Description

The present invention relates to a memory access control circuitry in a data processing system according to the preamble of claim 1 and accomplishes information movement or data transfer within the data processing system between a memory unit and a peripheral controller with the processor being involved in assisting the information movement.

In present data processing systems a time critical data movement problem is frequently encountered when data are transferred from a buffer in a peripheral controller to an ordered array of sequential locations in a memory unit of the random access type (RAM).

In present data processing systems, the peripheral controller is usually equipped with a control line to indicate that data in its buffer register are ready to be transferred.

This control line is usually connected to an interrupt terminal of the Processor. Upon the occurrence of a data ready signal at the interrupt terminal, the Processor causes a plurality of operations to be performed. These operations include saving the state of the data processing system. Data is moved from the buffer of the peripheral controller to an internal register of the Processor. Next, an index register is loaded with a previously saved pointer, and then the data is moved from the internal register to a location in RAM pointed to by the index register. The quantity in the index register is incremented and saved for the next data movement, and finally the data processing system is restored to the state existing before the interrupt occurred.

If there is sufficient time to execute all of these steps before the peripheral controller has a data over-run, this method of data movement is simple and works very well. But, as data transfer speeds increase, especially in high density floppy, winchester or communications applications, the Processor will run out of time to perform all of the above operations before the next byte or word of data is available.

A data transfer control method is known from the Japanese published patent application 59-72532 (A) (see Patent Abstracts of Japan, Vol.8, No.179 (P-295) 1616, August 17, 1984) providing a direct mutual transfer of data between storage and a peripheral device controller. In the addressing scheme the address bits of highest significance are used to select a certain memory segment in a memory map for the data transfer from memory to peripheral or vice versa. Since in this kind of data transfer the processor should not be involved with the task of data buffering, the data bus to the processor is interrupted (opened) and only a data bus connection between memory and peripheral control unit is maintained. A "no-operation" instruction is described as not performing any operations but merely putting an address and read/write control signals on the bus. Key in this structure is a NOP-generator which in the memory map has address spaces that are data storage and image-related address spaces. Thus address spaces of the memory modules (RAM and ROM) are mirrored to the NOP address space. The circuitry selects both the NOP-generator and the RAM-module at a time and sets an address space of the RAM equal to the NOP. The peripheral device controller performs read/write operations of a byte every time the NOP selection signal is inverted and supplied to the controller.

Despite the fact that in this structure the data are transferred directly and mutually with an accelerated processing speed the circuitry is still complex in that it requires a special NOP-generator and switching operations to isolate the data bus from the processor.

Hence, there is still a need to provide a circuitry for performing the desired data movement at the high speed transfer rates required in the aforementioned applications which circuitry is less komplex than the circuitry which is provided for the above mentioned method.

Therefore, it is an object of the present invention to provide an improved circuitry to perform high-speed data move operations in a data processing apparatus.

This and other objects are achieved by a circuitry as characterized in claim 1 and will become more apparant when taken in conjunction with the following description and attached drawings showing a preferred embodiment of the invention.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 shows a block diagram of a data processing system in which the present invention can be employed;

Figure 2 shows a defined memory map of the preferred embodiment of the present invention;

Figure 3 shows a block diagram of physical memory of the preferred embodiment of the present invention;

Figure 4, which is comprised of Figures 4A and 4B, shows input and output connections to a multiplexer for switching control signals thereby activating the appropriate circuitry to accomplish the desired operation;

Figure 5 shows a logic block diagram of enable logic; and

Figure 6 shows the steps in an interrupt routine performed by a microprocessor of the data processing system of the preferred embodiment

of the present invention to initiate the data move operation.

DETAILED DESCRIPTION

Referring to Figure 1, there is shown a block diagram of a data processing system in which the present invention can be employed. A microprocessor (up) 10 is operatively connected to a bus 20, having data and address lines. A memory 30 and a peripheral controller 40 are also operatively connected to the bus 20. A peripheral device 45 is operatively connected to the peripheral controller 40. Although, the preferred embodiment of the present invention specifically shows a disk controller and a disk device for the peripheral controller 40 and the peripheral 45, respectively, it will be understood by those skilled in the art that the present invention is applicable to a wide variety of peripheral controllers and peripheral devices. The peripheral controller 40 includes a buffer (or peripheral register) 41 which temporarily stores data from the bus 20 or from the peripheral 45. The discussion hereinunder focuses on transfers between the peripheral controller 40 and the data processing system, i.e., the memory 30 and the microprocessor 10, to specifically cover the peripheral controller 40 - data processing system interface rather than the peripheral controller 40 - peripheral device 45 interface

Referring to Figure 2, there is shown a memory map of the preferred embodiment of the present invention. The memory 30 (also referred to herein as memory modules) of the data processing system of the preferred embodiment of the present invention is mapped to four 16K segments, the four segments forming a total memory space. Sixteen address lines A_{15} - A_0 where A_0 is the least significant bit, are utilized in bus 20. The upper 16K segment (locations C000 - FFFF hexadecimal, all memory locations given herein are in hexadecimal) is mapped to include a 4K ROM from locations F000 - FFFF, and 8K RAM from locations D000 to EFFF, and an I/O register at location C000. For this defined upper 16K segment of memory, address bits A_{15} and A_{14} are both a logic one. This upper 16K segment of memory is also referred to herein as the first 16K segment of memory.

The next, or second, 16K segment of the defined memory map from location 8000 to BFFF includes a 4K ROM image, an 8K RAM image, and an I/O register image which mirrors (or duplicates) the first 16K segment. In the second 16K segment of defined memory, A_{15} is a logic one and A_{14} is a logic zero. The third 16K segment of the defined memory map from locations 4000 to 7FFF also includes a 4K ROM image, an 8K RAM image and an I/O register image which mirrors the upper 16K segment. In the third 16K segment of defined memory, A_{15} is a logic zero and A_{14} is a logic one. The fourth (or lower) 16K segment of the defined memory map from locations 0000 to 3FFF is not utilized in the preferred embodiment of the present invention. The second 16K segment of memory is addressed when a data movement is to be performed from the peripheral register 41 to RAM memory, and the third 16K segment of memory is addressed when a data move operation is to be performed from memory to the peripheral register 41. This operation will be described in further detail hereinunder. The addresses 0000 - FFFF are the memory space addresses within the total memory space.

