A non-volatile memory (12) connected to state logic (20) for autoloading peripheral target devices at power-on or system reset. The memory is preloaded (40) with commands and data. At power-on or system reset the commands are executed in sequence, transferring data to selected target devices. Data is output in bit-serial fashion on a single line (18). Target devices are individually selected through use of separate clock lines (24-30). Clock signals on the clock lines can be internally generated using the state logic or a target-device-supplied clock can be received under program selection. The system reset signal (36) is intercepted and retransmitted (38) to control target device mode. System reset polarity, enable signal polarity, data block length, clock direction, internal clock frequency, and power-saving shutdown upon completion of all transfers are all programmably selectable command features.
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>AT</td>
<td>Austria</td>
<td>FR</td>
<td>France</td>
<td>MR</td>
<td>Mauritania</td>
</tr>
<tr>
<td>AU</td>
<td>Australia</td>
<td>GA</td>
<td>Gabon</td>
<td>MW</td>
<td>Malawi</td>
</tr>
<tr>
<td>BB</td>
<td>Barbados</td>
<td>GB</td>
<td>United Kingdom</td>
<td>NE</td>
<td>Niger</td>
</tr>
<tr>
<td>BE</td>
<td>Belgium</td>
<td>GN</td>
<td>Guinea</td>
<td>NL</td>
<td>Netherlands</td>
</tr>
<tr>
<td>BF</td>
<td>Burkina Faso</td>
<td>GR</td>
<td>Greece</td>
<td>NO</td>
<td>Norway</td>
</tr>
<tr>
<td>BG</td>
<td>Bulgaria</td>
<td>HU</td>
<td>Hungary</td>
<td>NZ</td>
<td>New Zealand</td>
</tr>
<tr>
<td>BJ</td>
<td>Benin</td>
<td>IE</td>
<td>Ireland</td>
<td>PL</td>
<td>Poland</td>
</tr>
<tr>
<td>BR</td>
<td>Brazil</td>
<td>IT</td>
<td>Italy</td>
<td>PT</td>
<td>Portugal</td>
</tr>
<tr>
<td>BY</td>
<td>Belarus</td>
<td>JP</td>
<td>Japan</td>
<td>RO</td>
<td>Romania</td>
</tr>
<tr>
<td>CA</td>
<td>Canada</td>
<td>KP</td>
<td>Democratic People's Republic of Korea</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CF</td>
<td>Central African Republic</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CG</td>
<td>Congo</td>
<td>KR</td>
<td>Republic of Korea</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CH</td>
<td>Switzerland</td>
<td>KZ</td>
<td>Kazakhstan</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CI</td>
<td>Côte d'Ivoire</td>
<td>LI</td>
<td>Liechtenstein</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CM</td>
<td>Cameroon</td>
<td>LK</td>
<td>Sri Lanka</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CN</td>
<td>China</td>
<td>LU</td>
<td>Luxembourg</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CS</td>
<td>Czechoslovakia</td>
<td>LV</td>
<td>Latvia</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CZ</td>
<td>Czech Republic</td>
<td>MC</td>
<td>Monaco</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DE</td>
<td>Germany</td>
<td>MG</td>
<td>Madagascar</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DK</td>
<td>Denmark</td>
<td>ML</td>
<td>Mali</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ES</td>
<td>Spain</td>
<td>MN</td>
<td>Mongolia</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FI</td>
<td>Finland</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

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.
Description

Memory System for
Loading Peripherals on Power Up

Technical Field

The invention relates to integrated memory circuits, particularly adapted for conditioning a system at power-on or system reset.

Background Art

A current trend in digital equipment is toward the use of increasingly complex integrated circuits whose final behavior is determined by programmable parameters transferred at power-on or system reset. In a state-of-the-art notebook computer for example, the programming parameters of a number of "peripheral" devices are loaded under control of a microprocessor which transfers the information from a non-volatile memory to volatile storage located inside each such device. Typically, the microprocessor transmits the data serially, either supplying a clock signal or accepting a clock signal from the target device. Sometimes separate, simple microprocessors are provided for this purpose.

