MACINTOSH HARDWARE MEMORY MAP

Burrell Smith and Brian Howard

13 April 1983

----- CONFIDENTIAL ---------- CONFIDENTIAL -----

1. INTRODUCTION

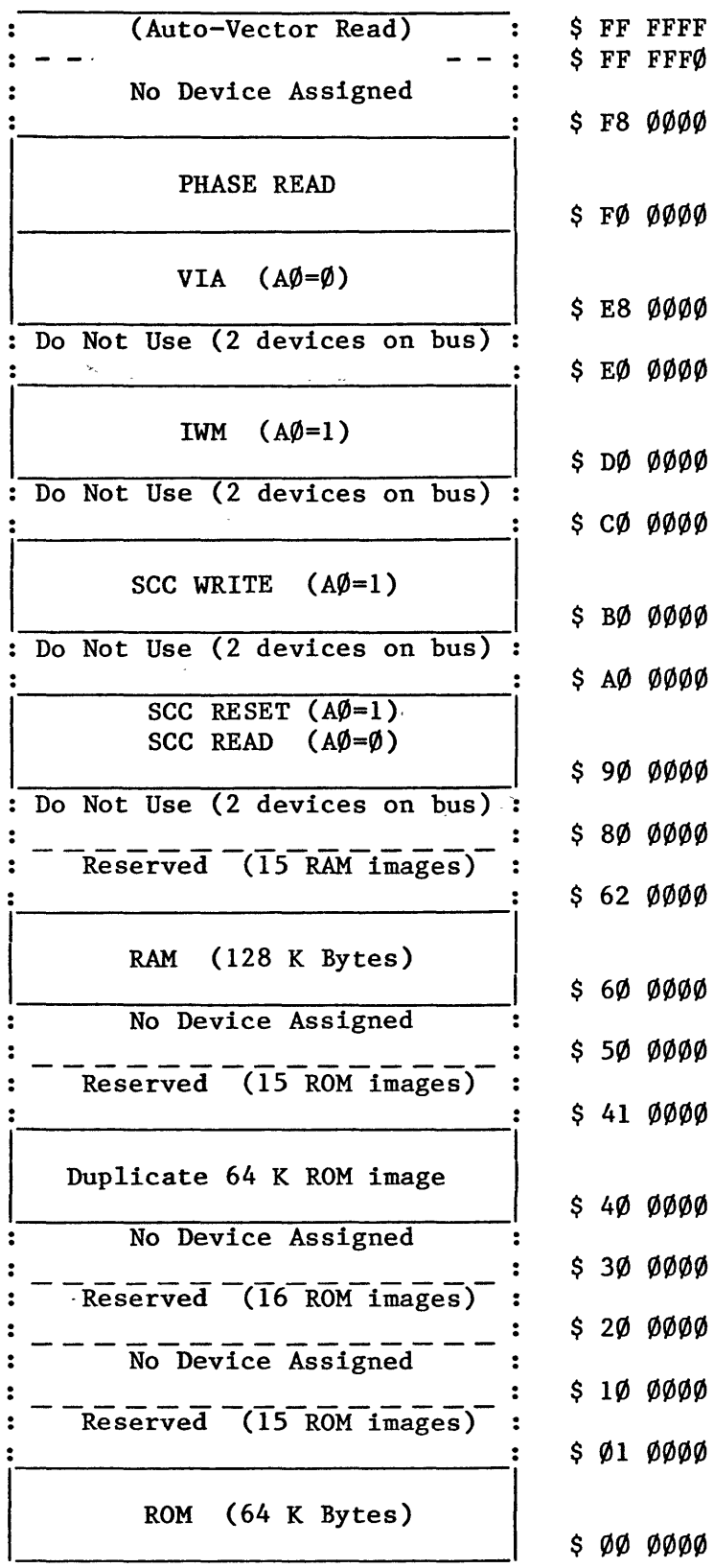
The principle portions of Macintosh's memory consist of volatile read/write memory (RAM) and permanent read-only memory (ROM). In addition to RAM and ROM, three input/output devices are also selected using address lines, so that they appear to occupy portions of the Macintosh memory space. These devices are the 6522 Versatile Interface Adapter (VIA), the 853Ø Serial Communications Chip (SCC), and the disk interface chip (IWM).

When the Macintosh is first turned on, ROM appears at the bottom (lowest addresses) portion of the memory space. This is useful for the ROM-stored software which starts the system running. After startup, the OVERLAY signal from the VIA is changed to a low (zero), mapping RAM into its normal place at the bottom of memory.