Referring to Figure 3, there is shown a block diagram of the first segment of physical memory 30 of the preferred embodiment of the present invention. The first segment of memory 30 of the preferred embodiment of the present invention includes a 4K ROM memory module 30 - 3 having real addresses 000 - FFF, a first 4K RAM1 memory module 30 - 1 having real addresses 000 - FFF, and a second 4K RAM2 memory module 30 - 2 having real addresses 000 - FFF and further having pseudo addresses 1000 - 1FFF. Address lines 21, A_{11} - A_0 of bus 20, are operatively connected to the corresponding 12 terminals of the ROM 30 - 3, RAM1 30 - 1 and RAM2 30 - 2. These 12 address lines which carry the corresponding address signals from microprocessor 10 are sufficient to address any location within a 4K space. A_{15}, and A_{14} are the upper bits utilized to address one of the 16K segments, and A_{13}, A_{12} are utilized to address a module within the selected 16K segment. In order to avoid memory accesses to a location within the multiple 4K modules, an enable signal is coupled from control logic 50. The control logic 50 generates the proper enable signal by decoding address information contained on address lines A_{15} - A_{12}, the control logic 50 to be described in detail hereinunder. An enable signal from control logic 50 is also coupled to an enable terminal, E, of peripheral register 41. Data lines 22 of bus 20 are connected to the corresponding data terminals, D, of the ROM and RAM memory modules and the data terminals of the buffer module in which peripheral register 41 is located. It will be understood that if more than a single peripheral register 41 is utilized in the peripheral controller 40, address lines 21 can be coupled to corresponding address terminals of the peripheral registers (not shown) in order to address a single peripheral register.

Referring to Figure 4A and 4B, which together make up Figure 4, there is shown a block diagram of a multiplexer, MUX, 51 which switches the con-

trol signals from the control logic 50 to the appropriate memory module 30 and/or peripheral register 41, thereby achieving the desired operation. The microprocessor read strobe signal is connected to input terminal A0, C0, B1, A2, B2, and C3. Input terminal B0, C1, C2, and B3 are tied to a logic zero which is effectively a disabling signal. The microprocessor write strobe signal is connected to input terminals A1 and A3. The output of MUX 51, outputs 0 - 3 are connected to the RAM/ROM read strobe terminal of the memory modules 30, RAM write strobe terminal of RAM 30-1, 30-2, peripheral read strobe terminal, and peripheral write strobe terminal, respectively. Address signals A_{15} and A_{14} are utilized to select between input terminals A, B, and C, in accordance with the table of Figure 4B. In a normal operation by the microprocessor 10 in which the microprocessor is accessing the memory modules 30 and/or the peripheral register 41, the microprocessor is addressing memory space in which A_{15} and A_{14} are a logic one. Hence, the A input terminals of MUX 51 are selected. In this case the microprocessor read strobe signal is coupled to the RAM/ROM read strobe terminal and is also coupled to the peripheral read strobe terminal via output terminal 2. Hence, normal reads are made from the ROM 30-3, RAM 30-1, 30-2, or the peripheral register 41 by the microprocessor 10. When write operations are performed, the microprocessor 10 again addresses address space is which A_{15} and A_{14} are both a logic one and the write strobe signal is coupled to the RAM write strobe terminal or the peripheral write strobe terminal as a function of the particular address, thereby performing normal write operations to the RAM 30-1, 30-2 or the peripheral register 41.

In an I/O operation, the microprocessor 10 also acts as a controller. When the peripheral register 41 has data to be transferred, assuming a read operation of the peripheral device 45, the data ready signal is generated by the peripheral controller 40, thereby interrupting the microprocessor 10. The microprocessor 10 then causes the data stored in peripheral register 41 to be directly inputted into memory bypassing the operation of reading the data into an internal register of the microprocessor and then outputting the data into memory 30. The microprocessor performs a "dummy read" of the RAM address in which the data is to be stored. The address space addressed is the second 16K of the defined memory map in which A_{15} and A_{14} are a logic one and a logic zero, respectively. This causes the B input terminals of MUX51 to be selected. The "dummy read" from the microprocessor causes the read strobe signal to be coupled via input B1 to the RAM write strobe terminal. The microprocessor "dummy read" also causes the read strobe signal to be coupled via input B2 to the peripheral read strobe terminal, the peripheral write strobe being disabled by the logic zero input via input B3. Hence, the peripheral operation is a read, or output of the data, the data being placed on the data lines 22 and the RAM accepts the data that is present on the data lines 22 and is written into the RAM at the address specified on the address lines, the address being the address of the "dummy read" operation. Hence, data is transferred directly from the peripheral register 41 into the RAM and bypasses the transfer of the data into the microprocessor 10. RAM1 30-1 and RAM2 30-2 both have addresses corresponding to the address on the address line but only a single RAM, i.e., either RAM1 or RAM2, is selected as a function of the enable signal. The generation of the proper enable signal to select the correct RAM, i.e., RAM1 or RAM2, will be discussed hereinunder. When a write operation from memory to the peripheral register 41 is to be performed, the microprocessor performs a "dummy read" operation of the address in memory to be read, and addresses the third 16K area in which A_{15} is a logic zero and A_{14} is a logic one, this area having been defined as the data move from memory to the peripheral, this having been described in conjunction with Figure 2 above. In this memory space input terminals C are selected. In this case the microprocessor "dummy read" causes the microprocessor read strobe signal to be coupled to the RAM read terminal and also causes the microprocessor read strobe signal to be coupled to the peripheral write terminal. Hence, the data is placed on the data lines 22 from the RAM and the data present on the data lines 22 is written into the peripheral register 41. Again, the enable signals coupled to the memory modules 30 cause data from a single RAM 30-1, 30-2, or ROM 30-3 to place data on the data lines 22. This operation again bypasses the microprocessor having to read the data into an internal register and then outputting the data to the peripheral controller 41. The logic zero connected to the input terminals assures the control signal is zero, disabling the particular function. The D input terminals of MUX51 are not utilized in the preferred embodiment of the present invention.

Referring to Figure 5, there is shown a block diagram of enable logic 52 of the control logic 50. The enable logic 52 generates an enable signal for each of the memory modules 30-1, 30-2, 30-3, and for the peripheral register 41, the enable signal being coupled to the corresponding enable terminal, E, of each module. When the microprocessor 10 is accessing the memory modules 30 or the peripheral register 41, the first 16K address space is utilized and A_{15} and A_{14} are both a logic one,

enabling a first (2:4) decoder 53. The decoder decodes the signals associated with address lines A_{13} and A_{12}. These two signals identify the particular module being addressed. Hence when A_{13} is a logic zero and A_{12} is a logic zero, the peripheral register 41 is being addressed by the microprocessor, and the enable signal is coupled to the peripheral register 41 via an OR gate 57. When A_{13} and A_{12} are both a logic one, the ROM module 30-3 is being addressed and the enable signal is directly coupled to the ROM. When the RAM is being addressed by the microprocessor 10, for address locations between D000 and DFFF, A_{13} is a logic zero and A_{12} is a logic one, thereby generating an enable signal which enables OR gate 55 allowing accesses to RAM1 30-1. Similarly, when the microprocessor is addressing memory locations E000 through EFFF of RAM the first decoder 53 decodes A_{13} as a logic one and A_{12} as a logic zero generating an enable signal which enables OR gate 56 thereby allowing accesses to RAM2 30-2. In the movement of data between the peripheral register 41 and the RAMS 30-1, 30-2, a second (2:4) decoder 54 is enabled which enables either RAM1 30-1 or RAM2 30-2 as a function of the address information signals on address lines A_{13} and A_{12}, the corresponding enable signal being coupled to the OR gate 57 for enabling the peripheral register 41 and the corresponding RAM1 or RAM2. A read or write operation of the peripheral register 41 or the RAM 30-1, 30-2 is a function of the control signals from the multiplexer 51.