It has been suggested that a specialized memory, much less complex than a microprocessor could be used instead for this purpose (c.f., XC1736A/XC1765 Serial Configuration PROM, The Programmable Gate Array Data Book, Xilinx Inc., 1991). The specialized memory is pre-loaded with the parameters for a "target" peripheral device. These parameters are retained in non-volatile storage. At power-on or system reset the specialized memory serially transfers the contents of its non-volatile storage to the programmable peripheral device. The peripheral device is the source of a transfer clock.
The prior art teaches many of the elements needed to create such a specialized memory. U.S. Pat. No. 4,245,302 to Amdahl, for example, teaches a typical instruction fetch-execute organization for a programmable device wherein coded instruction fields are decodable to organize hardware resources for accomplishing a complex task. Mackey et al., U.S. Pat. No. 4,791,302, teach data transfer from a non-volatile PROM to a volatile RAM, and programmed selection of a clock signal from either internal or external sources. Guillot, U.S. Pat. No. 4,882,711, teaches a transfer of data from a non-volatile memory to volatile RAM at power-on, and writing into non-volatile EEPROM, all accomplished within a single integrated circuit. Fung et al., U.S. Pat. No. 4,899,272, teach the transfer of data from non-volatile memory to volatile configuration registers under program control. The non-volatile memory is implemented using CMOS and battery backup. U.S. Pat. No. 5,021,963, to Brown et al., teaches a transfer of data from a non-volatile memory to a volatile memory at power-on.

As discussed in the literature, the specialized memory is of limited utility because it cannot send initialization data to a plurality of target devices. Its simple function is limited to transferring the contents of its non-volatile memory to a single device at power-on or system reset. What is needed is a specialized memory capable of initializing a plurality of target devices at power-on or system reset.

Summary of Invention

The above object has been met by the present invention which combines a non-volatile memory, a finite state controller, and a clock generator whereby clock pulses appropriate to a target device are generated under control of the finite state controller.

The finite state controller uses a portion of the non-volatile memory for storage of information to implement its own operation. This stored information is
organized into "command blocks". One block of data for output to a target device is associated with each command block. The information in a command block is used by the sequential logic to perform a number of tasks including (1) locating output data in the memory, (2) determining how many bits of data are to be output, (3) selecting a particular target device, (4) determining whether an internal clock signal or a clock signal supplied from an external source, such as the target device will be used, (5) if an internal clock is selected, selecting its frequency, (6) determining whether an enable signal will be provided to the target device and if so what its polarity will be, (7) determining what polarity the system reset signal will be, and (8) determining whether additional target devices are to be loaded.

The command blocks and associated data blocks are preloaded into the non-volatile memory so that they are ready to function when the power-on condition is signalled by another part of the system. The information in memory is organized as pairs of command and data blocks. The pairs are stored in consecutive locations of memory to simplify the task of locating them.

A command block typically includes 8 bytes of information, but this may vary depending upon an application. The sequencing logic fetches a command block from memory, stores its bytes into registers, and uses the contents of these registers to control the outputting of the associated data block.

In an alternative implementation the memory is made available to the system as an addressable, serial storage device once the power-on transfers have been completed. The preloaded command and data blocks remain undisturbed in this embodiment. Previously unused memory space is organized by the sequencing logic as a serial store. A user provides a memory address in serial format and a read/write command, and serial data is output for a read or received and stored for a write relative to the user supplied address. A serial input, parallel output
register is used to receive addresses and write data. Addresses are transferred to the finite state controller while write data is stored into memory.

In yet another alternative, data is output in parallel format on a byte-wide bus. In this embodiment the output register receives data from the memory in parallel format and outputs the data in parallel format.

Brief Description of the Drawings

Fig. 1 is a block diagram of the specialized memory in accordance with the present invention.

Fig. 2 is a pictorial diagram showing an example of field and bit coding in a command block for the specialized memory of Fig. 1.

Fig. 3 is a partial block diagram of an alternative embodiment of the specialized memory of Fig. 1 showing a byte-wide data output.

Fig. 4 is a partial block diagram of a second alternative embodiment of the specialized memory of Fig. 1 showing a serial data input register.

Best Mode for Carrying Out the Invention

With reference to Fig. 1, there is shown specialized memory 10 for loading operating parameters into programmable integrated circuits on system power-on or reset.