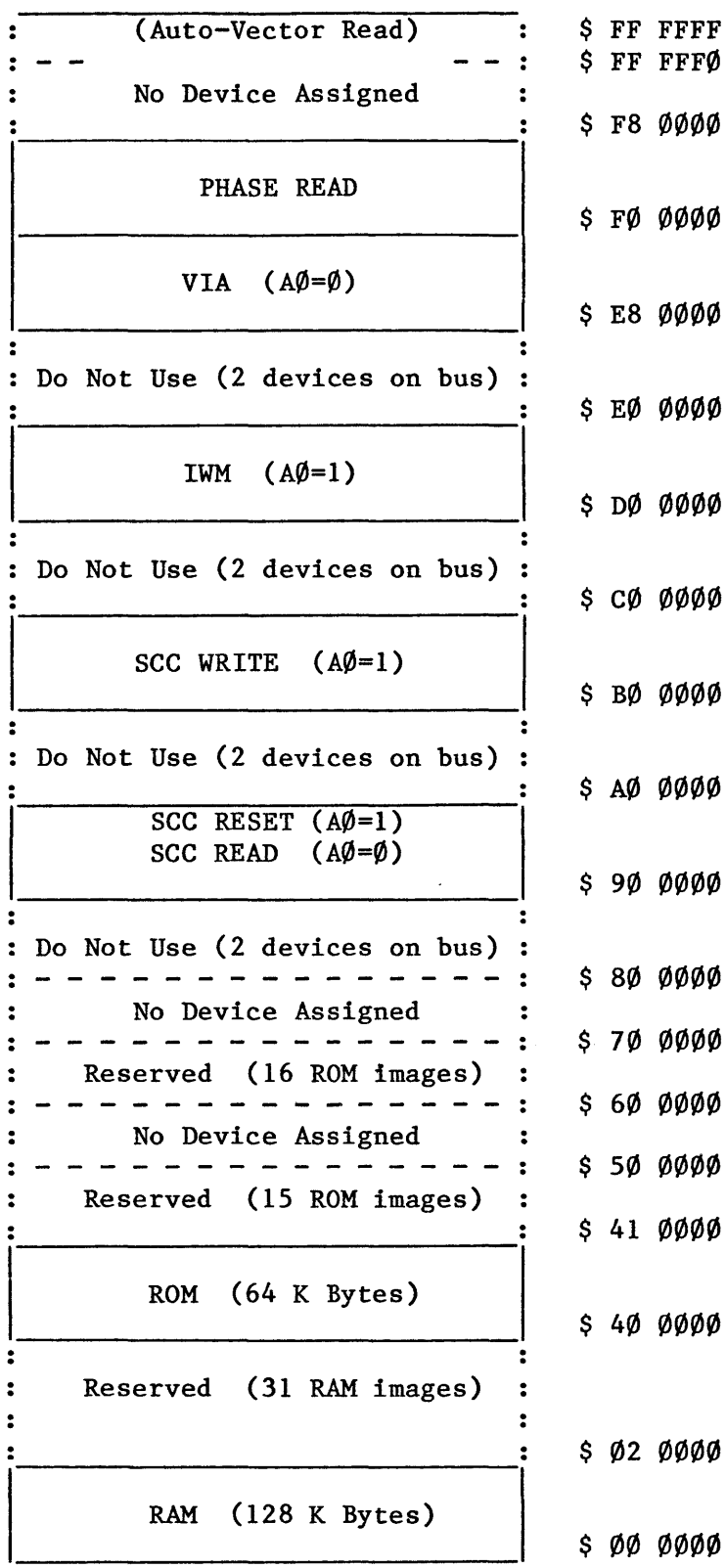
Selection of RAM, ROM, or other devices is done by from two to five of the highest-order address lines, A23-A19. The VIA and IWM also use the four address lines A12-A9 for further internal decoding and register selection, while the SCC uses the three lowest-order address lines A2-AØ for internal decoding.

In specifying "useful addresses" for most devices, unused address lines have been set high (to a one) to save a small amount of power and to improve noise immunity. Some address ranges are specified "Do Not Use" because they can select two devices simultaneously. While this does not cause any damage to the computer, data cannot be correctly transferred while these addresses are in use.