Referring to Figure 6, there is shown the operations performed by the interrupt routine of the microprocessor. As discussed before, when the peripheral register 41 is full and contains data ready to be transferred, the data ready signal is generated and is coupled to the interrupt terminal of the microprocessor 10. The microprocessor performs the operations of the interrupt routine of Figure 6. Namely, a first overhead operation is performed which requires saving the machine state and initializing a pointer which defines the memory location in which the data is to be stored. A "dummy read" is then performed to the location in which a data access is to be made, the location being defined in address bits A_{13} - A_0. The address space addressed by bits A_{13} - A_0 is sufficient to address any location within the defined 16K. A_{15} and A_{14} are included in the address signals placed on the address lines 21 which define which operation is to be performed, the type of operation being equated to one of four 16K areas as defined by the memory map as discussed above. Once the "dummy read" operation is performed, the second overhead operation is executed which essentially restores the pointer and restores the machine state. Then the interrupt routine is exited and the microprocessor continues to perform or execute its application program or task which was underway prior to the interrupt. In the preferred embodiment of the present invention, a Motorola 68000 microprocessor is utilized, and the steps required to perform the various overhead operation are minimized. This reduces the amount of overhead, thereby reducing the amount of time within the interrupt routine. This reduction in overhead allows the microprocessor to perform its task and also act as a direct memory access controller and not encounter an over-run condition. (An over-run condition occurs when the next piece or quantum of data is available in the peripheral register 41 before the microprocessor has had a chance to move the previous piece of data.) It will be understood by those skilled in the art that outputting data from memory 30 to the peripheral register 41 a similar operation is performed. Namely, a data request signal from the peripheral register is coupled to the microprocessor 10 which results in an interrupt to the microprocessor 10. A "dummy read" in the interrupt routine is then performed from the third 16K memory space in which A_{15} is a logic zero and A_{14} is a logic one.

Claims

1. Memory access control circuitry in a data processing system which includes:
   - a memory (30) comprising a plurality of addressable locations (16 kBytes) for storing data in different memory modules (RAMs 30-1, 30-2, ROM 30-3),
   - peripheral units (45) coupled to a peripheral controller (40) comprising a peripheral register (41) which subsequent to the issuance of a data request signal receives data read from said memory (30) and which subsequent to the issuance of a data ready signal transfers data for writing into said memory (30), said data request and data ready signals being generated by said peripheral controller (40) whenever occasion arises,
   - a processor (10) capable of responding to the said data request and data ready signals whenever they are received from said peripheral controller (40),
   - a bus (20) coupling together said memory, said peripheral controller (40) and said processor (10) for direct data transfer between the said units (10, 30, 40), said bus comprising
     - a data bus (22) for transferring the data between the said units (10, 30, 40), and

enabling a first (2:4) decoder 53. The decoder decodes the signals associated with address lines A_{13} and A_{12}. These two signals identify the particular module being addressed. Hence when A_{13} is a logic zero and A_{12} is a logic zero, the peripheral register 41 is being addressed by the microprocessor, and the enable signal is coupled to the peripheral register 41 via an OR gate 57. When A_{13} and A_{12} are both a logic one, the ROM module 30-3 is being addressed and the enable signal is directly coupled to the ROM. When the RAM is being addressed by the microprocessor 10, for address locations between D000 and DFFF, A_{13} is a logic zero and A_{12} is a logic one, thereby generating an enable signal which enables OR gate 55 allowing accesses to RAM1 30-1. Similarly, when the microprocessor is addressing memory locations E000 through EFFF of RAM the first decoder 53 decodes A_{13} as a logic one and A_{12} as a logic zero generating an enable signal which enables OR gate 56 thereby allowing accesses to RAM2 30-2. In the movement of data between the peripheral register 41 and the RAMS 30-1, 30-2, a second (2:4) decoder 54 is enabled which enables either RAM1 30-1 or RAM2 30-2 as a function of the address information signals on address lines A_{13} and A_{12}, the corresponding enable signal being coupled to the OR gate 57 for enabling the peripheral register 41 and the corresponding RAM1 or RAM2. A read or write operation of the peripheral register 41 or the RAM 30-1, 30-2 is a function of the control signals from the multiplexer 51.

Referring to Figure 6, there is shown the operations performed by the interrupt routine of the microprocessor. As discussed before, when the peripheral register 41 is full and contains data ready to be transferred, the data ready signal is generated and is coupled to the interrupt terminal of the microprocessor 10. The microprocessor performs the operations of the interrupt routine of Figure 6. Namely, a first overhead operation is performed which requires saving the machine state and initializing a pointer which defines the memory location in which the data is to be stored. A "dummy read" is then performed to the location in which a data access is to be made, the location being defined in address bits A_{13} - A_0. The address space addressed by bits A_{13} - A_0 is sufficient to address any location within the defined 16K. A_{15} and A_{14} are included in the address signals placed on the address lines 21 which define which operation is to be performed, the type of operation being equated to one of four 16K areas as defined by the memory map as discussed above. Once the "dummy read" operation is performed, the second overhead operation is executed which essentially restores the pointer and restores the machine state. Then the interrupt routine is exited and the microprocessor continues to perform or execute its application program or task which was underway prior to the interrupt. In the preferred embodiment of the present invention, a Motorola 68000 microprocessor is utilized, and the steps required to perform the various overhead operation are minimized. This reduces the amount of overhead, thereby reducing the amount of time within the interrupt routine. This reduction in overhead allows the microprocessor to perform its task and also act as a direct memory access controller and not encounter an over-run condition. (An over-run condition occurs when the next piece or quantum of data is available in the peripheral register 41 before the microprocessor has had a chance to move the previous piece of data.) It will be understood by those skilled in the art that outputting data from memory 30 to the peripheral register 41 a similar operation is performed. Namely, a data request signal from the peripheral register is coupled to the microprocessor 10 which results in an interrupt to the microprocessor 10. A "dummy read" in the interrupt routine is then performed from the third 16K memory space in which A_{15} is a logic zero and A_{14} is a logic one.