Specialized memory 10 includes non-volatile memory 12, such as an EEPROM memory block, for storage of command and data blocks, parallel load/serial output register 14 for receiving data in parallel format from memory 12 via bus 16 and serially outputting the data on output line DATA_OUT 18, finite state controller 20 for receiving command blocks from memory 12 via bus 22 and for interpreting each command block to provide a data transfer clock signal at a selected output line CLOCK/EN 24-30, or for receiving an externally generated transfer clock signal at a selected input line CLOCK/EN 24-30, clock oscillator 32 for generation of a basic clock
signal, and clock divider 34 for reducing the frequency of the basic clock signal. Specialized memory 10 also includes input line RESET_IN 36 for intercepting a system reset signal, output line RESET_OUT 38 for retransmitting the system reset signal to target devices, and input line DATA_IN 40 for receiving data to be preloaded into non-volatile memory 12.

Non-volatile memory 12 is preloaded with command/data block pairs, the data intended for transfer to selected target devices upon receipt of a system reset signal used to indicate detection of a system power-on condition, and the associated command used by the finite state controller 20 to accomplish the data transfer.

During the data transfer, parallel load/serial output register 14 receives data from non-volatile memory 12 in parallel format via byte-wide bus 16. The register contents are serially shifted, one bit at a time, to output line DATA_OUT 18. In a typical system a number of target devices are connected to receive the output of line 18.

A data transfer cycle includes the fetch of a command block from non-volatile memory 12 followed by the execution of the command block. The execution of the command block entails the output of the associated data. Finite state controller 20, typically a Programmable Logic Array (PLA), is responsible for fetching and executing command blocks. In the embodiment shown in FIG. 1 non-volatile memory 12 stores command blocks as a group of related 8-bit bytes. Controller 20 causes memory 12 to place the bytes comprising a command block to be placed, one at a time on bus 22. Controller 20 receives each of these bytes, and when all the bytes comprising a command block have been received, controller 20 has completed the "fetch" portion of a data transfer cycle.

As mentioned above, the "execution" portion of the cycle entails the transfer of data from non-volatile memory 12 to output line DATA_OUT 18. Execution of the command block involves the following primary activities:
(1) reading the data block from non-volatile memory 12 one byte at a time; (2) loading each data byte read from memory 12, one at a time into output register 14 in parallel format; (3) shifting the contents of register 14 to serially output the data one bit at a time on output line DATA_OUT 18; (4) counting the number of data bits transmitted and comparing the count with a total number of bits to be transmitted; and (5) terminating the transfer when the proper number of bits has been sent.

In many systems, output data line DATA_OUT 18 is connected in common to an input line of each target device to which data is to be transferred. Because in such an implementation all target devices simultaneously receive the data intended for a single device, some means must be provided to insure that only the intended target devices loads its internal parameter storage with the data on the shared line. In the present invention this selection is accomplished by sending a clock signal to the intended target device only (or when a clock signal is supplied by the target device, by receiving a clock signal from the intended device only), hence multiple CLOCK/EN lines 24-30. When the clock is internally generated, and also during command block fetch, clock oscillator 32 supplies a basic clock signal. The frequency of this basic signal may be reduced by clock divider 34 so that the frequency of the transmitted clock signal, CLOCK/EN 24-30, can more accurately match the capabilities of a target device some of which operate at a much slower clock rate than specialized memory 10.

In a typical system, a reset signal is generated upon detection of the power-on condition or in the event of a system reset. In such systems this reset signal is often used to initiate the transfer of programming parameters to various programmable integrated circuits of the type with which the present invention is intended to be used. When used in such systems the inventor intends that the reset signal line will be broken and that specialized memory 10 will be inserted in the broken path,
intercepting the system's reset signal at input line RE-SET_IN 36, and retransmitting a substitute reset signal at output line RESET_OUT 38. When this proposed configuration is adopted, the system reset signal's being asserted on input line RESET_IN 36 is remembered by the finite state controller 20. At the transition of the system reset signal from the asserted to the non-asserted level, the basic data transfer cycle described above is commenced. The target devices interpret an active reset signal as requiring that they receive programming parameters. To ensure that all data transfers are completed, specialized memory 10 will continue to assert its reset output signal on line RESET_OUT 38 throughout the combined intervals of system reset active (input line RESET_IN 36) and data transfers in progress. RESET_OUT 38 can be used to inhibit the system from operating until all programmed parameters have been transferred to their respective target devices.

