

Description**Background of the invention****Field of the invention**

5 This invention relates to storage addressing and, more particularly, to apparatus and method for controlling access to the same or different pages of storage by a plurality of direct storage (memory) access (DMA) channels.

Discussion of the prior art

10 This invention is an improvement on that of European Patent Application EP—A—0057815 for Page Addressing Mechanism and Method for Using the Same by D. J. Bradley et al. This application falls within the terms of Article 54(3) EPC.

The Bradley reference extends the size of memory which can be addressed by an address bus carrying N bits of information from the normal 2^N locations to a multiple of 2^N by providing a plurality of register means each of which is separately programmable to store data capable of being selectably provided as page signals. Selection of the page registers is made by control signals manifesting the then occurring storage operation, such as instruction fetch, storage read, or storage write operations. However, the Bradley addressing technique suffers in a system where there exists a plurality of direct storage (or memory) access channels (DMA). All DMA channels which can operate simultaneously must be directed to the same extended region of the storage address space. This may result in the necessity for double buffering of data, which reduces system performance and increases storage requirements.

The present invention provides, from one aspect, storage addressing control apparatus including a processor having an address bus for manifesting a first plurality of address signals, and a direct memory access controller coupled to the address bus (a) for manifesting a second plurality of address signals, less in number than the first plurality and (b) also providing means for accessing the storage by a plurality of selectively operable channels without the need for constant intervention by the processor,

a plurality of programmable registers coupled to the address bus for manifesting a third plurality of address signals equal to or less in number than the difference between the first and second pluralities; and decoding means responsive to operation of the direct memory access controller on behalf of a respective one of the operating channels for gating, to the address bus, the address signals manifested by the programmable register corresponding to the respective operating channel.

From another aspect, the present invention provides a method of operating a computing apparatus, comprising a processor, a system bus including a data bus and an address bus capable of manifesting a plurality N of address bits, a plurality of direct memory access channels, a direct memory access controller selectively capable of storing $M < N$ address bits on behalf of each of the channels and an array of page registers capable of storing $N - M$ address bits for a corresponding one of the channels, the method comprising the steps of:

responsive to a request from a requesting one of the channels, establishing control of the system bus in the direct memory access controller;

40 responsive to the establishment of such control of the system bus, signalling an acknowledgement signal to the requesting channel;

loading onto the address bus the $M < N$ address bits forming a data transfer address stored by said controller on behalf of the requesting channel; and

gating with the acknowledgement signal the $N - M$ address bits to the address bus from the page register corresponding to the requesting channel.

It is pointed out that EP—A—10 263 discloses an address extension unit interfacing between a processor and storage which has a plurality of registers, each selectable, in part or completely, by energised conductor activated by instruction decoding in the processor. One such register, REG4, is definitively selected by a single line, referred to as a "direct memory access" line and this register adds two fixed high order address bits to those provided by the processor on the address bus, these latter address bits being passed unaltered and unstored by the register. In the sense that the term "direct memory/storage access controller" is used in the specification and in the present form of the art, EP—A—10 263 does not disclose such. The whole point of having a "controller" is to remove an overhead from the processor by giving control of the address bus to the controller. In EP—A—10 263, the highest addressing activity is imposed on the processor in the so-called DMA mode. Were a plurality of REG4's to be incorporated in the extension unit of EP—A—10 263, and it to be otherwise modified to incorporate plural I/O channels, the overhead on the processor would be increased to a wholly unacceptable level.

The present invention will be described further by way of example with reference to an embodiment thereof and its method of operation as illustrated in the accompanying drawings in which:

60 Figure 1 is a block diagram showing a typical DMA apparatus for addressing DMA channels to system storage.

Figure 2 is a block diagram showing the apparatus of the selected embodiment of the invention for addressing a plurality of DMA channels into the same or distinct regions of system storage;

Figures 3A and 3B are block diagrams showing the bus and addressing apparatus of Figure 2 in greater detail; these are aligned as shown in Figure 3; and

Figure 4 is a map diagram of storage address space illustrating DMA page rollover.