Claims

1. Memory access control circuitry in a data processing system which includes:
   - a memory (30) comprising a plurality of addressable locations (16 kBytes) for storing data in different memory modules (RAMs 30-1, 30-2, ROM 30-3),
   - peripheral units (45) coupled to a peripheral controller (40) comprising a peripheral register (41) which subsequent to the issuance of a data request signal receives data read from said memory (30) and which subsequent to the issuance of a data ready signal transfers data for writing into said memory (30), said data request and data ready signals being generated by said peripheral controller (40) whenever occasion arises,
   - a processor (10) capable of responding to the said data request and data ready signals whenever they are received from said peripheral controller (40),
   - a bus (20) coupling together said memory, said peripheral controller (40) and said processor (10) for direct data transfer between the said units (10, 30, 40), said bus comprising
     - a data bus (22) for transferring the data between the said units (10, 30, 40), and

* C L M S *

- an address bus (21) for the transfer of address bits,
  - a first subgroup (14) of the total available (16) address bits representing the address access bits (A_{13} - A_0) sufficient in number to access all of said memory locations (16 kBytes), and
  - a second subgroup (2) of the total available (16) address bits representing the address control bits (A_{15}, A_{14}) for enabling direct transfer of data over said data bus (22),
- a control logic circuit (50) to enable the access to the data in said memory (30) and/or in said peripheral register (41), said control logic circuit comprising
  - a multiplexer (51) steering, under control of said address control bits (A_{15}, A_{14}), read/write signals (µP read and µP write strobe), supplied to its inputs (A0-3, B0-3, C0-3), to the units involved in said data transfer to effect reading and writing of said data from or into the accessed storage locations of said units involved and transferring said data over the data bus (22), and
  - address decoder means (52) responsive (at E of 53/54) to different configurations of said address control bits (A_{15}, A_{14}) to determine by this configuration the mode of data transfer between the units involved and to deliver, at output terminals (55, 56, 57) of the said address decoder means, output signals fed to the enable terminals (E of 30/41) of the memory modules and of the peripheral register to enable these to be operational and respond to the said read/write signals (RAM/ROM read, RAM write, peripheral read, peripheral write) delivered by said multiplexer (51), whereby
    - a first of said configurations of the said address control bits (A_{15} = "1"; A_{14} = "1") enables the processor (10) to access in a normal data transfer mode the memory (30) or the peripheral register (41) for reading data therefrom or supplying data thereto,
    - a second of said configurations of said address control bits (A_{15} = "1"; A_{14} = "0") effects direct memory access in a first direct access data transfer mode (MOVE PER - - -> MEM) whereby data are transferred from said peripheral register (41) to said memory (30), and
    - a third of said configurations of said address control bits (A_{15} = "0"; A_{14} = "1") effects direct memory access in a second direct access data transfer mode (MOVE MEM - - -> PER) whereby data are transferred from said memory (30) to said peripheral register (41),

  and whereby
  - whenever the first configuration of said address control bits (A_{15} = "1"; A_{14} = "1") is valid, only one decoder means output terminal delivers an enable signal to the units (30-1 or 30-2 or 30-3 or 41) involved, and
  - whenever the second or third configuration of said address control bits (A_{15}, A_{14} = "1 0" or "0 1") is valid, two decoder means output terminals deliver enable signals to the said units coupled thereto to effect direct transfer of data over said data bus (22) between the two units so enabled,

  and whereby
  - data transfer in the two direct access transfer modes is performed without resorting to an internal register of said processor (10),

characterized in that

the processor (10) in response to the receipt of the said data request or data ready signals performs an interrupt routine (Fig. 6) which includes a read command (Overhead 2: "dummy read") by which

- a first set of address access bits (A_{11} - A_0) is applied to said address bus (21) to identify the storage locations in said memory (30) for data access, and a second set of at least two address access bits (A_{13}, A_{12}) is applied to the address decoder means (52) of the control logic circuit (50) to enable the access to the data in said memory (30) and/or in said peripheral register (41),
- the said address control bits (A_{15}, A_{14}) are applied to said address decoder means (52) of the control logic circuit (50) for the determination of the mode of data transfer between the units involved, and
- read and write strobe signals (µP read and µP write) are generated which are supplied to the inputs (A0-3, B0-3,

* C L M S *

C0 – 3) of the multiplexer (51) for their steering, under the control of the address control bits, to the units involved in said data transfer,

and further characterized in that

the said decoder means (52), in response to the second and third configurations of said address control bits (A_{15}, A_{14} = "1 0" or "0 1"), steers the said at least two address access bits (A_{13}, A_{12}), after their decoding (2:4), to the said decoder means output terminals in such different ways that

- whenever the second configuration of said address control bits (A_{15}, A_{14} = "1 0") is valid, two decoder means output terminals deliver enable signals to the respective units coupled thereto, and
- whenever the third configuration (A_{15}, A_{14} = "0 1") is valid, another pair of decoder means output terminals deliver enable signals to the respective units coupled thereto

to effect the direct transfer of data over said data bus (22) between the two units without disrupting the connective integrity of said data bus (22) with the processor (10).

2. Memory access control circuitry according to claim 1, characterized in that the output terminals (Ø – 3) of said multiplexer (51) are coupled to said memory modules (RAM1, RAM2, ROM; 30 – 1, 30 – 2, 30 – 3) and to said peripheral register (41) for transferring to them the read/write strobe control signals, whereby
   - a first output (Ø) delivers a first (RAM/ROM) read strobe signal to those memory modules which are capable of delivering stored data;
   - a second output (1) delivers a first (RAM) write strobe signal to those memory modules which are capable of accepting data for storage;
   - a third output (2) delivers a second (peripheral) read strobe signal to the peripheral register (41) to initiate the reading of data therefrom; and
   - a fourth output (3) delivers a second (peripheral) write strobe signal to the peripheral register (41) to initiate the writing of data therein.

3. Memory access control circuitry according to claim 1, characterized in that said decoder means (52) comprise
   - a first decoder (53) whose output terminals (00, 11, 10, 01) are respectively coupled to each of said memory modules (30 – 1, 30 – 2, 30 – 3) and said peripheral register(41), and
   - a second decoder (54) whose output terminals (10, 01) are respectively coupled always to the peripheral register (41) and to a different memory module (01 of 54 via 55 to 30 – 1 and via 57 to 41; 10 of 54 via 56 to 30 – 2 and via 57 to 41).