An example of a typical command block 50 is shown in FIG. 2. The command block 50 includes 8 bytes, numbered left-to-right, 7 through 0, respectively. Bytes 2-0 represent a 24-bit address which is a location in non-volatile memory 12 at which the first byte of the associated data block will be found. Bytes 5-3 represent the number of data bits which are to be transferred, i.e., the length in bits of the associated data block. Bytes 7 (52) and 6 (54) include coded fields and bits used to customize the data transfer or the behavior of specialized memory 10 itself. The bits in each of these bytes are numbered from left to right, bits 7 through 0, respectively.

The four-bit field 7-4 of byte 7 (52) of this example is decoded by the finite state controller 20 to select one of the four CLOCK/EN input/output lines 24-30 of FIG. 1 for transmitting or receiving a data transfer clock signal. Bit 7 of command byte 6 is decodable to select the internally generated clock signal for output on the CLOCK/EN line specified by bits 7-4 of command
byte 7. Bit 4-2 of byte 6 comprise a 3-bit field containing a number used by the clock divider 34 to reduce the basic frequency of an internally generated clock signal. By specifying the contents of these three elements of the command block 50, the direction, I/O line, and, if internally generated and within the limitations of the disclosed apparatus, the frequency of the data transfer clock can be fully determined.

Bits 6-5 of command byte 6 (54) provide a 2-bit field which allows for selection of one of four built-in serial transfer protocols, such as \(^2C\), MicroWire, Xilinx, or 2-Wire. MicroWire and Xilinx are trademarks. Details of these protocols can be found in the following publications: \(^2C\), 80C51-Based 8-bit Micro-controller, Section 2, \(^2C\) Circuit Bus, Signetics Co., 1992; Xilinx, The Programmable Gate Array Data Book, supra; 2-Wire, 80C51-Based 8-bit Micro-controller, pages 31-37, supra; and MicroWire, Electronic Design News (EDN), page 60, October 1, 1991. In the example, finite state controller 20 is capable of emulating each of these protocols. The protocol specified will be one that is compatible with the needs of the target device with which the command block is intended to cooperate to effect the parameters transfer.

Bits 3-2 of command byte 7 (52) of the example are used to cause specialized memory 10 to provide an enable signal to target devices which expect one. When bit 3 is asserted the finite state controller 20 will generate an enable signal whose polarity will be determined by the level of bit 2. In one embodiment the enable signal is transmitted on the CLOCK/EN line next in order below the selected clock line. For example, if bits 7-4 of command byte 7 are selecting line CLOCK/EN 24, the enable signal will be output on CLOCK/EN 26. In a second example, if clock line CLOCK/EN 30 is selected, the enable signal will be output on CLOCK/EN 24. In this manner, the same I/O pins and system interconnections can be used for a dual purpose. In an alternative embodiment
(not shown in FIG. 2) the command block includes an additional byte which includes an enable select field similar to the clock select field of command byte 7. Use of this enable select field allows any CLOCK/EN line not being used for the data transfer clock signal to serve as an enable line. Such an arrangement provides increased flexibility in the connection of various target devices to the specialized memory 10.

Use of bit 1 of command byte 6 (54) as shown in the example of Fig. 2 allows the specialized memory 10 to accept and generate a reset signal having one of two polarities. Some systems use a reset signal which is "low active" while others use a reset signal which is "high active". Likewise the target devices found in each of these systems expect a reset signal of the correct polarity.

Bit 1 of command byte 7 informs the finite state controller 20 that a power-down power saving sequence is or is not to follow completion of the final data transfer cycle. Many modern integrated circuits for use in battery operated devices shut themselves down into a power conserving standby mode when not in use. Bit 1 of command byte 7 permits such operation of the specialized memory 10 to be enabled or overridden.