Description of the preferred embodiments

Herein, by way of example, processor unit 10 comprises an Intel 8088 microprocessor with a twenty bit address bus 12 capable of addressing in excess of one million locations in storage 16, and an 8 bit data bus 14. Storage 16 may comprise standard read only storage and random access memory devices.

Input/output devices 18 are coupled to data bus 14 for transferring data with respect to storage 16 under control of DMA controller 20. DMA controller 20 herein comprises an Intel 8237 or Intel DMA 8257 device with a capacity for coupling four channels (one is illustrated) to the eight bit data bus 14. DMA 20 address lines 28 couple sixteen address bits to the twenty bit address bus 12.

In the simplified diagram of Figure 1, line 22 represents the many control and status lines to be described hereafter for controlling the activity on busses 12, 14. Control lines interconnecting processor 10 and DMA 20 to be specifically mentioned at this point include bus request (also referred to as HOLD) line 26 and bus grant (also referred to as HOLD A) line 24. Control lines interconnecting DMA to a single input/output means (such as a plurality of I/O devices attached to an I/O adapter 18), and thus establishing a given instance of a channel, include data request (DRQ) line 31 and data acknowledge (DACK) line 30. A DRQ/DACK line pair exists for each channel.

In operation, approximately one typical bus cycle under DMA control includes the following steps:

(1) Processor 10 communicates address information over data busses 14, 29 to DMA 20, where it is stored in the DMA 20 address register addressed by the four low order bits A0—A3 on address bus 12, indicating the location in storage 16, with respect to which data is to be transferred.

(2) Processor 10 communicates count information to DMA 20 over data busses 14, 29, where it is stored in the count register addressed by the four low order bits A0—A3 on the address bus 12, indicating the number of words or characters to be transferred.

(3) I/O controller 18, when ready to transfer, brings up data request (DRQ) line 31.

(4) Responding to DRQ 31, DMA 20 signals HOLD 26 to processor 10 to request access to busses 12, 14.

(5) Responding to HOLD 26, processor 10 signals HOLD A 24 to DMA 20, granting access to busses 12, 14.

(6) DMA 20, now controlling system busses 12, 14, loads address bus 12 with the address stored at step (1) in the DMA 20 address register for the currently active channel.

(7) DMA 20 signals DACK 30 to I/O controller 18.

(8) I/O controller 18 loads its data on data bus 14.

(9) DMA 20 controls selected control (CTL) lines 22 to indicate storage work to be performed.

(10) DMA 20 releases the system busses.

A DMA operation typically includes transfer of a plurality of data characters (a character is 8 bits for the 8 bit data bus in this example), and includes the following steps:

(1) load address on address bus 12 from the DMA 20 address register corresponding to the active channel;

(2) perform bus operation (data transfer over bus 14);

(3) increment (or, equivalently, decrement) DMA 20 address register;

(4) decrement the DMA 20 count register corresponding to the active channel;

(5) repeat steps (1)—(4) until the count register reaches zero;

(6) signal end of operation (see Figure 3, pin EOP).

Referring now to Figure 2, a description will be given of the addressing apparatus of the invention. In Figure 2 there is provided, by way of example and not limitation, apparatus for controlling four DMA channels and for paging each DMA channel into selectable pages of 2^{16} , or approximately 64K, addressable locations within an address space comprising 2^{20} , or approximately one million such locations. This particular example results from the combination of a twenty bit system address bus 12, with a four channel DMA chip 20 having 16 address lines 28, 29, and a four bit page array register 40, as will be further described hereinafter.

As shown in Figure 2, processor chip 10 includes among its input/output lines an 8 bit data bus 14, a 20 bit address bus 12, a plurality of control and status lines 22, write control line 23, hold acknowledge line 24, and hold line 26. Control and status lines are coupled to I/O device read/write control and decode module 60, storage 16, and I/O device adapters 18. Eight bit data bus 14 is coupled to storage 16, adapters 18, via lines 29 to the eight data ports D0—D7 of DMA 20, and via the four (D0—D3) lines 62 to the data input terminals D1, D2, D3 and D4 of page array 40. The eight lines 29 from DMA 20 may be used for either data or address, and in the latter case are gated by AEN line 58 and ADSTB line 56 through 8 bit latch 50 via lines 54 to provide address bits A8—A15 to address bus 12. DMA 20 address bits A0—A7 are coupled (through buffer 51 shown in Figure 3) by lines 28 to address bus 12. Address bus 12 is also coupled to control and decode 60, storage 16, device adapters 18, and the two low order bits (A0, A1) to the WA and WB input terminals of page array chip 40.