4. Memory access control circuitry according to claim 2, characterized in that the multiplexer (51) provides output signal combinations as follows:
   - the first read strobe signal and the second read strobe signal in response to a read control signal (µP read) and
     to the said one configuration (A14/A15 = 11)
     of said address control bits;
   - the first write strobe signal and the second write strobe signal in response to a write control signal (µP write) and
     to the said one configuration (A14/A15 = 11)
     of said address control bits;
   - the second read strobe signal and the first write strobe signal in response to a read control signal (µP read) and
     to the said second configuration (A14/A15 = 0 1)
     of said address control bits; and
   - the second write strobe signal and the first read strobe signal in response to a read control signal (µP read) and
     to the said third configuration (A14/A15 = 1 0)
     of said address control bits.

5. Memory access control circuitry according to any one of the preceding claims, characterized in that the peripheral controller (40) produces data ready control signals when the data is to be transferred between the peripheral register (41) and the memory (30), whereupon the processor (10) performs an interrupt routine and together with the read control signal (µP read) produces the required address access and address control signals (AØ – A15) to access the respective memory module to effect the data transfer with respect to the peripheral register (41).

6. Memory access control circuitry according to any one of the preceding claims, characterized in that the desired storage location in a memory module which is activated through a select control signal at its enable terminal is addressed through the address access bit (A_{11} –

* C L M S *

A_0) signals sent over the address bus lines (21) whereby so many address access bits are provided in the said first set of address bits as are required to address the memory module selected (12 bits $A_{11} - A_0$ for a memory space of 4 kBytes).

7. Memory access control circuitry according to claim 6, characterized in that the memory (30) comprises at least two random access modules (RAM 30-1, 30-2) and at least one read-only memory module (ROM 30-3), each of these memory modules having a plurality (4 kBytes) of addressable locations which are addressed by a plurality of address access bits ($A_{11} - A_0$) commensurate in their number to unequivocally address the total address space available in each memory module.

Patentansprüche

1. Speicherzugriffssteuerungs-Schalttechnik in einem Datenverarbeitungssystem, die aufweist:
   - einen Speicher (30) der eine Vielzahl von adressierbaren Stellen (16 kBytes) zum Speichern von Daten in unterschiedlichen Speichermodulen (RAMs 30-1, 30-2, ROM 30-3) aufweist,
   - periphere Einheiten (45), die an eine Periphersteuerungseinrichtung (40) gekoppelt sind, die ein Peripherregister (41) aufweist, das nachfolgend zur Ausgabe eines Datenanforderungssignals Daten empfängt, die von dem Speicher (30) ausgelesen sind und das nachfolgend zu der Ausgabe eines Daten-Bereitsignals Daten zum Schreiben in den Speicher (30) überträgt, wobei das Datenanforderungssignal und das Daten-Bereitsignal durch die Periphersteuerungseinrichtung (40) erzeugt wird, wann immer sich eine Gelegenheit ergibt,
   - ein Prozessor (10) der in der Lage ist, auf das Datenanforderungssignal und das Daten-Bereitsignal zu antworten, wann immer sie von der Periphersteuerungseinrichtung (40) empfangen werden,
   - ein Bus (20) der den Speicher, die Periphersteuerungseinrichtung (40) und den Prozessor (10) zur direkten Datenübertragung zwischen den Einheiten (10, 30, 40) zusammenkoppelt, wobei der Bus aufweist:
     - einen Datenbus (22) zum Übertragen der Daten zwischen den Einheiten (10, 30, 40), und
     - einen Adreßbus (21) zur Übertragung von Adreßbits, wobei
       - eine erste Untergruppe (14) der insgesamt verfügbaren (16) Adreßbits die Adreßzugriffsbits ($A_{13} - A_0$) darstellen, die in der Anzahl ausreichen, auf alle Speicherstellen (16 kBytes) zuzugreifen und
       - eine zweite Untergruppe (2) der insgesamt verfügbaren (16) Adreßbits die Adreßsteuerbits (A_{15}, A_{14}) darstellen zum Ermöglichen direkter Datenübertragung über den Datenbus (22),
   - eine Steuerlogikschaltung (50), um den Zugriff auf die Daten in dem Speicher (30) und/oder in dem Peripherregister (41) zu ermöglichen, wobei die Steuerlogikschaltung aufweist:
     - einen Multiplexer (51) der unter der Steuerung der Adreßsteuerbits (A_{15}, A_{14}), der Lese-/Schreibsignale (μP-Leseimpuls und μP-Schreibimpuls), die an seine Eingänge (A0-3, B0-3, C0-3) geliefert sind, an die an der Datenübertragung beteiligten Einheiten lenkt, um Lesen oder Schreiben der Daten von oder in die zugegriffenen Speicherstellen der beteiligen Einheiten zu bewirken und die Daten über den Datenbus (22) zu übertragen, und
     - eine Adreßdekodiereinrichtung (52), die (bei E von 53/54) auf unterschiedliche Konfigurationen der Adreßsteuerbits (A_{15}, A_{14}) antwortet, um durch diese Konfiguration den Modus der Datenübertragung zwischen den beteiligten Einheiten zu bestimmen und an den Ausgangsanschlüssen (55, 56, 57) der Adreßdekodiereinrichtung Ausgabesignale zu liefern, die an die Freigabeanschlüsse (E von 30/41) der Speichermodule und des Peripherregisters geliefert sind, um jene zu befähigen, betriebsbereit zu sein und auf die Lese/Schreibsignale (RAM/ROM-Lese, RAM-Schreibe, Peripher-Schreibe) zu antworten, die von dem Multiplexer (51) geliefert sind, wobei
       - eine Erste der Konfigurationen der Adreßsteuerbits (A_{15} = "1"; A_{14} = "1") den Prozessor (10) befähigt, auf den Speicher (30) oder das Peripherregister (41) in einem normalen Datenübertragungsmodus zuzugreifen, um daraus Daten zu Lesen oder dorthin Daten zu liefern,

* C L M S *

- eine Zweite der Konfigurationen der Adreßsteuerbits (A_{15} = "1"; A_{14} = "0") direkten Speicherzugriff in einem ersten Direktzugriffs-Datenübertragungsmodus (MOVE PER ---> MEM) bewirkt, wobei Daten von dem Peripherregister (41) an den Speicher (30) übertragen werden, und
- eine Dritte der Konfigurationen der Adreßsteuerbits (A_{15} = "0"; A_{14} = "1") direkten Speicherzugriff in einem zweiten Direktzugriffs-Datenübertragungsmodus (MOVE MEM ---> PER) bewirkt, wodurch Daten von dem Speicher (30) zu dem Peripherregister (41) übertragen werden,

und wobei

- wann immer die erste Konfiguration der Adreßsteuerbits (A_{15} = "1"; A_{14} = "1") gültig ist, nur ein Dekodiereinrichtungs-Ausgangsanschluß ein Freigabesignal an die beteiligten Einheiten (30-1 oder 30-2 oder 30-3 oder 41) liefert, und
- wann immer die zweite oder dritte Konfiguration der Adreßsteuerbits (A_{15}, A_{14} = "1 0" oder "0 1") gültig ist, zwei Dekodiereinrichtungs-Ausgangsanschlüsse Freigabesignale an die daran gekoppelten Einheiten liefert, um direkte Übertragung von Daten über den Datenbus (22) zwischen den zwei derart freigegebenen Einheiten zu bewirken,

und wodurch

- Datenübertragung in den zwei Direktzugriffs-Übertragungsmodi ausgeführt wird, ohne auf ein internes Register des Prozessors (10) zurückzugreifen,

dadurch gekennzeichnet, daß

der Prozessor (10) als Antwort auf den Empfang des Datenanforderungsignals oder des Daten-Bereitsignals eine Interrupt-Routine (Fig. 6) ausführt, welche einen Lesebefehl (Overhead 2: "Dummy Read") einschließt, wodurch

