

THE MEMORY MANAGEMENT UNIT

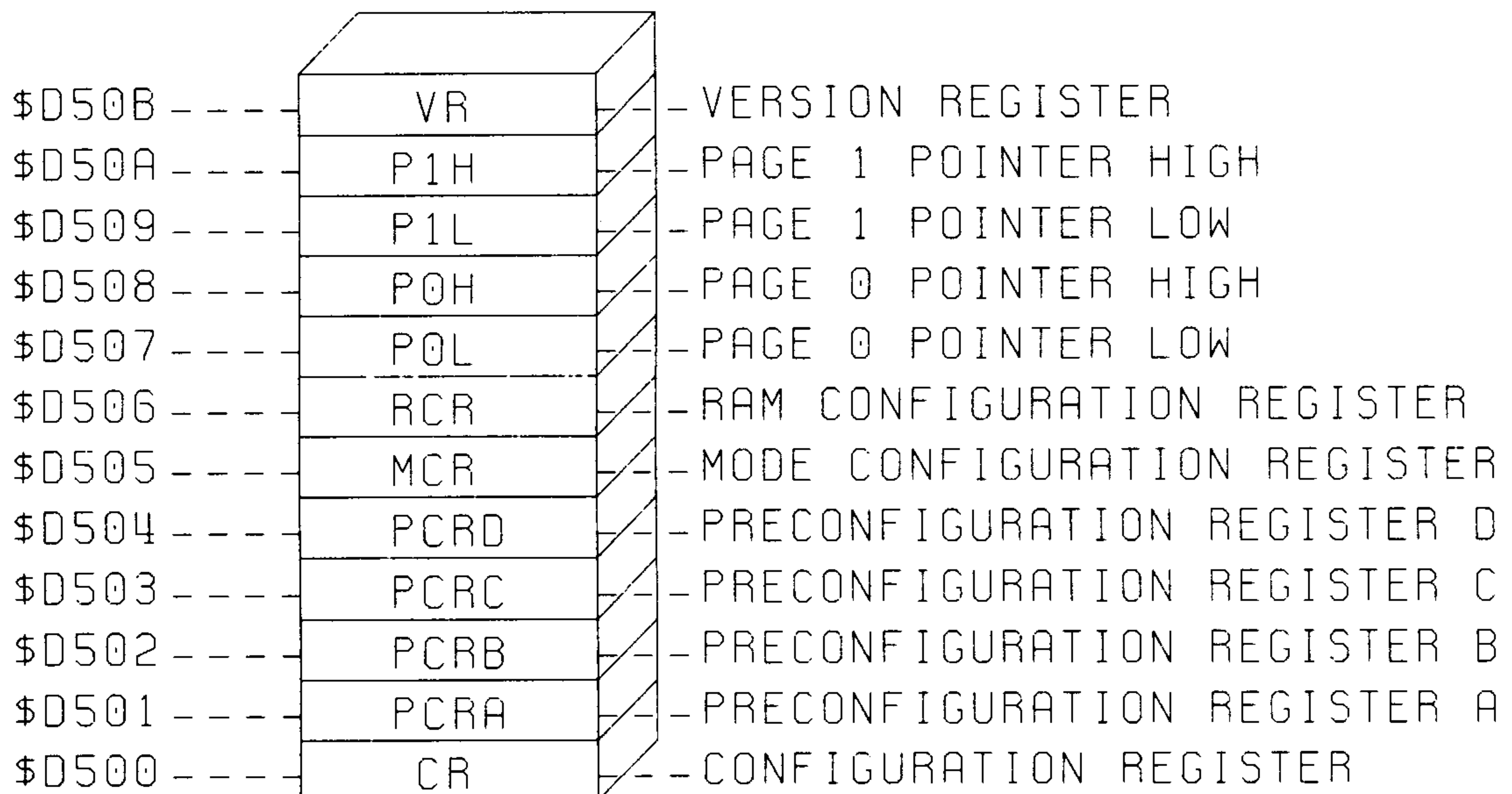
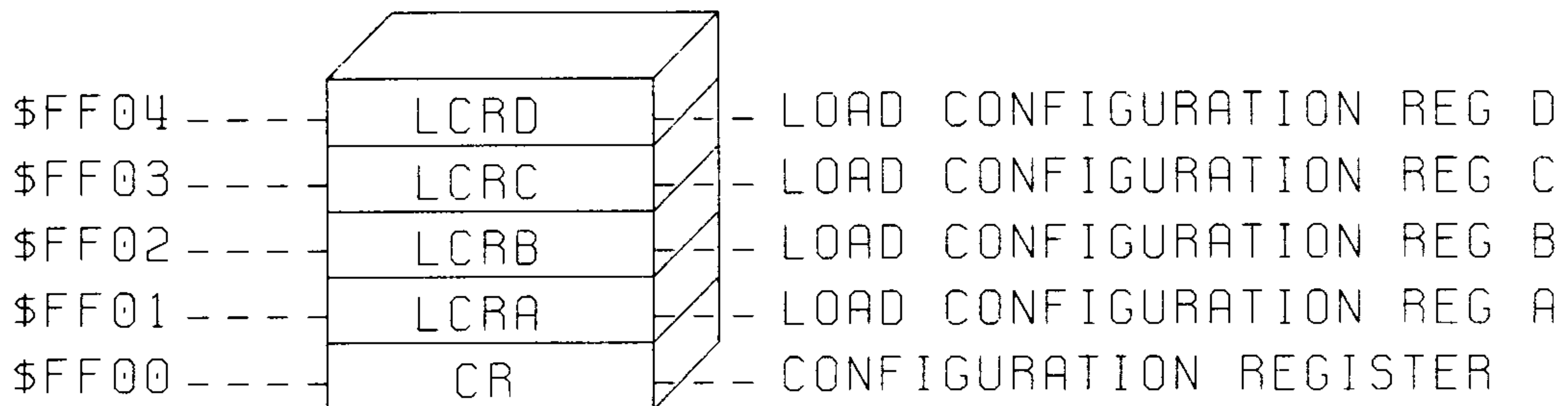
FOLD OUT SCHEMATIC SHEET 2, PAGE 74, FOR EASY REFERENCE.