The output of I/O device read/write control and decoder is a read-device/write-device line pair for each of the attached devices.

Write control line 23 of processor 10 is coupled to the WRITE terminal of page array register 40.

Address enable (AEN) line 58 indicates that DMA has control of the system busses 12, 14, and in this

exemplary implementation is fed from the AEN terminal of DMA 20 to the OE terminal of latch 50, and via inverter 52 to the READ terminal of page array register 40. The address strobe (ADSTB) terminal of DMA 20 is coupled to the clock CLK terminal of latch 50 by line 56, and becomes active to latch information on lines 29 into latch 50.

Device acknowledge DACK 1 line 37, DACK 2 line 38, DACK 3 line 39 are encoded by OR array 42, 46 and fed to the read A (RA) and read B (RB) input terminals of page array 40. Device request DRQ 0 line 32, DRQ 1 line 33, DRQ 2 line 34, and DRQ 3 line 35, together with lines 36—38 couple DMA 20 to I/O device adapters 18, with a DRQ/DACK pair for each DMA channel.

Page array 40 output terminals Q1—Q4 are coupled to the A16—A19 bits of address bus 12 by lines 46.

In this example, page array 40 comprises a Texas Instruments standard TTL circuit type 74LS670 module latch 50 a standard TTL circuit type LS373, buffer 51 (Figure 3) a standard TTL circuit type LS244, DMA 20 an Intel 8237 (or, equivalently, an Intel 8257) DMA device.

In Figure 3 is shown in greater detail the interconnection of DMA 20, page array 40, 8 bit latch 50 (all from Figure 2) and buffer 51 which buffers address bits A0—A7 from DMA 20 onto lines 28 and address bus 12. As illustrated in Figure 3, DMA chip 20 comprises 40 terminal pins, the pertinent pins including address bits A0—A7, combined data/address pins DB0—DB7, data request pins DREQ0—DREQ3, data acknowledge pins DACK0—DACK3, bus (or hold HOLD) request pin HRQ26, ADSTB pin 56, address enable AEN pin 58, and group of control pins from processor 10 including chip select CS, hold acknowledge HLDA24, I/O read pin IOR, and I/O write pin IOW.

In operation, the apparatus of Figures 2 and 3 extends the addressing capability of DMA 20 from the 16 bits provided by an Intel 8237 DMA chip to the full 20 bits system address bus provided by an Intel 8088 microprocessor chip (such as is described in A. P. Morse, *The 8086 Primer*, Hayden Book Co., Inc., copyright 1980, Library of Congress number QA76.8.1292M67 001.6'4'04 79-23932 SBN 0-8104-5165-4) by selecting a pre-programmed page register for each separate DMA channel. Thus, simultaneous DMA operations can take place in the same or different 64K byte block regions, or pages.

Page array 40 includes four four-bit registers, each of which is individually programmable by processor 10 to provide different or identical high order address bits A16—A19 at output terminals Q1—Q4. To program one of these array 40 registers, the register address (00, 01, 10, or 11) is established at the WA, WB terminals, the data bus 14 bits D0—D3 applied to array 40 terminals D1—D4 respectively, and loaded into the addressed register when WRITE 23 is activated.

When a specific channel is requested by an I/O device 18 on a DRQ line, DMA 20 indicates that this channel is active by sending a DACK signal unique to that channel and by activating the active enable AEN line 58 signalling that any channel is active. In order to gate address bits A16—A19 from one of array 40 registers to thereby establish the storage 16 page in which the active channel is to operate, DACK lines 37—39 are encoded at gates 42, 44 to provide at array 40 read address input terminals RA, RB the address of the array 40 register corresponding to the active DMA channel. When the array 40 READ terminal is activated by DMA 20 signal AEN on line 58, the contents of the array 40 register selected by RA, RB is placed at array 40 terminals Q1—Q4 to load the high order address bits A16—A19 on address bus 12. The remaining, low order 16 address bits A0—A15 are loaded onto address bus 12 by buffer 51 (bits A0—A7) and latch 50 (bits A8—A15) respectively, which receive their address bits from DMA 20.