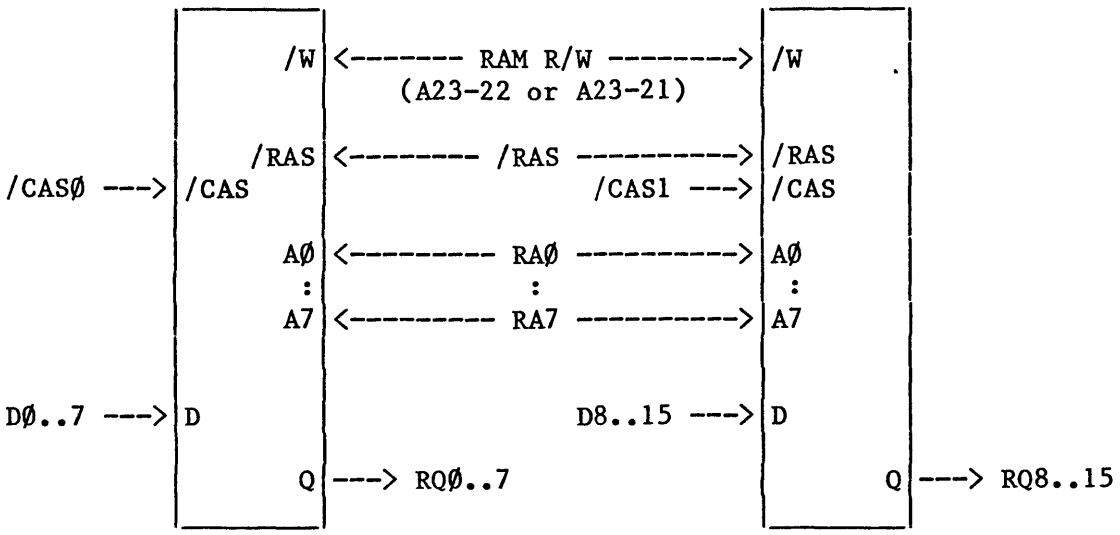
2. MEMORY MAP ON POWER-UP (OVERLAY = 1)



3. NORMAL MEMORY MAP (OVERLAY = Ø)



4. RAM



Eight 64k bit DRAMs
U5F...U12F

Eight 64k bit DRAMs
U5G...U12G

4.1 Address Decoding to Activate RAMs

Note: RAM is written when RAM R/W = Ø. RQØ-RQ15 are read onto the data bus when /RAM READ = Ø.

When RAM Addressed	Address Lines A23	A22	A21	A2Ø	Address Range
Startup: OVERLAY=1	Ø	1	1	X	$6ØØØØØ-$7FFFFF
Normal: OVERLAY=Ø	Ø	Ø	X	X	$ØØØØØØ-$3FFFFF

(Note: X indicates "don't care": either 1 or Ø)