- ein erster Satz von Adreßzugriffsbits (A_{11} - A_0) an den Adreßbus (21) angelegt wird, um die Speicherstellen in dem Speicher (30) für Datenzugriff zu identifizieren und ein zweiter Satz von wenigstens zwei Adreßzugriffsbits (A_{13}, A_{12}) an die Adreßdekodiereinrichtung (52) der Steuerlogikschaltung (50) angelegt wird, um den Zugriff auf die Daten in dem Speicher (30) und oder in dem Peripherregister (41) zu ermöglichen,
- die Adreßsteuerbits (A_{15}, A_{14}) an die Adreßdekodiereinrichtung (52) der Steuerlogikschaltung (50) angelegt werden zur Bestimmung des Modus der Datenübertragung zwischen den beteiligten Einheiten, und
- Lese- und Schreib-Impulssignale (μP-Lese und μP-Schreibe) erzeugt werden, die an die Eingänge (A0-3, B0-3, C0-3) des Multiplexers (51) zu deren Lenkung geliefert werden unter der Steuerung der Adreßsteuerbits an die an der Datenübertragung beteiligten Einheiten,

und weiterhin dadurch gekennzeichnet, daß die Dekodiereinrichtung (52) als Antwort auf die zweite und dritte Konfiguration der Adreßsteuerbits (A_{15}, A_{14} = "1 0" oder "0 1") die mindestens zwei Adreßzugriffsbits (A_{13}, A_{12}) nach deren Dekodierung (2:4) an die Dekodiereinrichtungs-Ausgangsanschlüsse in solch unterschiedlichen Weisen lenkt, daß

- wann immer die zweite Konfiguration der Adreßsteuerbits (A_{15}, A_{14} = "1 0") gültig ist, zwei Dekodiereinrichtungs-Ausgangsanschlüsse Freigabesignale an die jeweils daran gekoppelten Einheiten liefert, und
- wann immer die dritte Konfiguration (A_{15}, A_{14} = "0 1") gültig ist, ein weiteres Paar von Dekodiereinrichtungs-Ausgangsanschlüssen Freigabesignale an die jeweils daran gekoppelten Einheiten liefert,

um die direkte Übertragung von Daten über den Datenbus (22) zwischen den zwei Einheiten zu bewirken, ohne die verbindungsmäßige Integrität des Datenbusses (22) mit dem Prozessor (10) zu stören.

2. Speicherzugriffssteuer-Schalttechnik gemäß Anspruch 1, dadurch gekennzeichnet, daß die Ausgangsanschlüsse (0-3) des Multiplexers (51) an die Speichermodule (RAM 1, RAM 2, ROM; 30-1, 30-2, 30-3) und an das Peripherregister (41) gekoppelt sind, um die Lese/Schreibimpulssteuersignale an sie zu übertragen, wobei

- ein erster Ausgang (0) ein erstes (RAM-ROM) Leseimpulssignal an jene Speichermodule liefert, die in der Lage sind, gespeicherte Daten zu liefern;
- ein zweiter Ausgang (1) ein erstes (RAM) Leseimpulssignal an jene Speichermodule liefert, die in der Lage sind, Daten zum Speichern anzunehmen;

* C L M S *

- ein dritter Ausgang (2) ein zweites (Peripher-) Leseimpulssignal an das Peripherregister (41) liefert, um das Lesen von Daten daraus einzuleiten; und
- ein vierter Ausgang (3) ein zweites (Peripher-) Schreibimpulssignal an das Peripherregister (41) liefert, um das Schreiben von Daten darin einzuleiten.

3. Speicherzugriffssteuer-Schalttechnik gemäß Anspruch 1, dadurch gekennzeichnet, daß die Dekodiereinrichtung (52) aufweist:
   - einen ersten Dekodierer (53), dessen Ausgangsanschlüsse (00, 11, 10, 01) jeweils mit jedem der Speichermodule (30-1, 30-2, 30-3) und dem Peripherregister (41) gekoppelt sind, und
   - einen zweiten Dekodierer (54), dessen Ausgangsanschlüsse (10, 01) jeweils immer mit dem Peripherregister (41) und mit einem unterschiedlichen Speichermodul (01 von 54 über 55 an 30-1 und über 57 an 41; 10 von 54 über 56 an 30-2 und über 57 an 41) gekoppelt sind.

4. Speicherzugriffssteuer-Schalttechnik gemäß Anspruch 2, dadurch gekennzeichnet, daß der Multiplexer (51) Ausgangssignalkombinationen wie folgt bereitstellt:
   - das erste Leseimpulssignal und das zweite Leseimpulssignal als Antwort auf ein Lesesteuersignal (µP-Lese) und auf die eine Konfiguration (A14/A15 = 1 1) der Adreßsteuerbits;
   - das erste Schreibimpulssignal und das zweite Schreibimpulssignal als Antwort auf ein Schreibsteuersignal (µP-Schreibe) und auf die eine Konfiguration (A14/A15 = 1 1) der Adreßsteuerbits;
   - das zweite Leseimpulssignal und das erste Schreibimpulssignal als Antwort auf ein Schreibsteuersignal (µP-Lese) und auf die zweite Konfiguration (A14/A15 = 0 1) der Adreßsteuerbits; und
   - das zweite Schreibimpulssignal und das erste Leseimpulssignal als Antwort auf ein Lesesteuersignal (µP-Lese) und auf die dritte Konfiguration (A14/A15 = 1 0) der Adreßsteuerbits.

5. Speicherzugriffssteuer-Schalttechnik gemäß einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, daß die Peripherssteuerungseinrichtung (40) Daten-Bereit-Steuersignale erzeugt, wenn die Daten zwischen dem Peripherregister (41) und dem Speicher (30) übertragen werden sollen, worauf der Prozessor (10) eine Interrupt-Routine ausführt und zusammen mit dem Lesesteuersignal (µP-Lese) die erforderlichen Adreßzugriffs- und Adreßsteuersignale (A0 - A15) erzeugt, um auf das jeweilige Speichermodul zuzugreifen, um die Datenübertragung bezüglich des Peripherregisters (41) zu bewirken.

