electrically Erasable Programmable Read-Only Memory) has made it possible to provide this function (namely, consumer download of new stored programs, typically an updated version) economically. Flash memory devices come in at least two forms --bulk erase, and boot-block. Bulk erase flash devices must be erased in toto. In order for a device using a bulk erase flash memory device to permit the consumer to update its stored program, the device must first erase its entire stored program, and then re-program it. If the process is interrupted after erasure but prior to enough of the stored program to be programmed (typically an interval of many seconds), the device is rendered inoperable. Furthermore, because the boot program has been erased, the device cannot be reloaded. Boot block flash memory devices have been devised to address this problem. In these devices, a portion of the memory is separately erasable, permitting a boot program to be stored there.

Unfortunately, boot block flash memory devices are expensive. In addition, they require that the memory device be pre-programmed prior to fabrication, which adds manufacturing cost to the device.

Alternatively, a separate memory device is required, which also adds manufacturing cost. What is needed is a scheme to permit code to be downloaded to a bulk erase flash memory device without requiring that the device be pre-programmed.

SUMMARY OF THE INVENTION

According to the invention, in a computer-driven device, a start-up mode after reset is provided whereby boot-up instructions are by default always accepted directly from

an external source. The device may comprise a gate array (GA) which is connected directly via a serial port to a receiving or input device, to receive program code for transfer directly as instructions to a microprocessor-type central processing unit. The central processing unit is part of a computer controlled device containing a microprocessor, memory (a RAM), and typically a bulk erase flash memory device, the flash memory device being unprogrammed when the computer controlled device is fabricated originally. When the computer controlled device first powers up, a special mode of operation ensues in which the gate array directly monitor bits received via the serial port from the input device and shifts them into a local register from which bytes or words are loaded directly into the CPU as instructions. A boot sequence is thereby accepted in this manner to load random access memory of the free-running CPU which thereafter can control transfer of additional executable code or load a permanent boot sequence into bulk erase (nonvolatile) flash memory. The invention can be incorporated into a wireless modem/packet terminal node controller. The input device may be a coupling to an external computer system, typically a serial input.

The invention will be better understood by reference to the following detailed description in connection with the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a block diagram of the system having a CPU, flash memory, RAM, gate array, serial port and receiving device, according to the invention.

Figure 2A is a schematic of a first version of circuitry inside the gate array which, in conjunction with the data being shifted in, forms the boot loader.

Figure 2B is a schematic of a second version of circuitry inside the gate array which, in conjunction with the data being shifted in, forms the boot loader.

Figure 3 is a flowchart illustrating the sequence of operations according to the invention.

DESCRIPTION OF SPECIFIC EMBODIMENTS

Referring to Fig. 1, there is shown a block diagram of a computer-based system 10 (such as a terminal node controller) according to the invention having a CPU 12, flash memory 14, RAM 16, gate array 18, serial port 20 and receiving device 22. The CPU 12, flash memory 14, RAM 16, and gate array 18 are connected to a data bus 24 and an address bus 26. The gate array is coupled to the serial port 20 which in turn is coupled to receiving device 22. The receiving device is for example a serial connector. The gate array 18 has data i/o, address input and control input to/from the CPU 12, and a serial input line 28 and a control line 30. (The gate array may also provide a serial output path; however, that signal path is not pertinent to the present invention.)

Referring to Fig. 2A, there is shown a schematic diagram of a first version of the gate array 18. There is a serial I/O section 32, a receiving register section 34, and a CPU instruction control section 36. In this version, timing circuitry 38 is provided to automatically terminate the default boot sequence in the event no valid request for instruction is received.

Referring to Fig. 2B, there is shown a schematic diagram of a second version of the gate array 18. There is a serial I/O section 132, a receiving register section 134, and a CPU instruction control section 136. In this version, no timing circuitry is provided to automatically terminate the default boot sequence in the event no valid request for instruction is received. An external "push" signal 138 is provided which must be present to activate and maintain the boot sequence.