DMA 20 is primed with the 16 low order address bits by controller 10 via data bus 14, as was described in connection with Figure 1.

In Table 1 is set forth a source code listing in Intel ASM-86 assembly language (as described in Morse, supra) illustrating an example of the addressing technique of the invention, together with one approach for preventing rollover with DMA paging. Referring to Figure 4, to illustrate paging and rollover, assume that the high order address bits stored in the array register 40 for an active DMA channel are 1110, or hex 7; that the low order address bits stored in DMA 20 are 1111 1111 1111 1110, or hex FFFE. Assume further that a data transfer is initiated which will write into more than two storage (16) locations. After location 7FFFE and 7FFFF have been written, because there is no carry provided from address bit A15 to A16, the next storage 16 location to receive data will be 70000, and not 80000 as would normally be desired. Consequently, the procedure of Table 1 illustrates steps 1 and 2 of the sequence described above in connection with Figure 1, as adapted to the apparatus of Figures 2 and 3, to prime DMA 20 with the beginning address for, and the count of a data transfer.

TABLE 1: DMA SET-UP

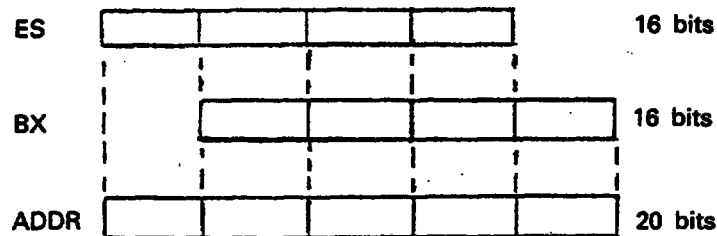
(10)

LINE	SOURCE			
5	611	-----		
	612	; DMA_SETUP		
	613	; THIS ROUTINE SETS UP THE DMA FOR READ/WRITE/VERIFY		
	614	; OPERATIONS.		
	615	; INPUT		
10	616	; [AL]=MODE BYTE FOR THE DMA		
	617	; [ES:BX]—ADDRESS TO READ/WRITE THE DATA		
	618	; OUTPUT		
	619	; [AX] DESTROYED		
	620	-----		
15	621	DMA_SETUP	PROC	NEAR
	622	PUSH	CX	; SAVE THE REGISTER
	623	OUT	DMA+12,AL	; SET THE FIRST/LAST F/F
	624	OUT	DMA+11,AL	; OUTPUT THE MODE BYTE
20	625	MOV	AX,ES	; GET THE ES VALUE
	626	MOV	CL,4	; SHIFT COUNT
	627	ROL	AX,CL	; ROTATE LEFT
	628	MOV	CH,AL	; GET HIGHEST NYBLE OF ES TO CH
	629	AND	AL,0F0H	; ZERO THE LOW NYBBLE FROM SEGMNT
	630	ADD	AX,EX	; TEST FOR CARRY FROM ADDITION
25	631	JNC	SKIP_INC	
	632	INC	CH	; CARRY MEANS HIGH 4 BITS MUST BE
	633	INC	CH	; INC
	634	SKIP_INC:		
	635	PUSH	AX	; SAVE START ADDRESS
30	636	OUT	DMA+4,AL	; OUTPUT LOW ADDRESS
	637	MOV	AL,AH	
	638	OUT	DMA+4,AL	; OUTPUT HIGH ADDRESS
	639	MOV	AL,CH	; GET HIGH 4 BITS
	640	AND	AL,0FH	
35	641	OUT	DMA_HIGH,AL	; OUTPUT THE HIGH 4 BITS
	642			
	643	;-----DETERMINE COUNT		
	644			
	645	MOV	AH,DH	; NUMBER OF SECTORS
40	646	SUB	AL,AL	; TIMES 256 INTO AX
	647	SHR	AX,1	; SECTORS * 128 INTO AX
	648	PUSH	AX	
	649	MOV	BX,6	; GET THE BYTES/SECTOR PARM
45	650	CALL	GET_PARM	
	651	MOV	CL,AH	; USE AS SHIFT COUNT [0=128,
	652	POP	AX	; 1=256 ETC]
	653	SHL	AX,CL	; MULTIPLY BY CORRECT AMOUNT
	654	DEC	AX	; -1 FOR DMA VALUE
50	655	PUSH	AX	; SAVE COUNT VALUE
	656	OUT	DMA+5,AL	; LOW BYTE OF COUNT
	657	MOV	AL,AH	
	658	OUT	DMA+5,AL	; HIGH BYTE OF COUNT
	659	POP	CX	; RECOVER COUNT VALUE
55	660	POP	AX	; RECOVER ADDRESS VALUE
	661	ADD	AX,CX	; ADD, TEST FOR 64K OVERFLOW
	662	POP	CX	; RECOVER REGISTER
	663	MOV	AL,2	; MODE FOR 8237
	664	OUT	DMA+10,AL	; INITIALIZE THE DISKETTE CHANNEL
60	665	RET.		; RETURN TO CALLER, CFL SET BY
	666	DMA_SETUP	ENDP	; ABOVE IF ERROR
	667	\$EJECT		