The MMU is designed to allow complex control of the C128 system memory resources. It handles all of the standard **C64 modes of operation** in a fashion as to be completely compatible with the C64. Additionally, it controls the management of particular C128 modes including the Z-80 mode.

Summary of MMU functions:

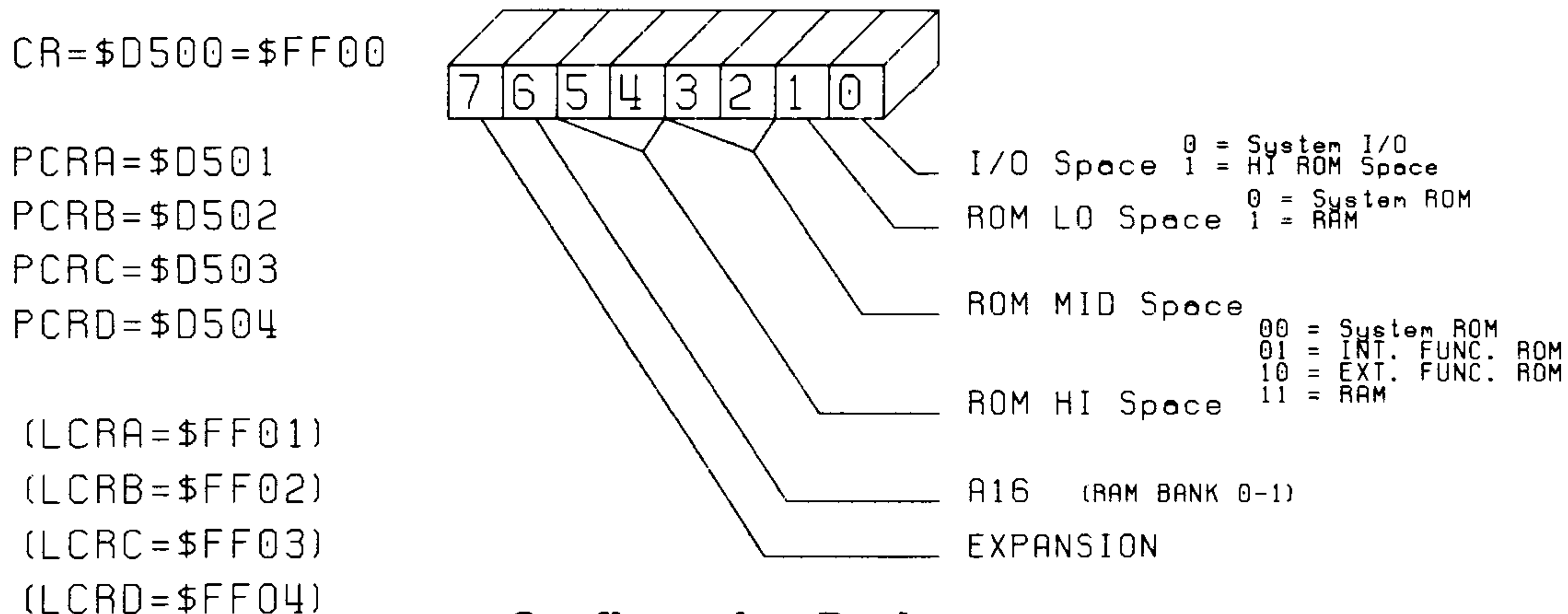
- Generation of Translated Address Bus, TA₈ – TA₁₅.
- Generation of control signals for different processor modes – C128, C64, Z-80.
- Generation of CAS select lines for RAM banking.
- Generation of ROMBANK (MS₀, MS₁) lines for ROM banking.

The MMU is the mechanism by which the various memory modes shown in the C128 Memory Map are chosen. Additionally, the MMU provides for Z-80 mode, which was not shown on that diagram. Following is a description of the MMU register types. Note that in C64 mode the MMU completely disappears from the system's memory map. Note that the data out of the MMU is valid **only** on AEC high. This is necessary to avoid bus contention during a VIC cycle.



MMU Register Map

THE MEMORY MANAGEMENT UNIT (Continued)



**Configuration Register
Preconfiguration Register**

The Configuration Register

The **Configuration Register**, CR, controls the ROM, RAM, and I/O configuration of the C128 system. It is located at \$D500 in I/O space and at \$FF00 in system space. Some of the bits in this register are at times reflected by hardware lines MS₀ and MS₁ in C128 mode, depending upon how RAM and ROM have been set. These MS lines are used to inform the PLA about the type of memory in a particular address range. In C64 mode, MS₀ and MS₁ are always high, and the selection of RAM and ROM is done by the PLA using standard C64 banking methods. The MS lines are alternately referred to as ROMBANK lines. They will be referred to as MS lines in this section in the interest of simplicity.

In C128 mode, bit 0 controls whether an I/O space, \$D000 — \$DFFF, or a ROM/RAM access occurs. A low will select I/O, a high will enable some kind of ROM/RAM access, the nature of which is controlled by other bits in this register. The value of this bit is stored in a pre-latch, until the fall of the clock, in order to prevent its changing in an unstable situation. Note that when not I/O space, the ROM/RAM access is controlled by the defined ROM Hi configuration bits, which are described later. This bit resets to 0. When the I/O bit is low, MMU registers \$D500 to \$D50B will assert themselves. When the bit is high, these registers disappear from the memory map. MMU registers \$FF00 to \$FF04 are always available in C128 mode. The hardware line I/OSE always reflects the polarity of this bit when in C128 mode. In C64 mode the I/OSE line, the hardware line driven by this bit, is completely ignored by the PLA, and the MMU is never asserted, even when C64 I/O is enabled. The C64 method of selecting I/O via HIROM and CHAREN takes over here. The I/O hardware line remains in its set state when in C128 mode, even though it has no effect in this mode.