4.2 Some Useful RAM Addresses

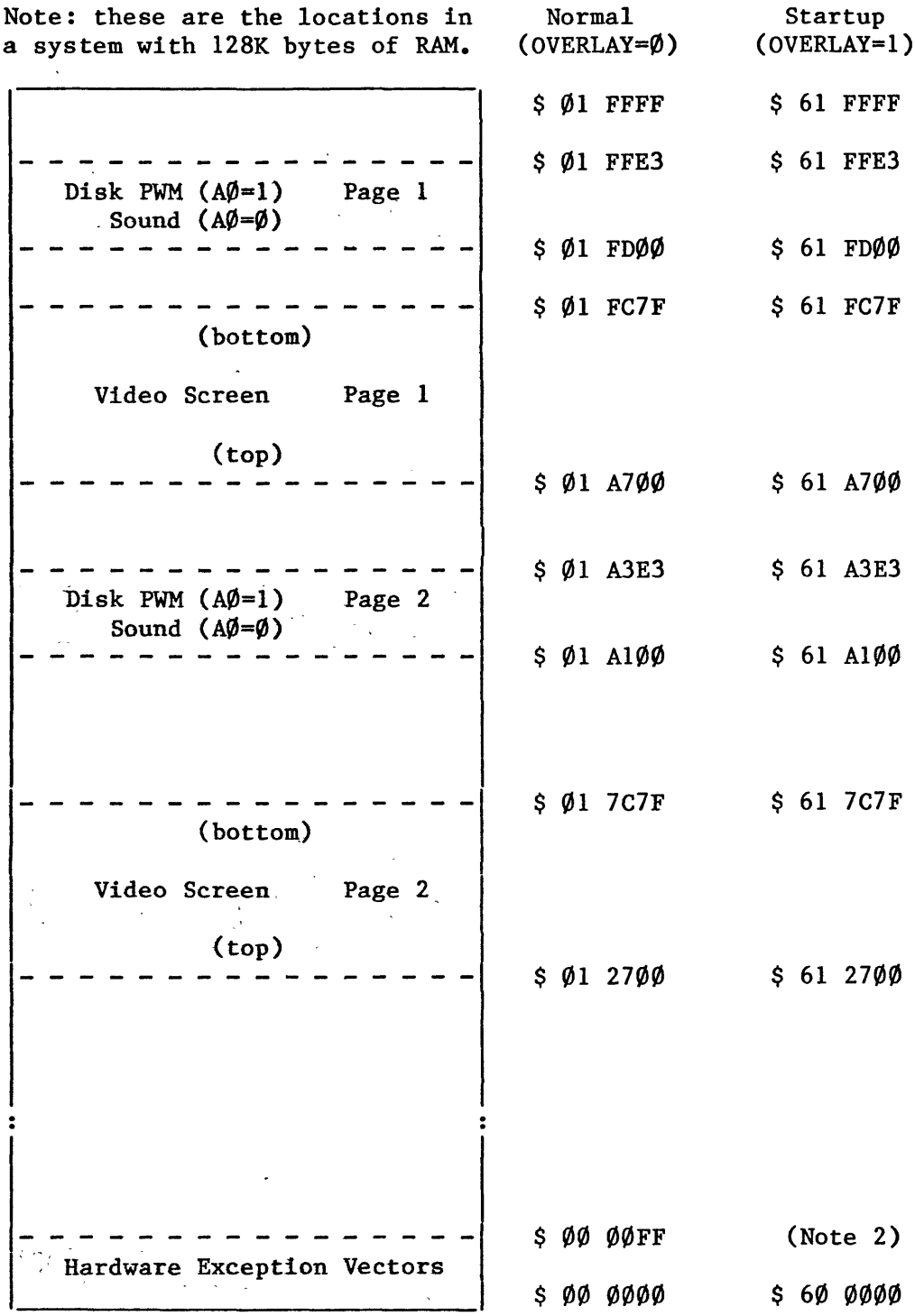
128K RAM Addresses	$ØØØØØØ - $Ø1FFFF	
Video Screen, Page 1	$Ø1A7ØØ (top) - $Ø1FC7F (bottom)	($558Ø bytes)
Video Screen, Page 2	$Ø127ØØ (top) - $Ø17C7F (bottom)	
Sound/PWM Buffer, Page 1	$Ø1FDØØ - $Ø1FFE3	($2E4 bytes)
Sound/PWM Buffer, Page 2	$Ø1A1ØØ - $Ø1A3E3	

(Note: Sound = high bytes, PWM = low bytes)

RAM Addresses during startup (OVERLAY = 1) $6ØØØØØ - $61FFFF

Video Screen, Page 1, during startup (OVERLAY = 1) $61A7ØØ - $61FC7F

4.3 More Detailed Map of RAM



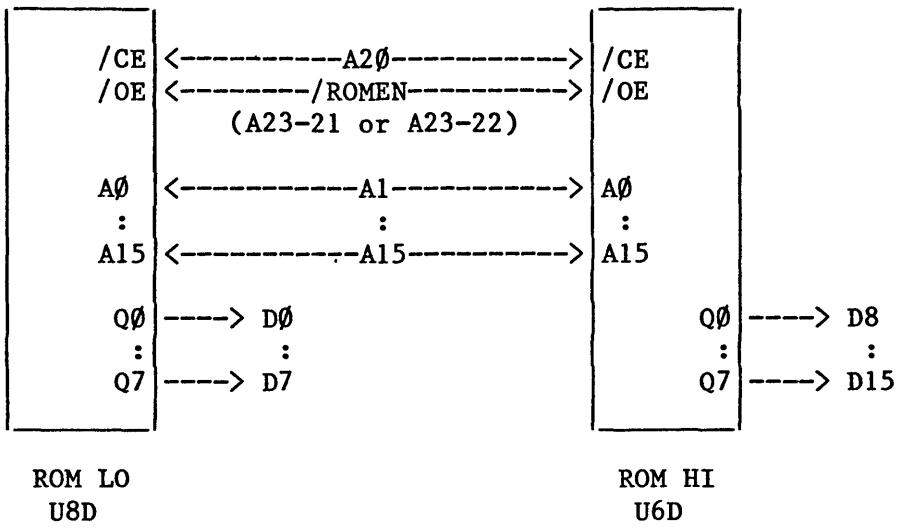
Map of RAM

Note 2: the hardware exception vectors are always at $ØØØØØØ-$ØØØØFF. This places them in the ROM address space during startup (OVERLAY = 1).