65 In Table 1, at lines 624—625 the read/write mode is selected; at lines 626—634 the twenty bit system address is formed from the ES and BX registers; at lines 636—638 the lower order 16 bits are loaded to

DMA 20; at lines 639—641 the high order 4 bits are loaded to a page register 40, and at line 661 a test is made to assure that rollover will not occur. (In the Intel 8088, a twenty bit address is generated from the two 16-bit ES and BX registers by offsetting the two registers by one four bit nybble and adding the offset registers, as is illustrated in Table 2.)

TABLE 2
TWENTY BIT ADDRESS FORMULATION



Claims

1. Storage addressing control apparatus including a processor (10) having an address bus (12) for manifesting a first plurality of address signals (A0—A19), and a direct memory access controller (20) coupled to the address bus (a) for manifesting a second plurality of address signals (A0—A15), less in number than the first plurality and (b) also providing means (24, 26, 32—39) for accessing the storage by a plurality of selectively operable channels without the need for constant intervention by the processor, a plurality of programmable registers (40) coupled to the address bus for manifesting a third plurality of address signals (A16—A19) equal to or less in number than the difference between the first and second pluralities; and

decoding means (42, 44) responsive to operation of the direct memory access controller on behalf of a respective one of the operating channels for gating (Q1—Q4), to the address bus, the address signals manifested by the programmable register (40) corresponding to the respective operating channel (18).

2. Apparatus as claimed in claim 1 wherein the programmable registers are loaded with distinct or overlapping page addresses.

3. Apparatus as claimed in either preceding claim wherein the means for accessing the storage by a plurality of selectively operable channels comprises signalling means for signalling an acknowledgement to a request to access the data bus by one of the plurality of channels, the decoding means being responsive thereto.

4. A method of operating a computing apparatus, comprising a processor, a system bus including a data bus and an address bus capable of manifesting a plurality N of address bits, a plurality of direct memory access channels, a direct memory access controller selectively capable of storing $M < N$ address bits on behalf of each of the channels and an array of page registers capable of storing $N - M$ address bits for a corresponding one of the channels, the method comprising the steps of:

responsive to a request from a requesting one of the channels, establishing control of the system bus in the direct memory access controller;

responsive to the establishment of such control of the system bus, signalling an acknowledgement signal to the requesting channel;

loading onto the address bus the $M < N$ address bits forming a data transfer address stored by said controller on behalf of the requesting channel; and

gating with the acknowledgement signal the $N - M$ address bits to the address bus from the page register corresponding to the requesting channel.