Bit number 1 controls processor access to ROM low space, \$4000 — \$7FFF, in C128 mode. If the bit is high, the area will appear as RAM, and a RAM access, CAS enable, will be generated to the appropriate RAM bank, which is determined by other bits in this register. If low, system ROM will be located in the space. This bit affects the memory status lines MS₀ and MS₁ which are decoded by the PLA to generate ROM chip selects. Selecting ROM here will drive both memory status lines low when the processor address falls within the specified low space range. This bit resets low to include the C128 Basic Low ROM. Of course in C64 mode, this bit is ignored.

The next two bits, bits 2 and 3, determine for C128 mode the type of memory that will be located in the mid space, \$8000 — \$BFFF. If they are both low, system ROM will be located here. If bit 2 alone is high, internal function ROM is located here. External function ROM appears for bit 3 being alone high, and RAM appears, along with the proper CAS generation, for both bits set high. These bits also affect the hardware memory access lines. When in the aforementioned mid block address range, MS₀ will reflect the status of bit 3, and MS₁ will reflect the status of bit 2. These bits both reset low to start out with Basic Hi. C64 mode ignores these bits.

THE MEMORY MANAGEMENT UNIT (Continued)

Bits 4 and 5 determine the contents of the Hi block, \$C000 — \$FFFF, for C128 mode, and have no effect on C64 mode. As with the mid space, both bits zero will set up system ROM, bit 4 high will set up internal function ROM, bit 5 high will set up external function ROM, and both bits high will set up RAM. Note that the I/O configuration bit, when set for I/O space, will leave the area from \$D000 to \$DFFF as I/O space, regardless of the values of these bits. If not set for I/O space, \$D000 to \$DFFF will contain the character ROM if the ROM chosen is System ROM. As with the other ROM selection bits, these bits are reflected by the memory status lines when this region of address is accessed. Bit 5 corresponds to MS₀ and bit 4 to MS₁. Both of these bits reset to low to permit Kernal and Character ROM to power up in this address space. Note that there is always a hole in high ROM during C128 mode for the MMU registers at \$FF00 to \$FF04. This hole is brought about by holding both MS lines high and both CAS enable lines high. These bits are ignored in C64 mode.

Finally, bit 6 controls the RAM bank selection. When low, it will select bank 0 by dropping CAS₀. When high, it will select bank 1 by dropping CAS₁. Bit 7 is unassigned at the present, left for future expansion. Note that a RAM share status that is non-zero will override the normal CAS enable generation to provide CAS₀ for all shared memory. Also, note that when the proper CAS enable is generated, any area of memory, even if that area does not have its ROM bank bits set for RAM, is accessed. It is up to the PLA to block CAS for a read from ROM. This allows RAM bleed through on a write to ROM. For any access to the MMU registers from \$FF00 to \$FF04, in any C128 mode configuration, both CAS enable lines and both MS lines will be high. Note that in C64 mode, the bank used follows the same rules as in C128 mode, though of course banks cannot be changed once in C64 mode.

The Preconfiguration Mechanism

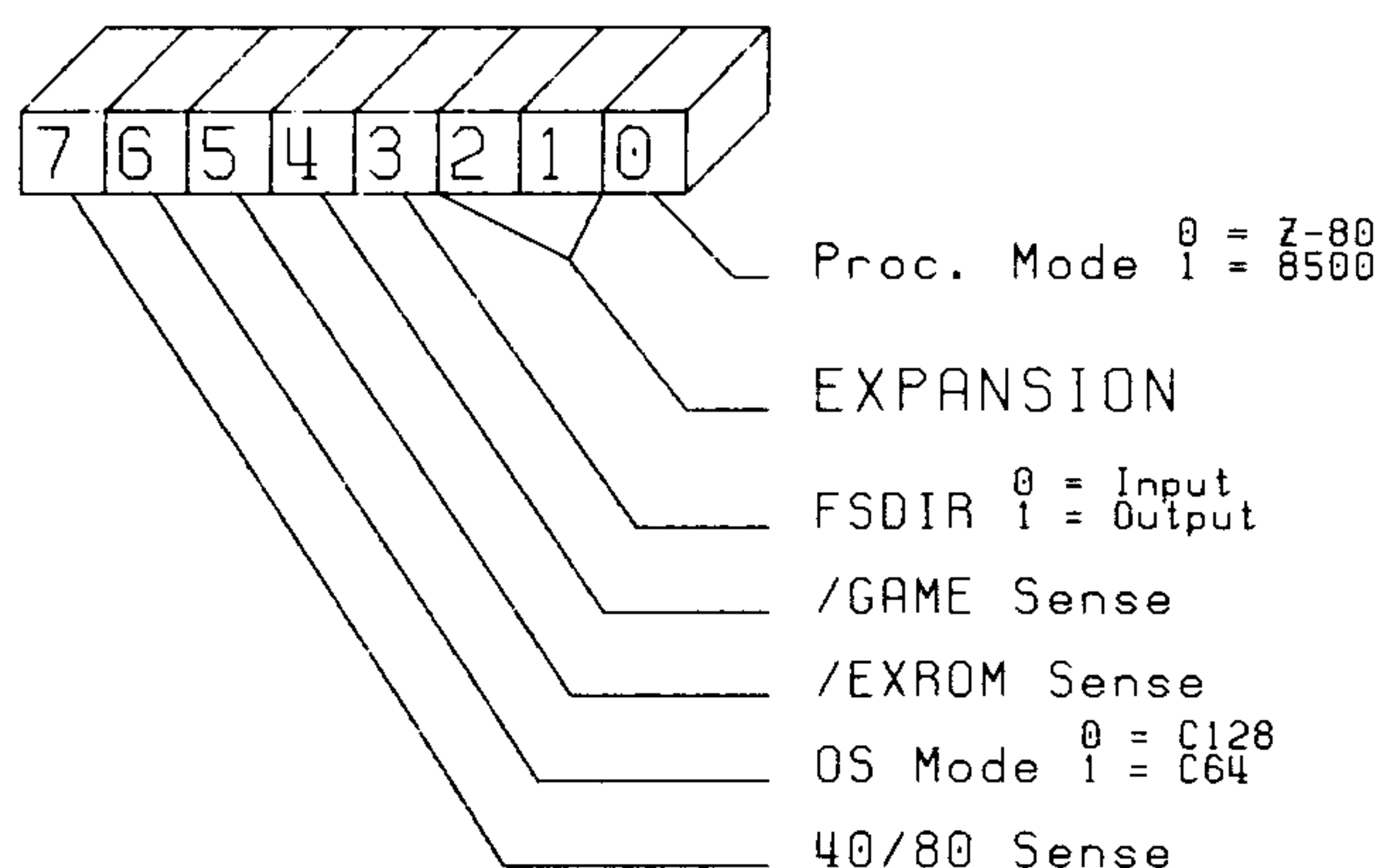
The Preconfiguration Mechanism is a feature of the MMU that allows the Configuration Register to be loaded with one of several memory configurations, with a minimum of time and memory on the part of the user. The scheme makes use of two sets of registers, the **Preconfiguration Registers** and the **Load Configuration Registers**.