Bit 0 of command byte 6 is used to inform the finite state controller 20 that upon completion of a current data transfer cycle, controller 20 is to fetch another command block to begin another data transfer cycle. The command block will be located in memory 12 such that its first byte is at a known location relative to the location of the last data byte of the current data transfer, typically the next byte.

Discussion of bit 0 of command byte 7 is reserved to a later point in this description.

Fig. 3 depicts an alternative embodiment which transfers output data via a byte-wide bus on data output lines DATA_OUT, 64. As shown, data is read from non-volatile memory 12 in byte-wide groups and loaded into
output register 60 which stores one byte. The contents of output register 60 are gated onto output lines DATA_OUTi 64 8 bits at a time. In this alternative embodiment the data block length field of the associated command block represents the length of the data block in bytes rather than in bits.

Fig. 4 depicts an alternative embodiment in which the specialized memory 10 is made available to the system for use as a randomly addressable serial memory following completion of the power-on data transfers. Bit 0 of command byte 7 of Fig. 2 is used to select whether or not the memory shall operate in this mode. When the auxiliary memory mode is selected, upon completion of execution of a command block having bit 0 of command byte 6 indicating "no more data", the finite state controller 20 organizes the specialized memory 10 to operate as a randomly addressable serial read/write memory.

In this alternative non-volatile memory 12 is replaced with a memory 70 which can be read from or written into and which is capable of retaining its data in the event of a power failure. There are a number of standard memory configurations suitable for this purpose.

Read/Write commands, memory addresses, data block length, and write data are received serially on input line DATA_IN 74 and are serially shifted into input register 72. Under control of the finite state controller 20, the commands and memory addresses are transferred, one byte at a time via bus 76 from input register 72 to finite state controller 20. Finite state controller 20 uses the command to determine whether to read or to write. It uses the memory address to locate the first data byte in the memory 70, and it uses the data block length to terminate the memory operation after the correct amount of data has been read or written. During a write operation, the data bytes are written 8 bits at a time from the input register to memory 70. During a read operation, data is transferred from the memory via bus 16 (Fig. 1) to serial output register 14 and serially
transmitted to the requester via data output line DATA_OUT 18. Alternatively the byte-wide output depicted in Fig. 3 can be used to return data read from the memory one byte at a time.

The use of the specialized memory 10 as an auxiliary random access serial memory is possible when the power-on initialization parameters use less than the full storage capacity of the device. If there is a concern that the parameter command and data blocks may inadvertently be overwritten while the specialized memory 10 is operated in the auxiliary mode, the finite state controller 20 can be adapted to reject user provided addresses which are less than a guard value. The guard value corresponds to the numerical address of a location beyond the last used parameter location.
Claims

1. An autoloading memory system for initializing peripheral devices comprising,
   a non-volatile memory having means for activation on power-up or reset of a computer system, the memory
   storing commands and data, the memory having a byte wide input register, a byte wide output register, and an array
   of memory elements grouped in addressable storage bytes holding commands and data for initialization of specific
   peripheral devices,
   state controller means connected to said memory for receiving commands and data therefrom, a clock
   circuit means connected to the state controller means for feeding clock pulses thereto, the state controller means
   having logic means operating on said clock pulses to produce pulses of selected length for said peripheral devices,
   output means connected to said state controller means for connection to specific peripheral devices for
   communicating said pulses of selected length thereto.

2. An autoloading memory system comprising:
   non-volatile memory means for storage of pre-loaded command and data blocks and having means
   for activation on power-up or reset of a computer; at least one data output line, DATA_OUT
   associated with the memory means;
   output register means connected to receive data from the memory means and to output the data at least one
   bit at a time on the DATA_OUT line;
   a plurality of CLOCK/EN lines;
finite state controller means connected to receive command blocks from the memory means and for interpreting command blocks to perform specified functions including the selection of a specific CLOCK/EN line for output of a clock signal used to transfer data on the DATA_OUT line;
oscillator means for generating a basic clock signal, having frequency divider means for changing the frequency of the basic clock signal and for supplying a resulting clock signal to the finite state controller means; and
logic means included within the finite state controller means for controlling the frequency divider means, said logic means connected to the plurality of CLOCK/EN output lines for transmitting the internally generated resulting clock signal, and for selectively transmitting an enable signal.