5. A method as claimed in claim 4, further comprising the steps of:

operating the processor to store in the direct memory access controller the $M < N$ address bits for a first channel, and in the page register corresponding to the first channel $N - M$ address bits;

storing in the direct memory access controller a count manifesting the number of data characters to be transferred;

transferring a data character on the data bus, and thereupon causing the controller to automatically increment the data transfer address and decrement the count; and

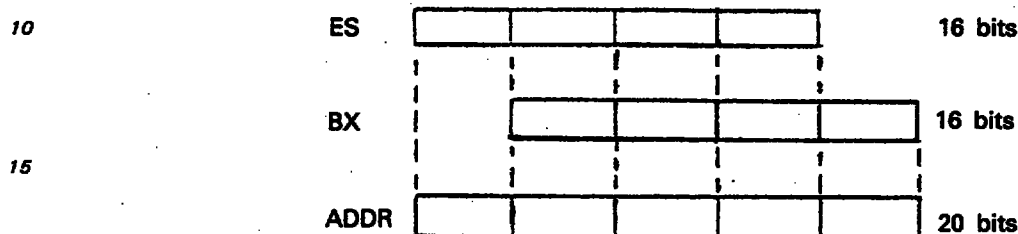
causing said controller to automatically repeat the loading and transferring steps until the count is decremented to zero.

Patentansprüche

1. Speicheradressiersteuervorrichtung, welche einen Prozessor (10) mit einem Adressenbus (12) zur Aufnahme einer ersten Anzahl von Adressensignalen (A0—A19) und eine mit dem Adressenbus gekoppelte Direktspeicherzugriffssteuerung (20) (a) zur Aufnahme einer zweiten Anzahl von Adressen-

DMA 20; at lines 639—641 the high order 4 bits are loaded to a page register 40, and at line 661 a test is made to assure that rollover will not occur. (In the Intel 8088, a twenty bit address is generated from the two 16-bit ES and BX registers by offsetting the two registers by one four bit nybble and adding the offset registers, as is illustrated in Table 2.)

TABLE 2
TWENTY BIT ADDRESS FORMULATION



Claims

20

1. Storage addressing control apparatus including a processor (10) having an address bus (12) for manifesting a first plurality of address signals (A0—A19), and a direct memory access controller (20) coupled to the address bus (a) for manifesting a second plurality of address signals (A0—A15), less in number than the first plurality and (b) also providing means (24, 26, 32—39) for accessing the storage by a plurality of selectively operable channels without the need for constant intervention by the processor, a plurality of programmable registers (40) coupled to the address bus for manifesting a third plurality of address signals (A16—A19) equal to or less in number than the difference between the first and second pluralities; and

decoding means (42, 44) responsive to operation of the direct memory access controller on behalf of a respective one of the operating channels for gating (Q1—Q4), to the address bus, the address signals manifested by the programmable register (40) corresponding to the respective operating channel (18).

2. Apparatus as claimed in claim 1 wherein the programmable registers are loaded with distinct or overlapping page addresses.

3. Apparatus as claimed in either preceding claim wherein the means for accessing the storage by a plurality of selectively operable channels comprises signalling means for signalling an acknowledgement to a request to access the data bus by one of the plurality of channels, the decoding means being responsive thereto.

4. A method of operating a computing apparatus, comprising a processor, a system bus including a data bus and an address bus capable of manifesting a plurality N of address bits, a plurality of direct memory access channels, a direct memory access controller selectively capable of storing M<N address bits on behalf of each of the channels and an array of page registers capable of storing N—M address bits for a corresponding one of the channels, the method comprising the steps of:

responsive to a request from a requesting one of the channels, establishing control of the system bus in the direct memory access controller;

responsive to the establishment of such control of the system bus, signalling an acknowledgement signal to the requesting channel;

loading onto the address bus the M<N address bits forming a data transfer address stored by said controller on behalf of the requesting channel; and

gating with the acknowledgement signal the N—M address bits to the address bus from the page register corresponding to the requesting channel.

5. A method as claimed in claim 4, further comprising the steps of:

operating the processor to store in the direct memory access controller the M<N address bits for a first channel, and in the page register corresponding to the first channel N—M address bits;

storing in the direct memory access controller a count manifesting the number of data characters to be transferred;

transferring a data character on the data bus, and thereupon causing the controller to automatically increment the data transfer address and decrement the count; and

causing said controller to automatically repeat the loading and transferring steps until the count is decremented to zero.