Referring to Fig. 3, there is shown a flow chart of the boot method according the invention. Upon reset or power up (Step A), the "virgin flash" mode is initiated (Step B) and the gate array looks for instruction requests from the CPU 12 (Step C). (In the preferred embodiment of Fig. 2A, the gate array waits for about 16 ms to determine if a valid instruction request is received. In the alternative embodiment of Fig. 2B, the "push" signal 138 must remain

asserted throughout the "virgin flash" mode. These alternative functions control the exit mechanism.)

The receiving register section 134, the so-called virgin receive register, routinely accepts serial data each cycle from the serial port 20. Each cycle the register is checked to see if it is full (Step E). If not, a filler instructions, such as a No-op, a jump-to-self instruction or equivalent instruction, is returned to the CPU (Step F). If it is full, the content of the virgin register is sent to the CPU as an instruction (Step G) and the register is cleared for receipt of the next stream of bits to form instructions. (A "one" bit is sent via the serial port as a start bit for the beginning of each instruction bit stream.)

The invention allows a CPU to execute instructions directly from external sources totally bypassing local memory. It is noteworthy that the instructions are executed one at a time as received, not waiting for an accumulation of a series of instructions, as in the case of conventional front-panel loading of boot instructions. This invention could even be used in applications of computer systems which have no RAM.

The invention could also be used in connection with a parallel input port by appropriate choice of an input register to receive and output data in parallel, e.g., a dual-port parallel register. What is important is a start-up mode after reset which is the acceptance of instructions directly from an external source. The invention can be used in a number of applications, such as loading BIOS's into PC's; updating code in PDA's.

The invention has now been explained with reference to specific embodiments. Other embodiments will be apparent to those of skill in the art. It is therefore not intended that this invention be limited, except as indicated by the appended claims.

WHAT IS CLAIMED IS:

1 1. A method for downloading a boot sequence to a
2 computing device having an input element, a gate array, and a
3 CPU, said method in said gate array comprising the steps of:
4 upon reset looking for instruction requests from the
5 CPU;
6 accepting at a receiving register digital data from
7 said input element;
8 checking said register each cycle to determine if
9 said register contains an instruction;
10 returning a filler instruction to the CPU until said
11 register contains a desired instruction; and thereupon
12 transferring content of said register to the CPU as
13 said desired instruction.

1 2. The method according to claim 1 wherein said
2 input element is a serial input device and wherein said
3 register is a shift register.

1 3. The method according to claim 1 wherein said
2 register is cleared upon said transferring step for receipt of
3 the next bits to form instructions.

1 4. The method according to claim 3 wherein a "one"
2 bit is sent via the serial port as a start bit for the
3 beginning of each instruction bit stream.

1 5. The method according to claim 1 wherein said
2 input element performs parallel data input.

1 6. The method according to claim 1 wherein a
2 timeout is used to exit from a default state of receiving
3 instructions from an external source.

1 7. An apparatus for downloading a boot sequence to
2 a computing device, said apparatus comprising:

3 an input element for receiving digital data from an
4 external source;
5 a receiving register means coupled to receive said
6 digital data from said input element;
7 means for checking said register each cycle to
8 determine if said register contains a desired instruction; and
9 means for repetitively transferring each said valid
10 instruction to said CPU.

1 8. The apparatus according to claim 7 wherein said
2 input element is a serial input device.

1 9. The apparatus according to claim 7 wherein said
2 input element is a parallel input device.