4.4 Hardware Exception Vectors (in RAM if OVERLAY = Ø)

Reset: Initial SSP	$ØØ ØØØØ
Reset: Initial PC	$ØØ ØØØ4
Bus Error	$ØØ ØØØ8
Address Error	$ØØ ØØØC
Illegal Instruction	$ØØ ØØ1Ø
Divide by Zero	$ØØ ØØ14
CHK Instruction	$ØØ ØØ18
TRAPV Instruction	$ØØ ØØ1C
Privilege Violation	$ØØ ØØ2Ø
Trace	$ØØ ØØ24
Line 1Ø1Ø Emulator	$ØØ ØØ28
Line 1111 Emulator	$ØØ ØØ2C
(Unassigned: Reserved)	$ØØ ØØ3Ø - $ØØ ØØ3B
Uninitialized Interrupt	$ØØ ØØ3C
(Unassigned: Reserved)	$ØØ ØØ4C - $ØØ ØØ5F
Spurious Interrupt	$ØØ ØØ6Ø
VIA Interrupt Auto-Vector	$ØØ ØØ64
SCC Interrupt Auto-Vector	$ØØ ØØ68
VIA+SCC (temp.) Auto-Vector	$ØØ ØØ6C
Interrupt Switch Auto-Vector	$ØØ ØØ7Ø
Int.Sw.+VIA Auto-Vector	$ØØ ØØ74
Int.Sw.+SCC Auto-Vector	$ØØ ØØ78
Int.Sw.+VIA+SCC Auto-Vector	$ØØ ØØ7C
TRAP Instruction Vectors	$ØØ ØØ8Ø - $ØØ ØØBF
(Unassigned: Reserved)	$ØØ ØØCØ - $ØØ ØØFF

5. ROM



5.1 Address Decoding to Activate ROMs

Note: ROM is activated whenever A2Ø=Ø and /ROMEN=Ø.

When ROM Addressed		Address Lines A23	A22	A21	A2Ø	Address Range	(Exceptions)
Startup: OVERLAY=1		1	Ø	X	Ø	$ØØØØØØ-$ØFFFFF, $2ØØØØØ-$2FFFFF	
	and	Ø	1	Ø	Ø	$4ØØØØØ-$4FFFFF	
Normal: OVERLAY=Ø		Ø	1	X	Ø	$4ØØØØØ-$4FFFFF, $6ØØØØØ-$6FFFFF	
Anytime: OVERLAY=X		1	Ø	X	Ø	$8ØØØØØ-$8FFFFF, $AØØØØØ-$AFFFFF	(Do not use: SCC is also on the bus)
	and	1	1	Ø	Ø	$CØØØØØ-$CFFFFF	(Do not use: IWM is also on the bus)

(Note: X indicates "don't care": either 1 or Ø)