60

Patentansprüche

1. Speicheradressiersteuervorrichtung, welche einen Prozessor (10) mit einem Adressenbus (12) zur Aufnahme einer ersten Anzahl von Adressensignalen (A0—A19) und eine mit dem Adressenbus gekoppelte Direktspeicherzugriffssteuerung (20) (a) zur Aufnahme einer zweiten Anzahl von Adressen-



- signalen (A0—A15), die kleiner als die erste Anzahl ist, und (b) auch zur Schaffung von Mitteln (24, 26, 32—39) für einen Zugriff auf den Speicher durch eine Anzahl von ausgewählt betreibbaren Kanälen ohne die Notwendigkeit für eine konstante Intervention durch den Prozessor,
- 5 eine Anzahl von mit dem Adressenbus gekoppelten programmierbaren Registern (40) zur Aufnahme einer dritten Anzahl von Adressensignalen (A16—A19), die gleich oder kleiner als die Differenz zwischen der ersten und zweiten Anzahl ist; und
- 10 Dekodiermittel (42, 44), die auf ein Betreiben der Direktspeicherzugriffssteuerung zugunsten eines betreffenden der arbeitenden Kanäle ansprechen, für ein Tor (Q1—Q4), auf den Adressenbus, der durch das programmierbare Register (40), welches dem betreffenden arbeitenden Kanal (18) entspricht, aufgenommenen Adressensignale enthält.
2. Vorrichtung nach Anspruch 1, bei welcher die programmierbaren Register mit getrennten oder überlappenden Seitenadressen geladen sind.
3. Vorrichtung nach irgendeinem vorstehenden Anspruch, bei welcher die Mittel für einen Zugriff auf den Speicher durch eine Anzahl von ausgewählt betreibbaren Kanälen Signalisierungsmittel zur
- 15 Signalisierung einer Bestätigung zu einer Anforderung eines Zugriff auf den Datenbus durch einen der Anzahl von Kanälen umfassen, wobei die Dekodiermittel darauf ansprechen.
4. Verfahren zum Betreiben einer Rechenvorrichtung mit einem Prozessor, einem Systembus, welcher einen Datenbus und einen Adressenbus, der eine Anzahl N von Adressenbits aufnehmen kann, enthält, einer Anzahl von Direktspeicherzugriffskanälen, einer Direktspeicherzugriffssteuerung, die ausgewählt in
- 20 der Lage ist, $M < N$ Adressenbits zugunsten eines jeden der Kanäle zu speichern, und einer Anordnung von Seitenregistern, die in der Lage sind, $N - M$ Adressenbits für einen entsprechenden der Kanäle zu speichern, wobei das Verfahren die Verfahrensschritte des
- Errichtens einer Steuerung des Systembusses in der Direktspeicherzugriffssteuerung ansprechend auf eine Anforderung eines anfordernden der Kanäle;
- 25 Signalisierens eines Bestätigungssignals an den anfordernden Kanal ansprechend auf die Errichtung einer solchen Steuerung des Systembusses,
- Ladens der $M < N$ Adressenbits, die eine durch die Steuerung zugunsten des anfordernden Kanals gespeicherte Datenübertragungsadresse bilden, auf den Adressenbus; und
- Torens der $N - M$ Adressenbits auf den Adressenbus aus dem dem anfordernden Kanal entsprechenden
- 30 Seitenregister mit dem Bestätigungssignal umfaßt.
5. Verfahren nach Anspruch 4, welches ferner die Verfahrensschritte des
- Betreibens des Prozessors zu einer Speicherung der $M < N$ Adressenbits für einen ersten Kanal in der Direktspeicherzugriffssteuerung und in dem Seitenregister, das den $N - M$ Adressenbits des ersten Kanals
- entspricht;
- 35 Speicherns einer die Anzahl von zu übertragenden Datenzeichen wiedergebenden Zählung in der Direktspeicherzugriffssteuerung;
- Übertragens eines Datenzeichens auf den Datenbus und des darauffolgenden Bewirkens, daß die Steuerung automatisch die Datenübertragungsadresse inkrementiert und die Zählung dekrementiert; und
- 40 Bewirkens, daß die Steuerung die Lade- und Übertragungsschritte automatisch wiederholt, bis die Zählung auf null dekrementiert ist, umfaßt.