The Preconfiguration Registers (PCRA — PCRD) are used to store several different memory configurations that may be accessed with a single store instruction. The format of each preconfiguration register is the same as for the Configuration Register but, when a value is stored to a preconfiguration register, no immediate effect takes place. They occupy I/O space from \$D501 to \$D504. These registers always reset to all zeros.

Load Configuration Registers (LCRA — LCRD) directly correspond with the preconfiguration registers on a one-to-one basis. A write to a Load Configuration Register causes the contents of the corresponding Preconfiguration Register to be transferred to the Configuration Register. A read of any Load Configuration Register returns the value of its corresponding Preconfiguration Register. Load Configuration Registers are located in system space from \$FF01 to \$FF04. Neither the Load Configuration Registers nor the Preconfiguration Registers have any effect in C64 mode. These registers reset to all zeros. Note that these, and the configuration register at \$FF00, will **always** be available, completely independent of the ROM, RAM, or bank configuration defined for Hi ROM space. Any address in this range will cause the MMU to force both memory status lines and both CAS enable lines high.

THE MEMORY MANAGEMENT UNIT (Continued)

MCR=\$D505



Mode Configuration Register

The Mode Configuration Register

The control of the current system mode is governed by the **Mode Configuration Register, MCR**. It controls which processor, 8502 or Z-80, and which operating system mode, C64 or C128, is currently in operation, and handles other overhead of the different operating modes. This register is located in the I/O space at \$D505.

Several of the bits in this register function as bidirectional ports, including the FSDIR, GAME, EX-ROM, and 40/80 bits. This type of port functions like an output port. If a value is written to the port, its hardware line will reflect that value written, and a read will return that value. The only exception to this is if an external source is pulling down the corresponding port line. When pulled down, a read of the port will return a low. Once the external source has been removed, a read will return the value previously stored. Thus, as an input, the port can be driven low, but not high, by an external source. Under each bit description, both the input and output functions of each port bit will be described in detail.

The first bit, bit 0, controls which processor is enabled. It is reflected by the output line **Z80EN**. When low, it indicates that the processor is the Z-80. This is the reset configuration, and will cause the Z-80 processor to be active and all accesses to memory to follow the Z-80 mapping rules. In Z-80 mode, any address to RAM bank 0 in the range from \$0000 to \$0FFF will be translated to the corresponding address in the range from \$D000 to \$DFFF, where the Z-80 CP/M BIOS physically exists in System ROM. Additionally, the memory status lines MS₀ and MS₁, will reflect system ROM (both low) for accesses in the range of the BIOS, and the page zero and page one offset pointers will be disabled. RAM can still be banked by the CR A₁₆ bit, which controls CAS₀ and CAS₁. When in bank one, the BIOS ROM disappears, allowing the RAM from \$0000 to \$FFFF to be used by the system, and enabling the page zero and one offset pointers.

A change to this processor select bit is held in pre-latch until a clock transition, in order to prevent processor changing in the midst of an instruction execution. Bringing this bit high will cause the Z-80 to be disabled and the 8502 to take over. Upon system power up, the Z-80 will turn itself off and bring up C128 mode by setting this bit and allowing the 8502 to take over.

Bits 1 and 2 are unused, but are reserved for future expansion as possible port lines. Currently, they will return high if read, and cannot be written to.

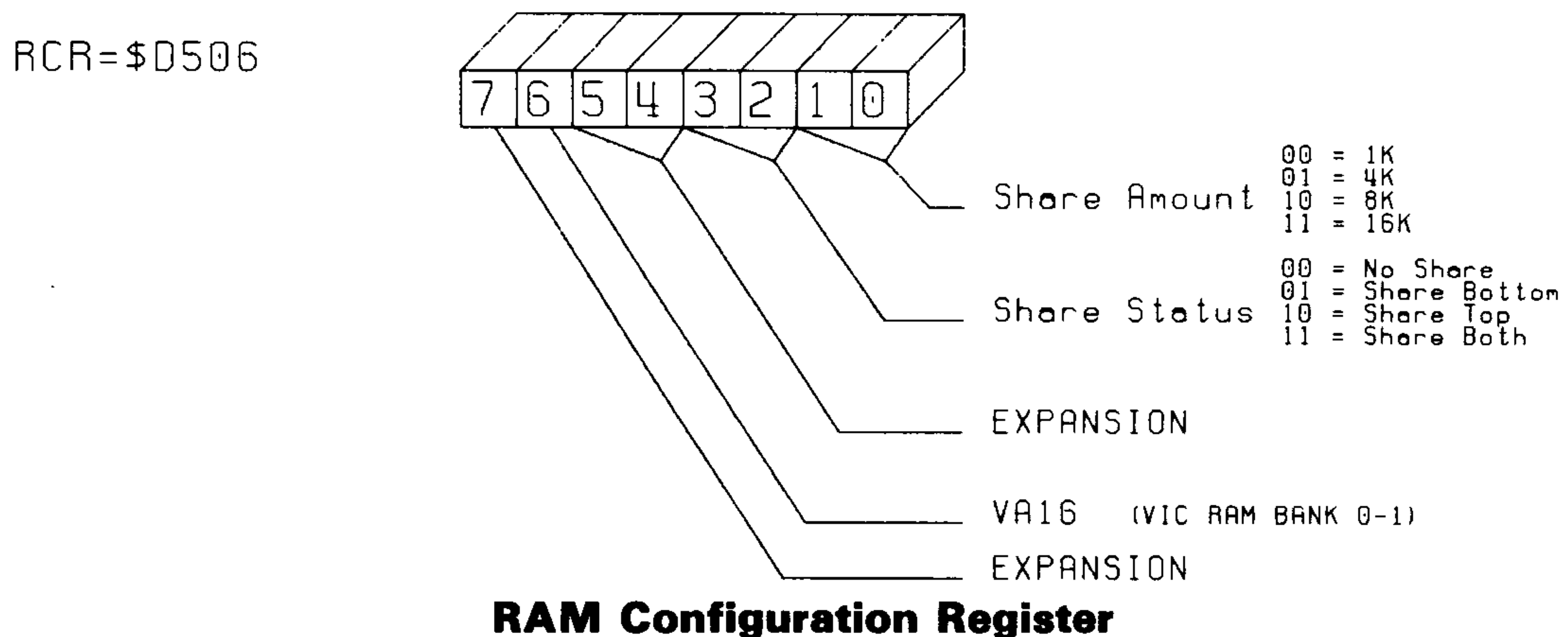
Bit 3 is the **FSDIR** control bit. It is used as an output to control the fast serial disk data direction buffer hardware, and as an input to sense a fast disk enable signal. This bit is a bidirectional port bit as explained above, and its hardware line is called **FSDIR**.