3. An autoloading memory system as set forth in claim 2 wherein the memory means outputs its stored contents on DATA_OUT lines in parallel format.

4. An autoloading memory system as set forth in claim 3 wherein the output register means receives data from the memory means in parallel format and outputs its register contents serially.

5. The autoloading memory system of claim 2 wherein said activation means comprises,
a RESET_IN input line for receiving a signal which initializes the finite state controller when at an active level and which initiates output data transfers upon a transition from the active to an inactive level;
a RESET_OUT output line for transmitting an active level while the RESET_IN signal is active or while the output data transfers are in progress.
6. An autoload memory system as set forth in claim 2 wherein the command blocks include:
   a field specifying one of the plurality of CLOCK/EN lines and a single bit indicating whether the
   clock signal is received or transmitted via the specified CLOCK/EN line;
   a field specifying an integer wherein the clock divider receives and divides the basic clock frequency by
   the integer to obtain a resulting clock signal; and
   a field used to specify a clock transition di-
   rection for the selected CLOCK/EN output signal, a first
   direction is defined by a signal transition from a first
   level to a second level while a second direction is de-
   fined as the converse.

7. An autoload memory system as set forth in claim 2 wherein the command blocks include:
   a bit decodable to cause an enable signal to be output on a predetermined CLOCK/EN line, the enable
   signal remaining at a predetermined level throughout an output data transfer; and
   a field which specifies one of the plurality of CLOCK/EN lines for output of the enable signal.

8. An autoload memory system as set forth in claim 3 wherein the command blocks include a field specifying a serial data transfer protocol to be used in making the output data transfer.

9. An autoload memory system as in claim 2 wherein the memory means is an electrically erasable memory, and the autoload memory further comprises serial input register means connected to receive serial information via a DATA_IN input line for transferring its register contents to the memory means for storage or to the finite state controller means for interpretation and execution as a command.
10. An autoload memory system as set forth in claim 3 wherein the command blocks include a bit which specifies operation of the memory means as a randomly addressable, serial read/write memory upon completion of the power-on data transfers, and wherein further the finite state controller means interprets the command block bit to modify controller behavior following completion of the power-on transfers to include (1) serially receiving a memory address, data block length, and read/write command via the DATA_IN input line and serial input register means, (2) interpreting and executing the read/write command using the received memory address as a starting address for the electrically erasable memory, (3) storing serially received write data during execution of a write command, alternatively (4) reading data from the memory for serial output via a DATA_OUT output line during execution of a read command, (5) modifying the memory address to effect a data transfer to/from a segment of memory, (6) comparing the length of the data transfer with the received command block field specifying data block length and terminating the data transfer upon completion, and (7) remaining in a state of readiness to receive and execute additional read/write commands.

11. An autoload memory system as in claim 10 further including a guard address in said command blocks and means for comparing a received read/write memory address and for not executing the read/write command if the received address is less than the guard address.
AMENDED CLAIMS
[received by the International Bureau on 14 December 1993 (14.12.93); original claims 1-11 replaced by amended claims 1-11 (5 pages)]

1. An autoload memory system for initializing peripheral devices comprising,

   a memory having means for transferring its stored contents on power-up or reset of a computer system, said memory storing both commands and data simultaneously, said memory having a byte wide input register, a byte wide output register, and an array of memory elements grouped in addressable storage bytes holding said commands and said data for initialization of specific peripheral devices,

   a finite state controller means connected to said memory for receiving said commands and said data therefrom, an oscillator means connected to said finite state controller means to produce pulses of selected length to facilitate the timing of a communication between said finite state controller and said peripheral devices,

   said finite state controller further comprising an output means connected to a plurality of said peripheral devices for communicating said pulses thereto.

2. An autoload memory system comprising:

   a non-volatile memory means for storage of both pre-loaded command blocks and pre-loaded data blocks simultaneously, said memory having means for transferring its stored contents on power-up or reset of a computer system;

   said memory means having at least one data output line, DATA_OUT associated with said memory means;

   an output register means connected to receive said data blocks from said memory means and to output said data at least one bit at a time on said DATA_OUT line;

   a plurality of CLOCK/EN lines;
a finite state controller means connected to said memory means for receiving said command blocks and said data blocks therefrom, said finite state controller having means for interpreting command blocks to perform specified functions including selecting a specific CLOCK/EN line for output of a clock signal used to transfer data on the DATA_OUT line;

an oscillator means for generating a basic clock signal, having a frequency divider means for changing a frequency of said basic clock signal and for supplying a resulting clock signal to said finite state controller means; and

said finite state controller means further comprising a means connected to the plurality of CLOCK/EN output lines for transmitting both said resulting clock signal and an enable signal.