Revendications

1. Dispositif de commande d'adressage de mémoire comprenant un processeur (10) comportant un
- 45 bus (12) de transmission d'adresses servant à transmettre une première pluralité de signaux d'adresses (A0—A19), et un contrôleur (20) d'accès direct en mémoire, accouplé au bus de transmission d'adresses (a) pour transmettre une seconde pluralité de signaux d'adresses (A0—A15), en un nombre inférieur à celui de la première pluralité, et (b) pour fournir également des moyens (24, 26, 32—39) permettant un accès à la
- 50 mémoire au moyen d'une pluralité de canaux pouvant fonctionner de façon sélective, sans le besoin d'une intervention permanente du processeur,
- une pluralité de registres programmables (40) accouplés au bus de transmission d'adresses pour la transmission d'une troisième pluralité de signaux d'adresses (A16—A19) en un nombre égal ou inférieur au nombre formé par la différence entre les première et seconde pluralités; et
- 55 des moyens de décodage (42, 44) sensibles au fonctionnement du contrôleur d'accès direct en mémoire pour le compte d'un canal respectif faisant partie des canaux en fonctionnement pour commander le transfert (Q1—Q4), au bus de transmission d'adresses, des signaux d'adresses délivrés par le registre programmable (40) et correspondant au signal respectif (18) en fonctionnement.
2. Dispositif selon la revendication 1, dans lequel les registres programmables sont chargés par des
- 60 adresses de pages distinctes ou en chevauchement.
3. Dispositif selon l'une ou l'autre des revendications précédentes, dans lequel les moyens permettant à une pluralité de canaux, que l'on peut faire fonctionner de façon sélective, d'accéder à la mémoire comportent des moyens de signalisation servant à signaler un accusé de réception d'une demande d'accès d'un canal faisant partie de la pluralité de canaux au bus de transmission de données, à des moyens de
- 65 décodage répondant à une telle demande d'accès.
4. Procédé pour faire fonctionner un dispositif de calcul, comprenant un processeur, un bus du



système incluant un bus de transmission de données et un bus de transmission d'adresses, pouvant transmettre une pluralité N de bits d'adresses, une pluralité de canaux d'accès direct en mémoire, un contrôleur d'accès direct en mémoire apte à mémoriser de façon sélective M<N bits d'adresses pour le compte de chacun des canaux et un réseau de registres de pages, capable de mémoriser N-M bits d'adresses pour l'un correspondant des canaux, ce procédé incluant les étapes consistant à:

- 5 établir la commande du bus du système dans le contrôleur d'accès direct en mémoire, en réponse à une demande provenant de l'un des canaux effectuant la demande;
signaler un signal d'accusé de réception au canal demandeur, en réponse à l'établissement d'une telle commande du bus du système;
10 charger dans le bus de transmission d'adresses les M<N bits d'adresses constituant une adresse de transmission de données mémorisée par ledit contrôleur pour le compte du canal demandeur; et commander, au moyen du signal d'accusé de réception, le transfert des N-M bits d'adresses au bus de transmission d'adresses à partir du registre de pages correspondant au canal demandeur.
5. Procédé selon la revendication 4, comprenant en outre les étapes consistant à:
15 faire fonctionner le processeur de manière à mémoriser les M<N bits d'adresses pour un premier canal dans le contrôleur d'accès direct en mémoire et dans le registre de pages correspondant au N-M bits d'adresses du premier canal;
mémoriser, dans le contrôleur d'accès direct en mémoire, un nombre représentant le nombre des caractères de données devant être transférés;
20 transférer un caractère de données dans le bus de transmission de données et amener alors le contrôleur à incrémenter de façon automatique l'adresse de transfert des données et à décrémenter ledit nombre; et
amener ledit contrôleur à répéter de façon automatique les étapes de chargement et de transfert jusqu'à ce que le nombre soit décrémenté jusqu'à zéro.

25

30

35

40

45

50

55

60

65