6. Speicherzugriffssteuer-Schalttechnik gemäß einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, daß die gewünschte Speicherstelle in einem Speichermodul, das durch ein Auswählsteuersignal an seinem Freigabeanschläß aktiviert ist, durch die Adreßzugriffsbitsignale (A_{11} - A_0) adressiert wird, die über die Adreßbusleitungen (21) übersandt werden, wodurch so viele Speicherzugriffsbits in dem ersten Satz von Adreßbits bereitgestellt sind, wie erforderlich sind, um das ausgewählte Speichermodul (12 Bits A_{11} - A_0 für einen Speicherrahmen von 4 kBytes) zu adressieren.

7. Speicherzugriffssteuer-Schalttechnik gemäß Anspruch 6, dadurch gekennzeichnet, daß der Speicher (30) wenigstens zwei Direktzugriffsmodule (RAM 30-1, 30-2) und wenigstens ein Nur-Lese-Speichermodul (ROM 30-3) aufweist, wobei jedes dieser Speichermodule eine Vielzahl (4 kBytes) von adressierbaren Stellen aufweist, die durch eine Vielzahl von Adreßzugriffsbits (A_{11} - A_0) adressiert werden, die in ihrer Zahl entsprechen, um den in jedem Speichermodul verfügbaren Adreßspeicherraum unzweideutig zu adressieren.

Revendications

1. Circuit de commande d'accès mémoire dans un système de traitement de données qui inclut :
   - une mémoire (30) comprenant une pluralité de positions adressables (16 kilobytes) pour mémoriser des données dans différents modules de mémoire (RAM 30-1, RAM 30-2, ROM 30-3);
   - des unités périphériques (45) couplées à un contrôleur de périphérique (40) comprenant un registre périphérique (41) qui, consécutivement à l'émission d'un signal de demande de données, reçoit des données lues à partir de ladite mémoire (30) et qui, consécutivement à l'émission d'un signal de données prêtes, transfère des données pour une écriture dans ladite mémoire (30), lesdits signaux de demande de données et de données prêtes étant générés par ledit contrôleur

* C L M S *

de périphérique (40) chaque fois que l'occasion se présente;

- un processeur (10) capable de répondre auxdits signaux de demande de données et de données prêtes chaque fois qu'ils sont reçus à partir dudit contrôleur de périphérique (40);
- un bus (20) couplant ensemble ladite mémoire, ledit contrôleur de périphérique (40) et ledit processeur (10) pour un transfert direct de données entre lesdites unités (10, 30, 40), ledit bus comprenant :
  - un bus de données (22) pour transférer les données entre lesdites unités (10, 30, 40), et
  - un bus d'adresse (21) pour le transfert de bits d'adresse,
    - un premier sous-groupe (14) de la totalité (16) des bits d'adresse disponibles représentant les bits d'accès d'adresse (A_{13} - A_0) en nombre suffisant pour accéder à toutes lesdites positions mémoire (16 kilobytes), et
    - un deuxième sous-groupe (2) de la totalité (16) des bits d'adresse disponibles représentant les bits de commande d'adresse (A_{15}, A_{14}) pour valider un transfert direct de données sur ledit bus de données (22);
- un circuit logique de commande (50) pour valider l'accès aux données dans ladite mémoire (30) et/ou dans ledit registre périphérique (41), ledit circuit logique de commande comprenant :
  - un multiplexeur (51) dirigeant, sous le contrôle desdits bits de commande d'adresse (A_{15}, A_{14}), des signaux de lecture/écriture (impulsions de lecture μP et d'écriture μP), fournis à ses entrées (A0-3, B0-3, C0-3), vers les unités impliquées dans ledit transfert de données pour effectuer la lecture et l'écriture desdites données à partir des ou dans les positions mémoire accédées desdites unités impliquées et transférant lesdites données sur le bus de données (22), et
  - des moyens de décodage d'adresse (52) sensibles (en E de 53/54) à différentes configurations desdits bits de commande d'adresse (A_{15}, A_{14}) pour déterminer par cette configuration le mode de transfert des données entre les unités impliquées et pour délivrer, aux bornes de sortie (55, 56, 57) desdits moyens de décodage d'adresse, des signaux de sortie alimentant les bornes de validation (E de 30/41) des modules de mémoire et du registre périphérique pour les valider pour qu'ils soient opérationnels et de répondre auxdits signaux de lecture/écriture (lecture RAM/ROM, écriture RAM, lecture périphérique, écriture périphérique) délivrés par ledit multiplexeur (51), par lequel :
    - une première desdites configurations desdits bits de commande d'adresse (A_{15} = "1", A_{14} = "1") valide le processeur (10) pour accéder dans un mode de transfert de données normal à la mémoire (30) ou au registre périphérique (41) pour lire des données à partir de là ou pour lui fournir des données,
    - une deuxième desdites configurations desdits bits de commande d'adresse (A_{15} = "1", A_{14} = "0") effectue un accès mémoire direct dans un premier mode de transfert de données à accès direct (TRANSF PER -> MEM) par lequel des données sont transférées à partir dudit registre périphérique (41) vers ladite mémoire (30), et
    - une troisième desdites configurations desdits bits de commande d'adresse (A_{15} = "0", A_{14} = "1") effectue un accès mémoire direct dans un deuxième mode de transfert de données à accès direct (TRANSF. MEM -> PER) par lequel des données sont transférées à partir de ladite mémoire (30) vers ledit registre périphérique (41),

  et par lequel :
  - chaque fois que la première configuration desdits bits de commande d'adresse (A_{15} = "1", A_{14} = "1") est valable, seule une borne de sortie des moyens de décodage délivre un signal de validation auxdites unités (30-1, 30-2, 30-3 ou 41) impliquées, et
  - chaque fois que la deuxième ou la troisième configuration desdits bits de commande d'adresse (A_{15}, A_{14} = "10" ou "01") est valable, deux bornes de sortie des moyens de décodage délivrent des signaux de validation auxdites unités qui y sont

* C L M S *

couplées pour effectuer un transfert direct de données sur ledit bus de données (22) entre les deux unités ainsi validées,

et par lequel :

- le transfert de données dans les deux modes de transfert à accès direct est exécuté sans avoir recours à un registre interne dudit processeur (10);

caractérisé en ce que le processeur (10), en réponse à la réception desdits signaux de demande de données ou de données prêtes, exécute un sous-programme d'interruption (fig. 6) qui inclut une instruction de lecture (Annexe 2 : "lecture fictive") par laquelle :