THE MEMORY MANAGEMENT UNIT (Continued)

Bits 4 and 5 are the $\overline{\text{GAME}}$ and $\overline{\text{EXROM}}$ sense bits, respectively, which are implemented as bidirectional ports as explained above. As inputs, they directly reflect the hardware cartridge control lines $\overline{\text{GAME}}$ and $\overline{\text{EXROM}}$ as used in C64 mode. C128 cartridges do not use $\overline{\text{EXROM}}$ and $\overline{\text{GAME}}$, so if they are detected in C128 mode, a C64 cartridge is present and C64 mode should be asserted. They have no dedicated C128 function.

The operating system mode is set by bit 6. This bit is cleared to zero upon reset and its presence enables all MMU registers and other C128 features, as well as asserting the C128 control line in hardware. Setting this bit removes the MMU from the memory map and sets the system up in C64 mode. Note that the C128 MS3 hardware line reflects a logical inversion of the level of this bit.

Bit 7 is used to detect the status of the screen mode switch, as presented in hardware to the Sense40 column pin. If this bit is high, the 40/80 column switch is open, if low, the switch is closed. The display mode will be set according to a software interpretation of this bit. This bit is a bidirectional port bit, but its output function is undedicated at this time.



The RAM Configuration Register

The **RAM Configuration Register** sets up the RAM segmenting parameters for both the processor and the block pointer for the VIC chip. This register is located in the I/O space at \$D506.

Bits 0 and 1 function together to determine the size of the RAM to be shared between banks, assuming that sharing is enabled. With common RAM, the RAM bank bits of the configuration register are basically overridden, as the selected bank of RAM will be used for the non-common areas, while bank 0 will be used for the specified common areas. ROM and I/O block configuration bits, however, are still important. If the value of the bits together is 0, then 1K of RAM is held common. If the value is 1, then 4K; 2, then 8K; 3, then 16K. These bits have no effect in C64 mode, and the reset value of both bits is defined to be zero.

Bits 2 and 3 function to determine how and if RAM is kept common. If both are low, no sharing takes place. If bit 2 is set, the bottom RAM is shared. If bit 3 is set, the top RAM is shared. Both may be set at the same time for sharing both top and bottom memory. The reset configuration sets both of these bits zero, such that no common memory is present.

The next two bits, numbers 4 and 5, are not used in this MMU. They are available for possible future expansion. They read low, and cannot be written to.

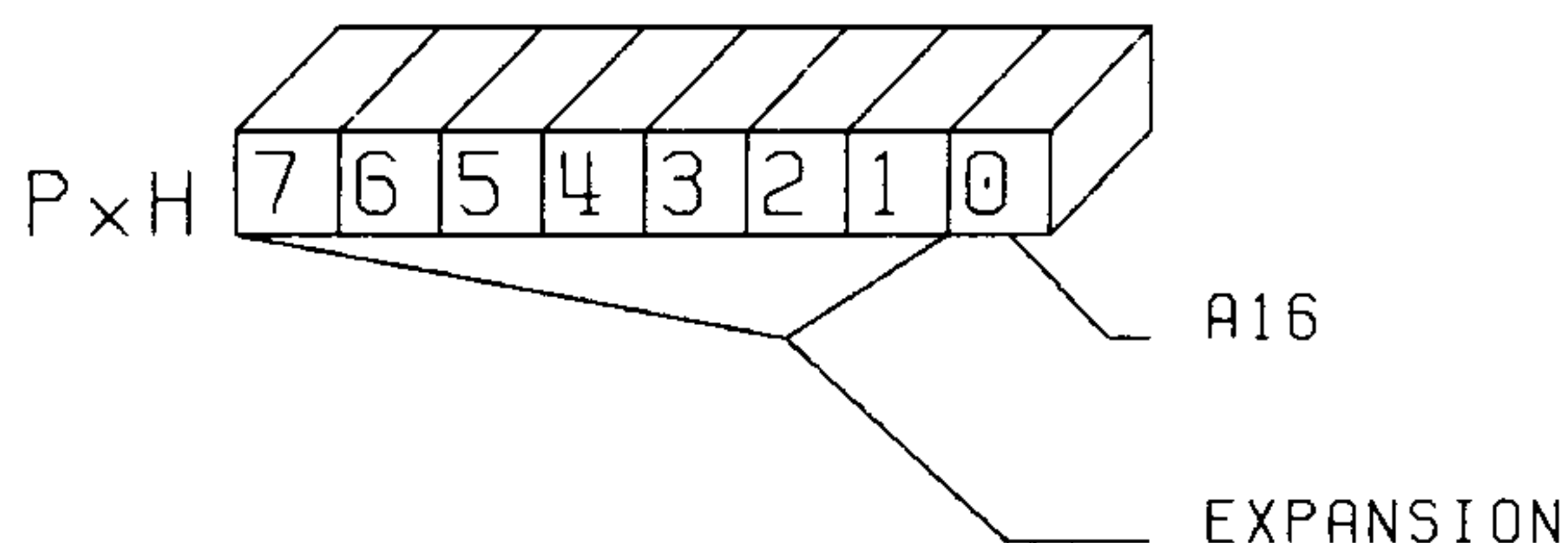
THE MEMORY MANAGEMENT UNIT (Continued)