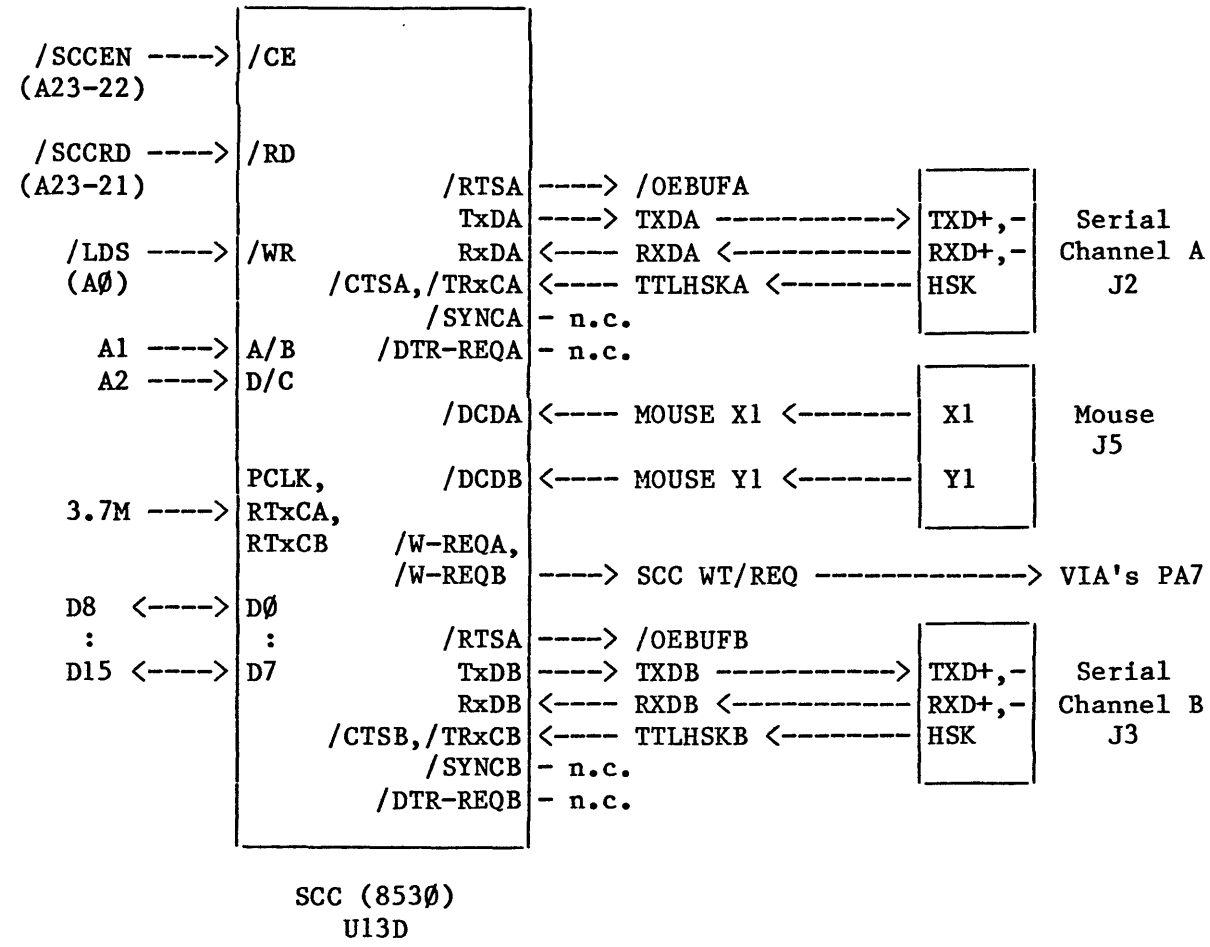
5.2 Useful ROM Addresses

Note: these addresses are for a system with 64K bytes of ROM.

Startup ROM Addresses (OVERLAY = 1) $ØØØØØØ - $ØØFFFF
or $4ØØØØØ - $4ØFFFF (duplicate ROM image)

Normal ROM Addresses $4ØØØØØ - $4ØFFFF

6. SCC



6.1 Address Decoding to Activate SCC

Device Addressed	Address Lines A23	A22	A21	A2Ø	But Do Not Use	Address Range (Exceptions)
SCC Read (/SCCEN=Ø, /SCCRD=Ø)	1	Ø	Ø	X	1ØØØ: ROM is also on bus	$8ØØØØØ-$9FFFFF (But do not use $8ØØØØØ-$8FFFFF)
SCC Write (/SCCEN=Ø, /SCCRD=1)	1	Ø	1	X	1Ø1Ø: ROM is also on bus	$AØØØØØ-$BFFFFF (But do not use $AØØØØØ-$AFFFFF)

(Note: X indicates "don't care": either 1 or Ø)

6.2 Further SCC Address Decoding

	A2	A1	AØ
/LDS = Ø	X	X	1
/LDS = 1	X	X	Ø
Channel A	X	1	X
Channel B	X	Ø	X
Data Register	1	X	X
Control Register	Ø	X	X

READ
Byte Read: AØ = Ø
The SCC uses the upper byte of the data bus, so use AØ = Ø for reading the SCC. This sets /LDS high and the CPU reads data from D8-D15.

RESET
Byte Read: AØ = 1
A byte access to any SCC READ address with AØ = 1 sets /LDS and /SCCRD both low. This resets the SCC.