- un premier jeu de bits d'accès d'adresse (A_{11} - A_0) est appliqué audit bus d'adresse (21) pour identifier les positions mémoire dans ladite mémoire (30) pour l'accès à des données, et un deuxième jeu d'au moins deux bits d'accès d'adresse (A_{13}, A_{12}) est appliqué aux moyens de décodage d'adresse (52) du circuit logique de commande (50) pour valider l'accès aux données dans ladite mémoire (30) et/ou dans ledit registre périphérique (41);
- lesdits bits de commande d'adresse (A_{15}, A_{14}) sont appliqués auxdits moyens de décodage d'adresse (52) du circuit logique de commande (50) pour la détermination du mode de transfert des données entre les unités impliquées; et
- des signaux d'impulsion de lecture et d'écriture (lecture μP et écriture μP) sont générés, qui sont fournis aux entrées (A0-3, B0-3, C0-3) du multiplexeur (51) pour être dirigés, sous le contrôle des bits de commande d'adresse, vers les unités impliquées dans ledit transfert de données;

et de plus caractérisé en ce que lesdits moyens de décodage (52), en réponse aux deuxième et troisième configurations desdits bits de commande d'adresse (A_{15}, A_{14} = "10" ou "01"), dirigent lesdits au moins deux bits d'accès d'adresse (A_{13}, A_{12}), après leur décodage (2/4), vers les bornes de sortie desdits moyens de décodage de manières différentes telles que :

- chaque fois que la deuxième configuration desdits bits de commande d'adresse (A_{15}, A_{14} = "10") est valable, deux bornes de sortie des moyens de décodage délivrent des signaux de validation aux unités respectives qui y sont couplées; et
- chaque fois que la troisième configuration (A_{15}, A_{14} = "01") est valable, une autre paire de bornes de sortie des moyens de décodage délivre des signaux de validation aux unités respectives qui y sont couplées;

pour effectuer le transfert direct de données sur ledit bus de données (22) entre les deux unités sans rompre l'intégrité connective dudit bus de données (22) avec le processeur (10).

2. Circuit de commande d'accès mémoire selon la revendication 1, caractérisé en ce que les bornes de sortie (0 à 3) dudit multiplexeur (51) sont couplées auxdits modules de mémoire (RAM1, RAM2, ROM; 30-1, 30-2, 30-3) et audit registre périphérique (41) pour y transférer les signaux de commande d'impulsion de lecture/écriture, moyen par lequel :
   - une première sortie (0) délivre un premier signal d'impulsion de lecture (RAM/ROM) aux modules de mémoire qui sont capables de délivrer des données mémorisées;
   - une deuxième sortie (1) délivre un premier signal d'impulsion d'écriture (RAM) aux modules de mémoire qui sont capables d'accepter des données pour les mémoriser;
   - une troisième sortie (2) délivre un deuxième signal d'impulsion de lecture (périphérique) au registre périphérique (41) pour initier la lecture de données à partir de celui-ci; et
   - une quatrième sortie (3) délivre un deuxième signal d'impulsion d'écriture (périphérique) au registre périphérique (41) pour initier l'écriture de données dans celui-ci.

3. Circuit de commande d'accès mémoire selon la revendication 1, caractérisé en ce que lesdits moyens de décodage (52) comprennent :
   - un premier décodeur (53) dont les bornes de sortie (00, 11, 10, 01) sont respectivement couplées à chacun desdits modules de mémoire (30-1, 30-2, 30-3) et audit registre périphérique (41); et
   - un deuxième décodeur (54) dont les bornes de sortie (10, 01) sont respectivement couplées toujours au registre périphérique (41) et à un module de mémoire différent (01 de 54 via 55 à 30-1 et via 57 à 41; 10 de 54 via 56 à 30-2 et via 57 à 41).

4. Circuit de commande d'accès mémoire selon la revendication 2, caractérisé en ce que le

* C L M S *

multiplexeur (51) fournit des combinaisons de signaux de sortie comme suit :

- le premier signal d'impulsion de lecture et le deuxième signal d'impulsion de lecture en réponse à un signal de commande de lecture (lecture µP) et à ladite première configuration (A_{14}, A_{15} = "11") desdits bits de commande d'adresse;
- le premier signal d'impulsion d'écriture et le deuxième signal d'impulsion d'écriture en réponse à un signal de commande d'écriture (écriture µP) et à ladite première configuration (A_{14}, A_{15} = "11") desdits bits de commande d'adresse;
- le deuxième signal d'impulsion de lecture et le premier signal d'impulsion d'écriture en réponse à un signal de commande de lecture (lecture µP) et à ladite deuxième configuration (A_{14}, A_{15} = "01") desdits bits de commande d'adresse; et
- le deuxième signal d'impulsion d'écriture et le premier signal d'impulsion de lecture en réponse à un signal de commande de lecture (lecture µP) et à ladite troisième configuration (A_{14}, A_{15} = "10") desdits bits de commande d'adresse;

5. Circuit de commande d'accès mémoire selon l'une quelconque des revendications précédentes, caractérisé en ce que le contrôleur de périphérique (40) produit des signaux de commande de données prêtes lorsque les données doivent être transférées entre le registre périphérique (41) et la mémoire (30), après quoi le processeur (10) exécute un sous-programme d'interruption et, conjointement avec le signal de commande de lecture (lecture µP), produit les signaux d'accès d'adresse et de commande d'adresse (A_0 - A_{15}) exigés pour accéder au module de mémoire respectif pour effectuer le transfert des données relativement au registre périphérique (41).

6. Circuit de commande d'accès mémoire selon l'une quelconque des revendications précédentes, caractérisé en ce que la position mémoire désirée dans un module de mémoire qui est activé par un signal de commande sélectif à sa borne de validation est adressée par l'intermédiaire des signaux binaires d'accès d'adresse (A_{11} - A_0) envoyés sur les lignes du bus d'adresse (21), moyen par lequel autant de bits d'accès d'adresse que nécessaires sont fournis dans ledit premier jeu de bits d'adresse pour l'adressage du module de mémoire choisi (12 bits A_{11} - A_0 pour un espace mémoire de 4 kilobytes).

7. Circuit de commande d'accès mémoire selon la revendication 6, caractérisé en ce que la mémoire (30) comprend au moins deux modules à accès direct (RAM 30-1, RAM 30-2) et au moins un module de mémoire morte (ROM 30-3), chacun de ces modules de mémoire ayant une pluralité (4 kilobytes) de positions adressables qui sont adressées par une pluralité de bits d'accès d'adresse (A_{11} - A_0) dont le nombre est approprié à l'adressage univoque de l'espace d'adressage total disponible dans chaque module de mémoire.

* C L M S *