Bit 6 functions as a RAM bank pointer for VIC. It is used to drive CAS₀ low when set low or CAS₁ low when set high, thus selecting either RAM bank 0 or RAM bank 1 for the VIC, independently from the processor bank. When in 2 MHz mode the 80-column chip takes over, causing the VIC to be disabled. This disabling is affected by the VIC chip itself holding AEC constantly high, and thus is not directly effected by actions of the MMU. Note that since a VIC cycle is detected by AEC low, that any DMA will put the MMU into VIC configuration, as it too brings AEC low. This allows independent bank selection for DMAs in 80 column mode.

Bit 7 is currently unused.

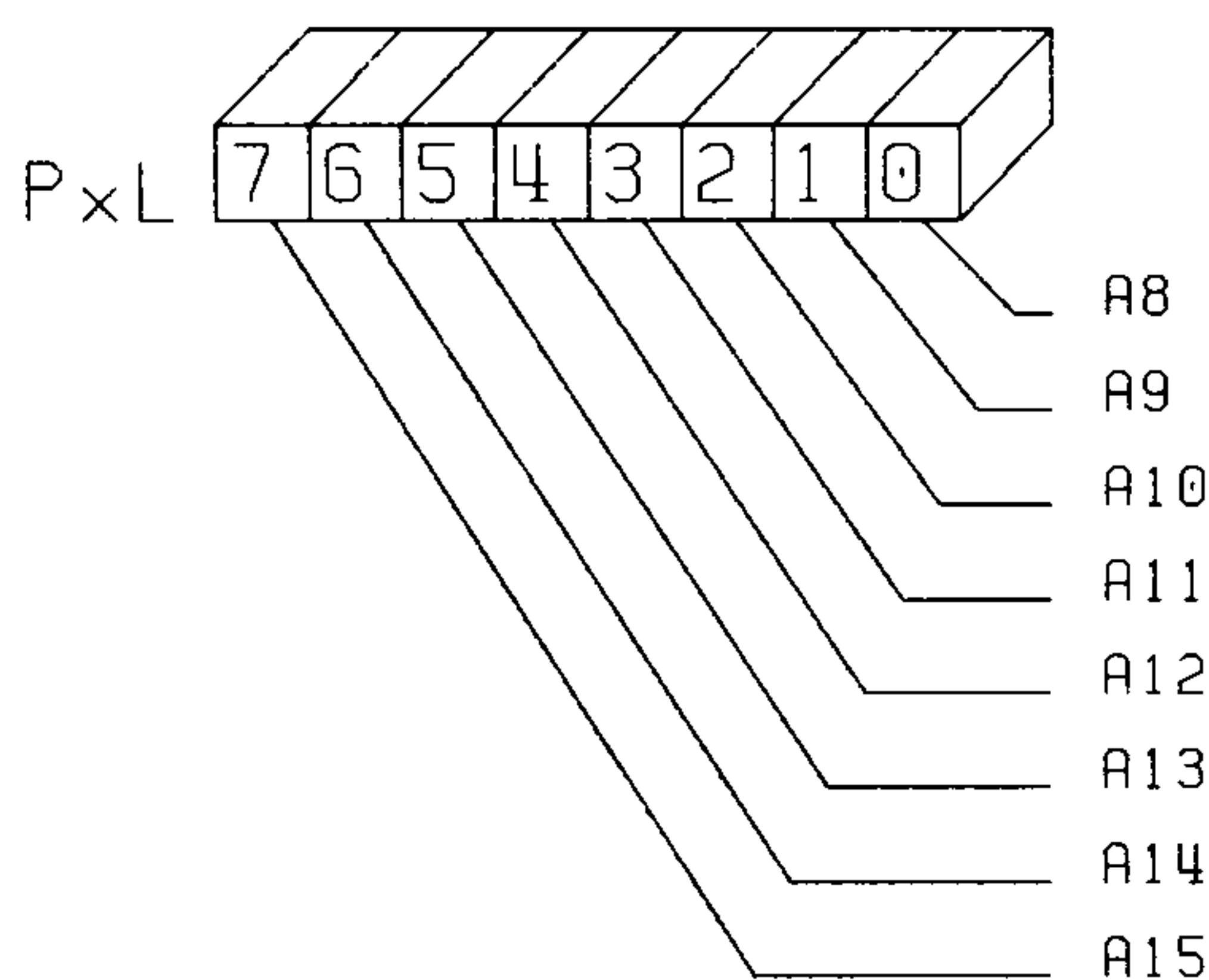
P0H=\$D508

P1H=\$D50A



P0L=\$D507

P1L=\$D509



Page Pointer

The Page Pointers

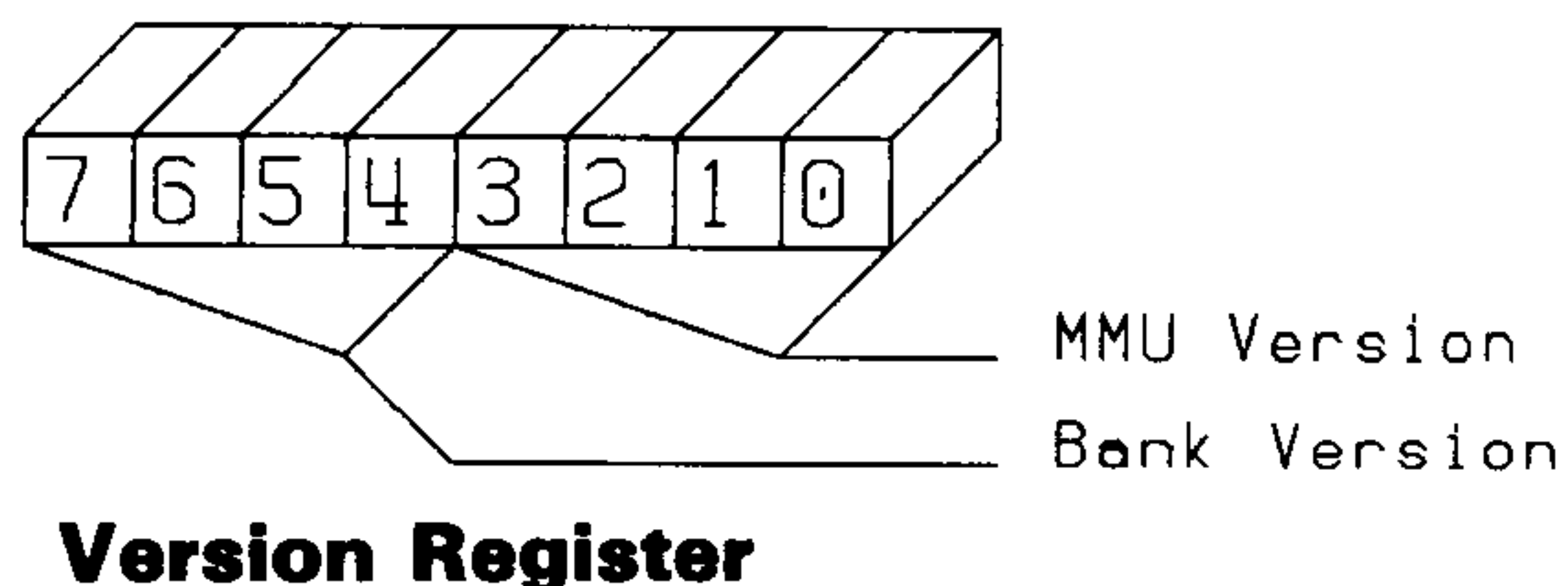
The page pointers are four registers that allow independent relocation of pages zero and one, when running under either processor. These are especially useful when running under the 8502 as they help to remove some of the zero page and stack size limitations normally associated with 6502 family processors.

For zero page relocation, the MMU provides the **Page Zero Pointer High (P0H)** and **Page Zero Pointer Low (P0L)** registers. Bit 0 of the P0H register corresponds to translated addresses TA₁₆ for any zero page access, \$0000 — \$00FF, controlling the generation of CAS₀ or CAS₁ depending on whether it is low or high. The remaining bits are currently unused, and