3. An autoload memory system as set forth in claim 2 wherein said memory means outputs its stored contents on said DATA_OUT lines in parallel format.

4. An autoload memory system as set forth in claim 3 wherein said output register means receives data from said memory means in parallel format, said output register means outputs its register contents serially.
5. The autoloading memory system of claim 2 wherein said means for transferring its stored contents comprises,
   a RESET_IN input line for receiving a signal which initializes said finite state controller when at an active level and which initiates a plurality of output data transfers upon a transition from an active to an inactive level;
   a RESET_OUT output line for transmitting an active level while said RESET_IN signal is active or while said output data transfers are in progress.

6. An autoloading memory system as set forth in claim 2 wherein said command blocks include:
   a bit field specifying one of the plurality of CLOCK/EN lines and a single bit defining a two position switch allowing a clock signal to be initiated by either said peripheral device or said finite state controller as determined by the position of said switch;
   a field specifying an integer wherein a clock divider receives and divides a basic clock frequency by said integer to obtain a resulting clock signal; and
   a bit field used to specify a clock transition direction for a selected CLOCK/EN output signal, a first direction is defined by a signal transition from a first level to a second level and a second direction is defined by a signal transition from a second level to a first level.
7. An autoload memory system as set forth in claim 2 wherein the command blocks include:
   a bit field defining a switch which determines whether an enable signal is transmitted as determined by
   the position of the switch, said enable signal remaining at a predetermined level throughout an output data trans-
   fer; and
   a field which specifies one of the plurality of CLOCK/EN lines for output of said enable signal.

8. An autoload memory system as set forth in claim 3 wherein said command blocks include a field defining a
   serial data transfer protocol used by a selected peripheral device to facilitate an output data transfer.

9. An autoload memory system as in claim 2 wherein said memory means is an electrically erasable memory, and said
   autoload memory further comprises serial input register means connected to receive serial information via a
   DATA_IN input line for transferring its register contents to said memory means for storage or to said finite state
   controller means for interpretation and execution as a command.
10. An autoload memory system as set forth in claim 3 wherein said command blocks include a bit field defining a switch, said switch redefines a function of said memory means from a read only memory, to a serial read/write memory upon completion of a power-on data transfer, said switch redefines a function of said finite state controller means to include (1) serially receiving a memory address, a data block length, and a read/write command via a DATA_IN input line and a serial input register means, (2) interpreting and executing said read/write command using said memory address as a starting address for an electrically erasable memory, (3) storing serially received write data during execution of a write command, alternatively (4) reading data from said memory for serial output via a DATA_OUT output line during execution of a read command, (5) modifying said memory address to effect a data transfer to/from a segment of memory, (6) comparing a length of said data transfer with a command block field specifying a data block length and terminating said data transfer upon completion, and (7) remaining in a state of readiness to receive and execute additional read/write commands.

11. An autoload memory system as in claim 10 further including a guard address in said command blocks and means for comparing a received read/write memory address and for not executing said read/write command if said received address is less than the guard address.
STATEMENT UNDER ARTICLE 19

In response to the International Search Report mailed November 5, 1993, claims 1-11 were rewritten to point out that the memory means in Applicant's invention stores both commands and data simultaneously and that the finite state controller means is connected to receive said commands and said data directly from said memory means. This structure distinguishes Applicant's invention from the prior art cited in the International Search Report.