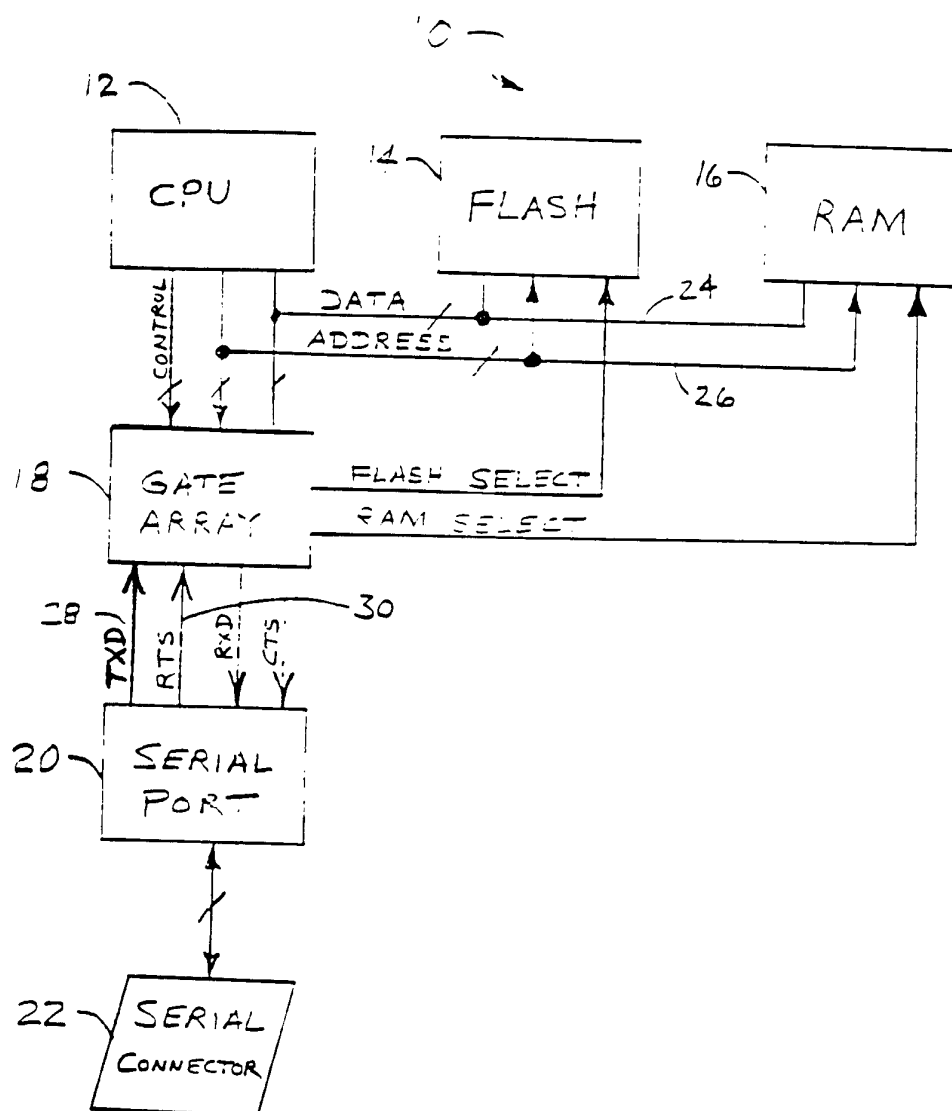


FIG. 1

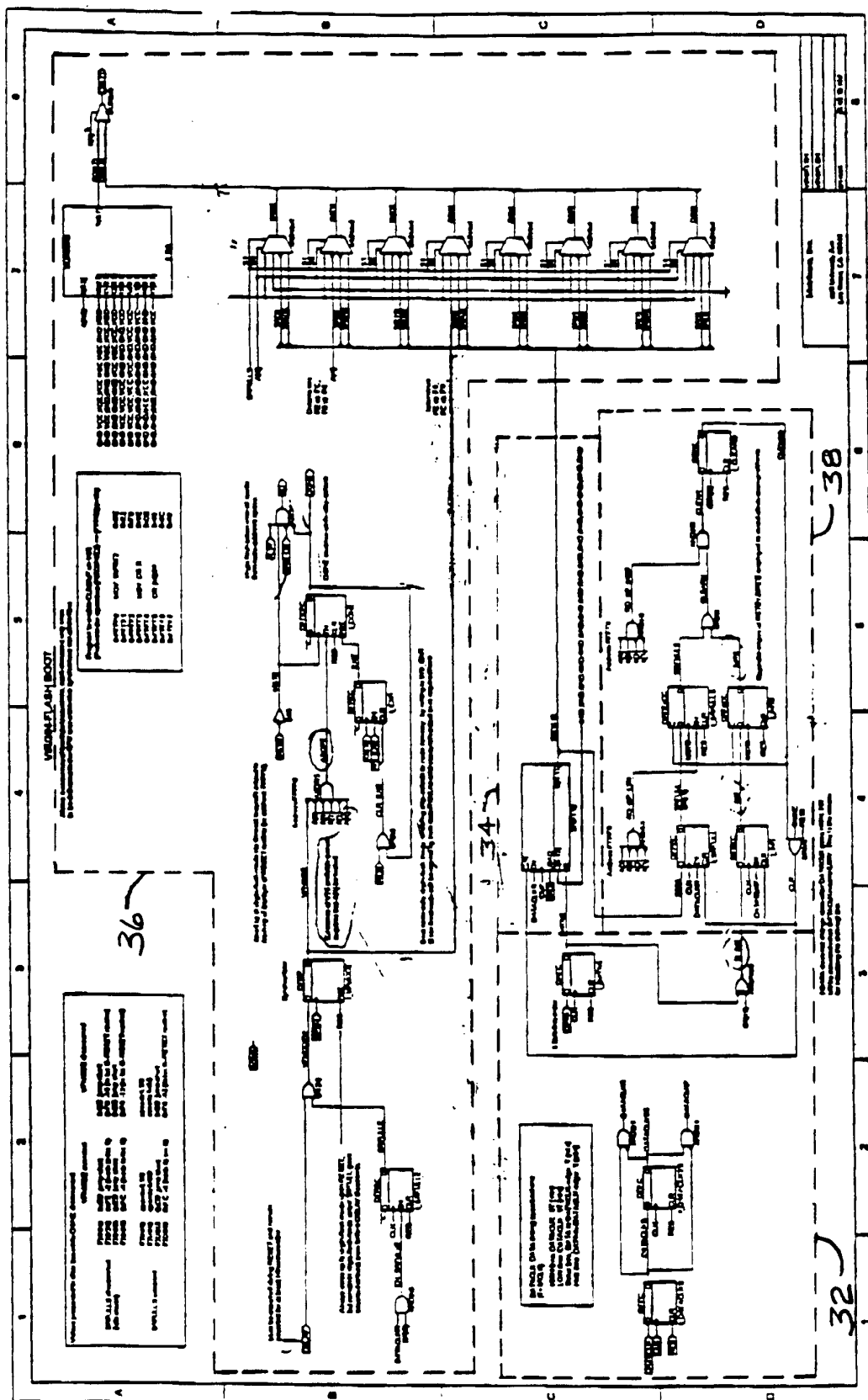


FIG. 2A

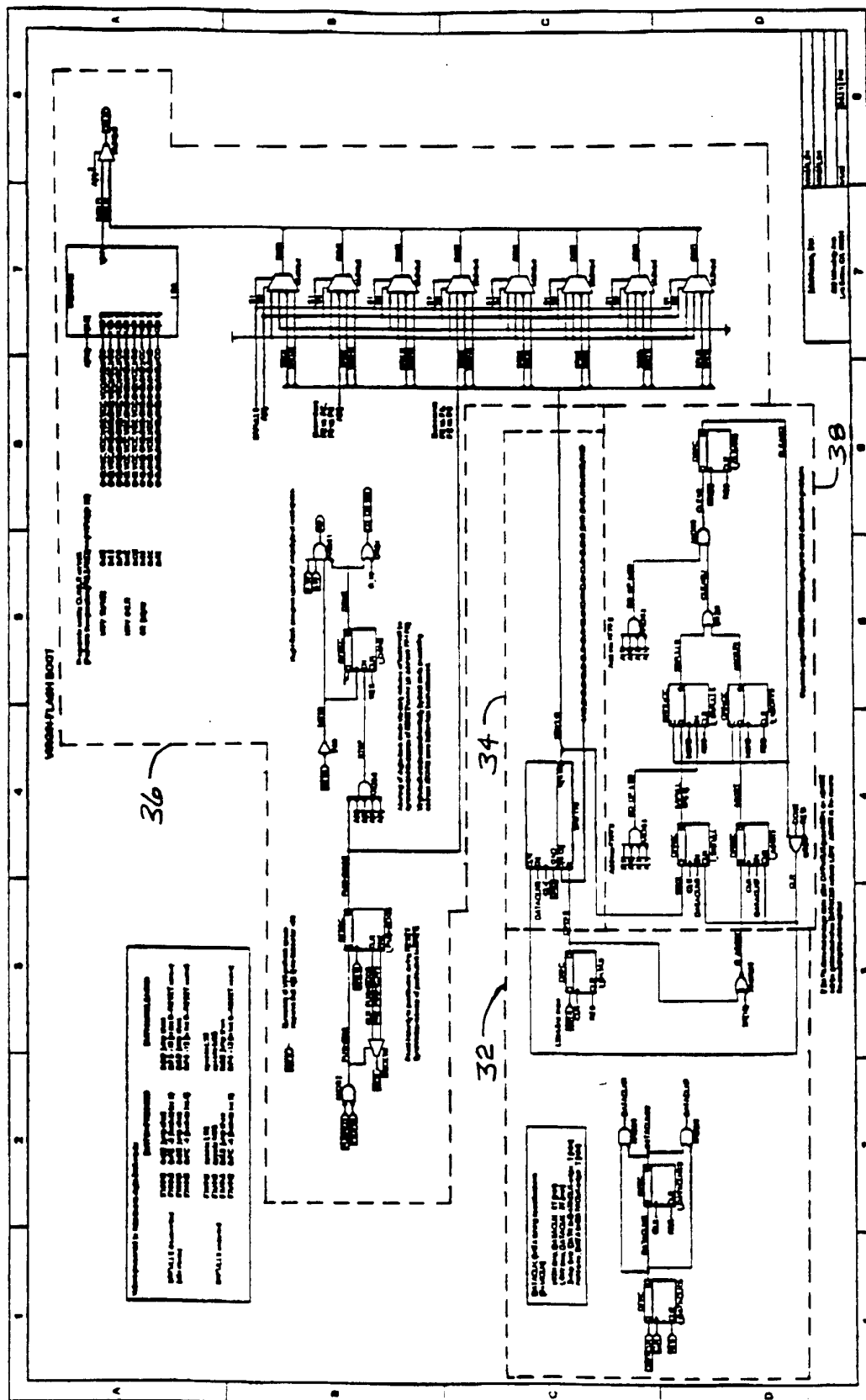


FIG. 2B

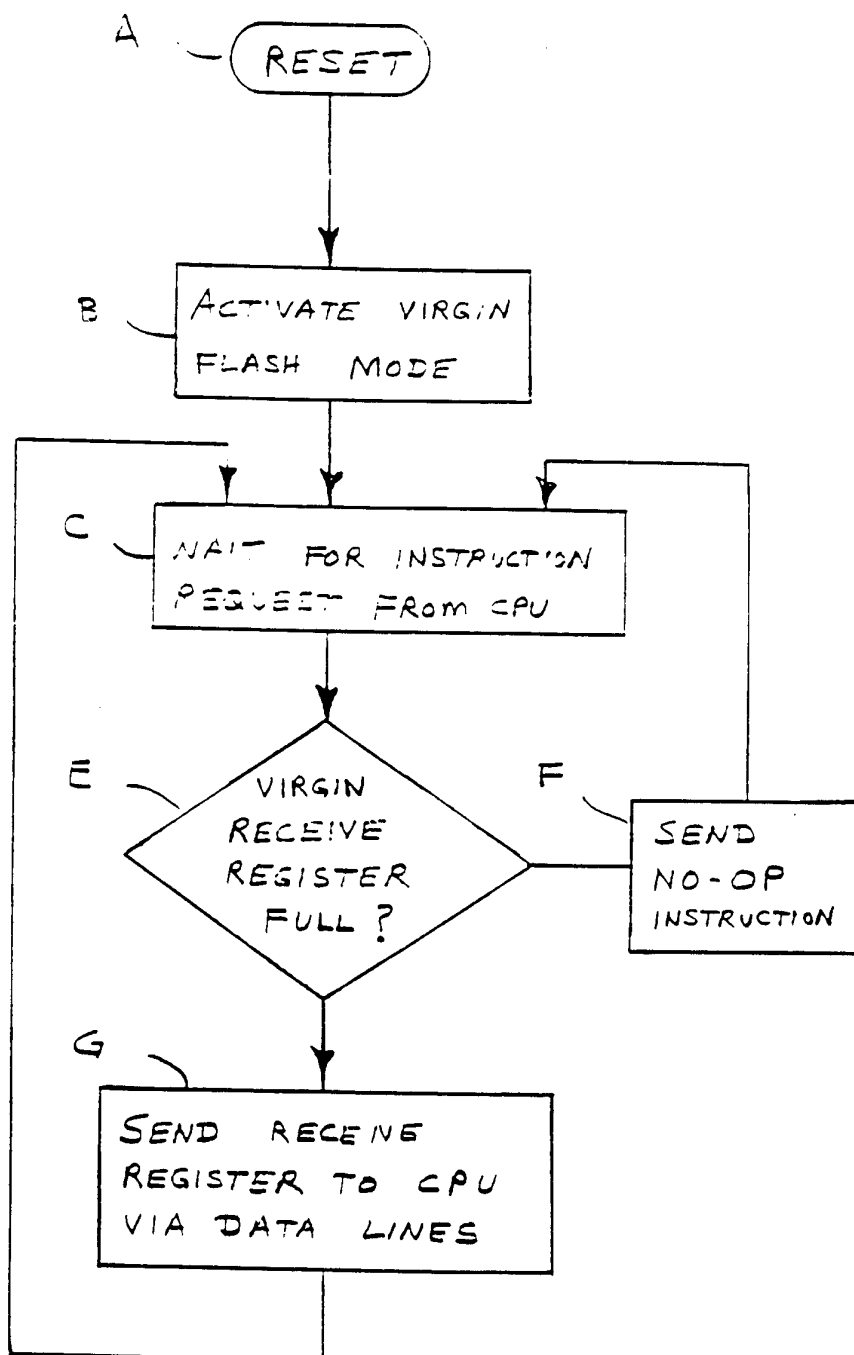


FIG. 3

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US96/14082

A. CLASSIFICATION OF SUBJECT MATTER

IPC(6) :G06F 9/445, 13/12

US CL :395/700

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 395/700

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
APS, JPO, STN

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 5,408,624 A (RAASCH ET AL) 18 April 1995, see the entire document	1,5,7-9
A	US 5,325,529 A (BROWN ET AL) 28 June 1994, col 1 line 1-col 7 line 2	1,5,7-9
A	US 5,444,861 A (ADAMEC ET AL) 22 August 1995, see the entire document	1,5,7-9

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

* Special categories of cited documents:	*T	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
A document defining the general state of the art which is not considered to be of particular relevance	*X*	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
E earlier document published on or after the international filing date	*Y*	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
L document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	*Z*	document member of the same patent family
O document referring to an oral disclosure, use, exhibition or other means		
P document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search

05 DECEMBER 1996

Date of mailing of the international search report

07 JAN 1997

Name and mailing address of the ISA/US
Commissioner of Patents and Trademarks
Box PCT
Washington, D.C. 20231

Facsimile No. (703) 305-3230

Authorized officer

JOHN Q. CHAVIS

Telephone No. (703) 305-9600