will always return zero. These bits override the RAM bank bits, the ROM block, and the I/O block bits to determine which physical page appears as zero page for all zero page accesses. A write to the P0H register is stored in pre-latch until a write to the P0L register occurs. Bits 0 to 7 of the P0L correspond to Translated Addresses TA₈ to TA₁₅ for any zero page access, thus relocating the zero page. Any access to the area that has become the relocated zero page will be switched back to the original zero page if that area is mapped as RAM. If mapped as ROM, then the reverse mapping is not done, allowing access to the ROM. A write to this register sets up the zero page transfer, which can occur as soon as the next low clock cycle. Register P0L is located in the I/O space at \$D507, while register P0H is located at \$D508.

THE MEMORY MANAGEMENT UNIT (Continued)

The registers for page one relocation, the **Page One Pointer High** (P1_H) and the **Page One Pointer Low** (P1_L) do for page one essentially what P0_H and P0_L do for the zero page. The functions and bit correspondences are exactly the same. P1_L is located in the I/O space at \$D509 and P1_H at \$D50A. Note that both register pairs are initialized upon reset to reflect true page zero and true page one access for the 8502 processor. Note that these registers continue to take effect in Z-80 mode, as well as in 8502 mode, when set to bank one. When set to bank zero, they are disabled to provide true Z-80 BIOS access.