The memory means of Applicant's invention stores both commands and data simultaneously. None of the prior art cited discloses a memory means storing both data and commands simultaneously. Moreover, Applicant's controller receives all commands and data directly from the memory means and communicates the data to the peripheral devices. Herein lies the novelty of Applicant's invention. The finite state controller in Applicant's invention communicates directly with peripherals and initializes said devices without direction from a microprocessor. Kaiser et al., Frieder et al., Duwel et al., Shah et al. and Watters et al. each disclose an apparatus for communicating with the peripheral devices of a computer system. However, none of these patents disclose a structure similar to Applicant's whereby peripheral devices in a computer system are initialized without direction from a microprocessor. Rather, the controllers disclosed in the patents named above receive commands directly from a microprocessor, thereby facilitating microprocessor-peripheral communication. Said controllers do not communicate directly with the peripheral devices.
Fig. 1
<table>
<thead>
<tr>
<th>BIT 7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reset/M</td>
<td>More</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Fig. 2**

BYTE 7: Byte 7
BYTE 6: Byte 6
BYTE 5: Byte 5
BYTE 4: Byte 4
BYTE 3: Byte 3
BYTE 2: Byte 2
BYTE 1: Byte 1
BYTE 0: Byte 0

Data Block Start Address
Data Block Length (In Bits)
### A. CLASSIFICATION OF SUBJECT MATTER

<table>
<thead>
<tr>
<th>IPC(5)</th>
<th>GO6F 13/12</th>
</tr>
</thead>
<tbody>
<tr>
<td>US CL</td>
<td>395/400,425</td>
</tr>
</tbody>
</table>

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.     | USCL 395/400,425 |

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)

NOT APPLICABLE

### C. DOCUMENTS CONSIDERED TO BE RELEVANT

<table>
<thead>
<tr>
<th>Category</th>
<th>Citation of document, with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y</td>
<td>US, A, 4,222,491, (GEPPERT) 16 September 1980, see col. 2, lines 7-60.</td>
<td>1-11</td>
</tr>
<tr>
<td>Y</td>
<td>US, A, 4,410,694 (DANIELS et al.) 18 October 1983, see col. 2, line 31 to col. 3, line 35.</td>
<td>1-11</td>
</tr>
<tr>
<td>Y</td>
<td>US, A, 4,516,199 (FRIEDER et al.) 7 May 1985, see col. 1, line 29 to col. 3, line 5.</td>
<td>1-11</td>
</tr>
<tr>
<td>Y</td>
<td>US, A, 4,535,421 (DUJWEL et al.) 13 August 1985, see col. 2, line 46 to col. 5, line 26.</td>
<td>1-11</td>
</tr>
</tbody>
</table>

[X] Further documents are listed in the continuation of Box C.  
[ ] See patent family annex.

<table>
<thead>
<tr>
<th>*</th>
<th>Special categories of cited documents:</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;A&quot;</td>
<td>document defining the general state of the art which is not considered to be of particular relevance</td>
</tr>
<tr>
<td>&quot;E&quot;</td>
<td>earlier document published on or after the international filing date</td>
</tr>
<tr>
<td>&quot;L&quot;</td>
<td>document which may throw doubt on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)</td>
</tr>
<tr>
<td>&quot;O&quot;</td>
<td>document referring to an oral disclosure, use, exhibition or other means</td>
</tr>
<tr>
<td>&quot;P&quot;</td>
<td>document published prior to the international filing date but later than the priority date claimed</td>
</tr>
</tbody>
</table>

"T"  later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X"  document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y"  document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

Document member of the same patent family

Date of the actual completion of the international search: 10 SEPTEMBER 1993

Date of mailing of the international search report: NOV 05 1993

Name and mailing address of the ISA/US Commissioner of Patents and Trademarks

Box PCT  
Washington, D.C. 20221

Authorized officer

REBA I. ELMORE

Telephone No. (703) 305-3819

Facsimile No. NOT APPLICABLE

Form PCT/ISA/210 (second sheet)(July 1992)
<table>
<thead>
<tr>
<th>Category</th>
<th>Citation of document, with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y</td>
<td>US, A, 4,589,063 (SHAH et al.) 13 May 1986, see col. 2, lines 5-42.</td>
<td>1-11</td>
</tr>
<tr>
<td>X, Y</td>
<td>US, A, 4,942,606 (KAISER et al.) 17 July 1990, see col. 2, lines 20-46.</td>
<td>1-2, 7, 10-11, 3-6, 8-9</td>
</tr>
</tbody>
</table>