VR=\$D50B

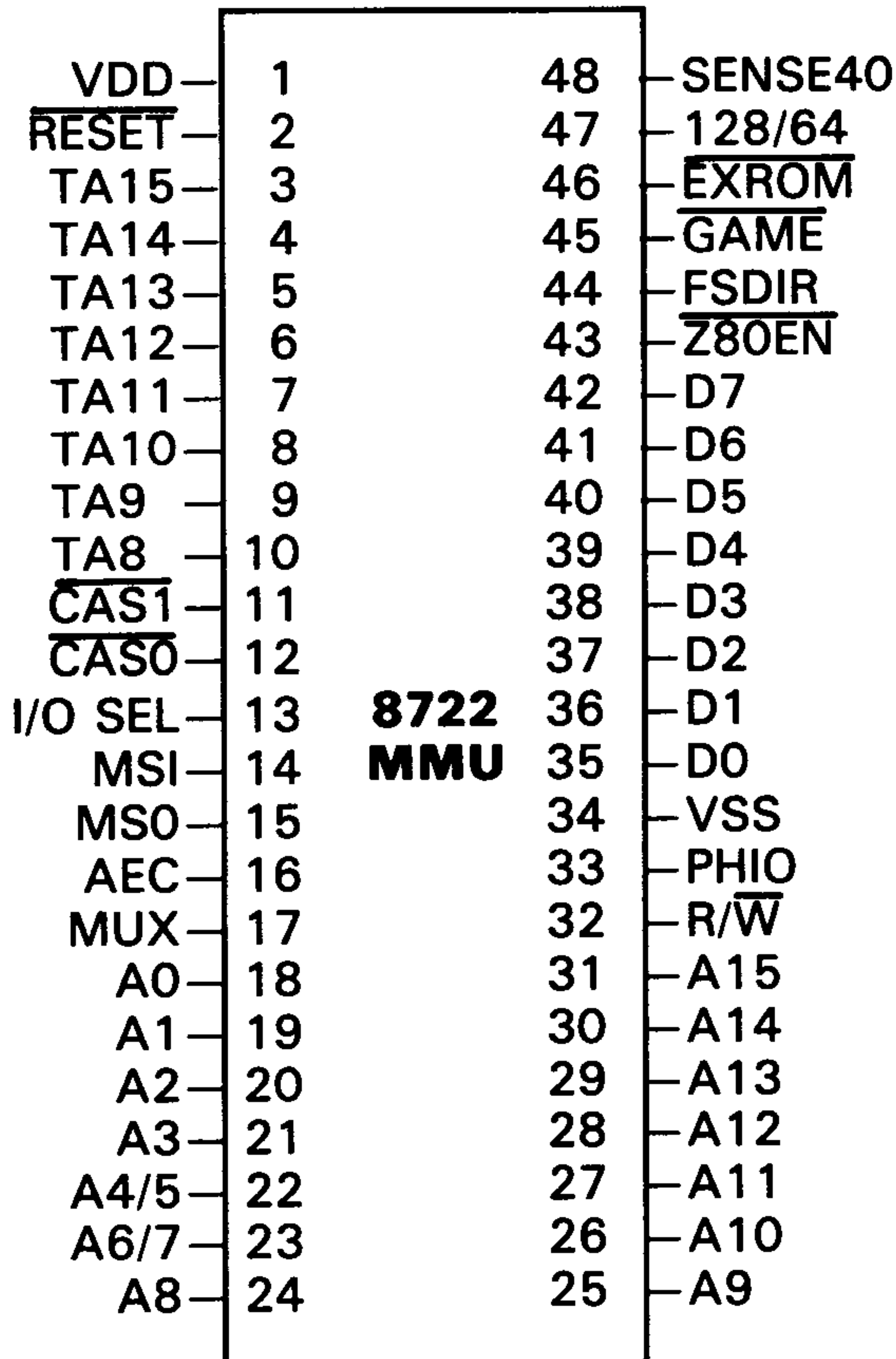


System Version Register

The final register is the **System Version Register**, which is located at \$D50B in the I/O block. This register is a read-only register that returns a code containing the version of the MMU and the size and capability of the system's memory. The lower nybble, bits 0 through 3, contain the version of the MMU in the system. The upper nybble, bits 4 through 7, contains a code relating the number of memory blocks available in the system. This allows software to compensate for any later systems with more available memory, and should make it quite simple for the current C128 to remain compatible with any software written in the future for an expanded C256, etc. system. The initial C128 will read a 2 here, indicating two 64K blocks are available. A zero in this nybble would indicate sixteen 64K blocks.

THE MEMORY MANAGEMENT UNIT (Continued)

310389 8722 MEMORY MANAGEMENT UNIT



1	VDD	5VDC input.
2	RESET	System Reset. This input initializes internal registers on a power up or hardware reset.
3-10	TA8-TA15	Translated Address outputs. Tri-stated for VIC cycles during AEC, provides translated physical address for use on the Multiplexed Address Bus and the Shared Static Bus. TA ₁₂ to TA ₁₅ are each defined to have an internal, depletion mode pullup with an equivalent resistance of 3.3KΩ. TA ₈ to TA ₁₁ each go tristate during VIC time (AEC low). Column Address Strobe. The CAS output lines control DRAM banking. CAS ₀ enables the first bank of 64K, CAS ₁ enables the second bank of 64K.
11,12	CAS ₀ , CAS ₁	
13	I/O SEL, MS ₂	I/O Select, Memory Status 2. This output is identified by both identifiers. It is used to select memory mapped I/O in C128 mode and is also known as MS ₂ . In C128 mode, this line always reflects the polarity of the I/O bit. It is ignored by the PLA in C64 mode, and remains high throughout C64 mode.
14,15	MS ₀ , MS ₁	Memory Status 0, 1. Also called ROMBANK ₀ and ROMBANK ₁ , these outputs control ROM banking for all ROM slots. These lines are used to decode ROM bank selection for any ROM access in C128 mode. If they are both low, a system ROM has been selected. If MS ₁ is high alone, then a built in function ROM has been selected. If MS ₀ is alone high, then an external function ROM has been selected. Finally, if both are high, the RAM that occupies the particular slot has been selected. In C64 mode the PLA completely ignores these lines.
16	AEC	The Address Enable Control indicates whether the 8502 processor or the VIC has access to the shared bus. When low, VIC or an external DMA has the bus and VA ₁₆ and VA ₁₇ have the processor bus, and no pointer or BIOS translation takes place.
17	MUX	The Memory Multiplex signal, used to clock various sections of the MMU.
18-31	A0-A15	A0 - A3, A8 - A15: Addresses from the microprocessor. Used to derive chip selects as well as multiplexed address lines. A4/5, A6/7: Combined addresses from the microprocessor. Used along with simple addresses, combined in this fashion to lower the pin count of the MMU.
32	R/W	R/W: System Read/Write control line. This input is high for a processor read, low for a processor write.
33	PHIO	Presystem clock. Used for early transition of gated signals on write operations. Processor address is valid on the rising edge and data is valid on the falling edge.
34	VSS	System Ground.
35-42	D ₀ -D ₇	Data inputs from the microprocessor. Used for writing to internal registers.
43	Z80EN	This output is used to enable the Z80 processor and disable the normal operation of the 8502 processor. It goes low to indicate Z-80 mode, high for all other modes.
44	FSDIR	Fast Serial Direction. This port is a bidirectional line that in output mode controls the data direction of the fast serial disk interface.
45	GAME	Game ROM Enable. This signal is used to sense the GAME line on the expansion connector in C64 mode and as the color RAM bank control line in C128 mode.
46	EXROM	External ROM Enable. This signal is used to sense the EXROM line on the expansion connector in C64 mode and as an expansion control line in C128 mode.
47	128/64, MS ₃	This output directs the system to act in either C128 or C64 mode. It goes low to indicate C64 mode, high for C128 mode.
48	40/80	Sense 40/80. This bidirectional port is used in input mode to sense the 40/80 column switch. It detects whether or not the switch is closed. The output mode is not used at this time.