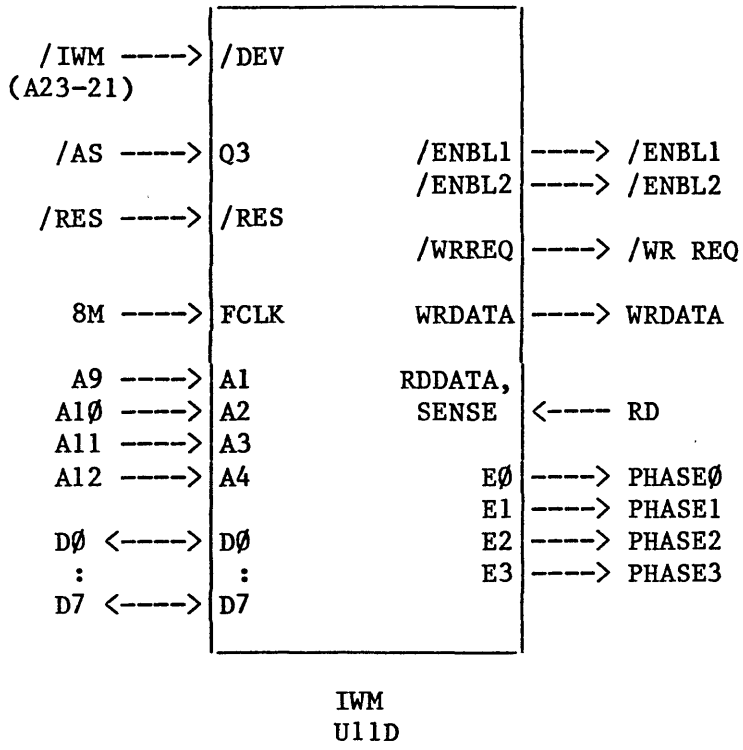
WRITE
Byte Write: AØ = 1
This uses a special feature of the 68ØØØ CPU: a write to the lower byte of the data bus (AØ=1) also places the same data on the upper byte of the data bus. /LDS is set low and the CPU writes the same byte of data to DØ-D7 and to D8-D15.

PHASE ADJUST
Word Read or Word Write
Normal accesses to the SCC are byte accesses. A word access to the SCC adjusts the phase of the computer's high-frequency timing signals by 128 nS. See section 9.3, PHASE READ, to determine whether the timing is in phase or out of phase.

6.3 Some Useful SCC Addresses

Channel A: Write to data register	$BFFFFF
Read from data register	$9FFFFE
Channel B: Write to data register	$BFFFFD
Read from data register	$9FFFFC
Channel A: Write to control register specified in Write Register Ø	$BFFFFB
Channel A: Read from control register specified in Write Register Ø	$9FFFFA
Channel B: Write to control register specified in Write Register Ø	$BFFFF9
Channel B: Read from control register specified in Write Register Ø	$9FFFF8
Reset SCC	$9FFFFF
Adjust timing phase: word read from	$9FFFFE

7. IWM

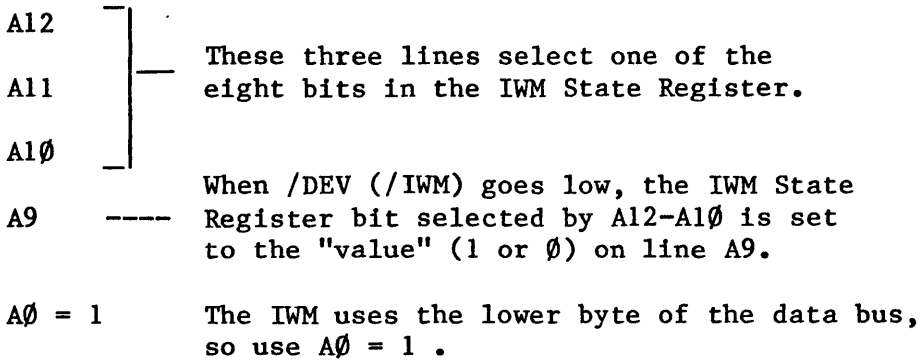


7.1 Address Decoding to Activate IWM

Device Addressed	Address Lines A23	A22	A21	A2Ø	But Do Not Use	Address Range (Exceptions)
IWM (/IWM=Ø)	1	1	Ø	X	11ØØ: ROM is also on bus	$CØØØØØ-$DFFFFF (But do not use $CØØØØØ-$CFFFFF)

(Note: X indicates "don't care": either 1 or Ø)

7.2 Further IWM Address Decoding

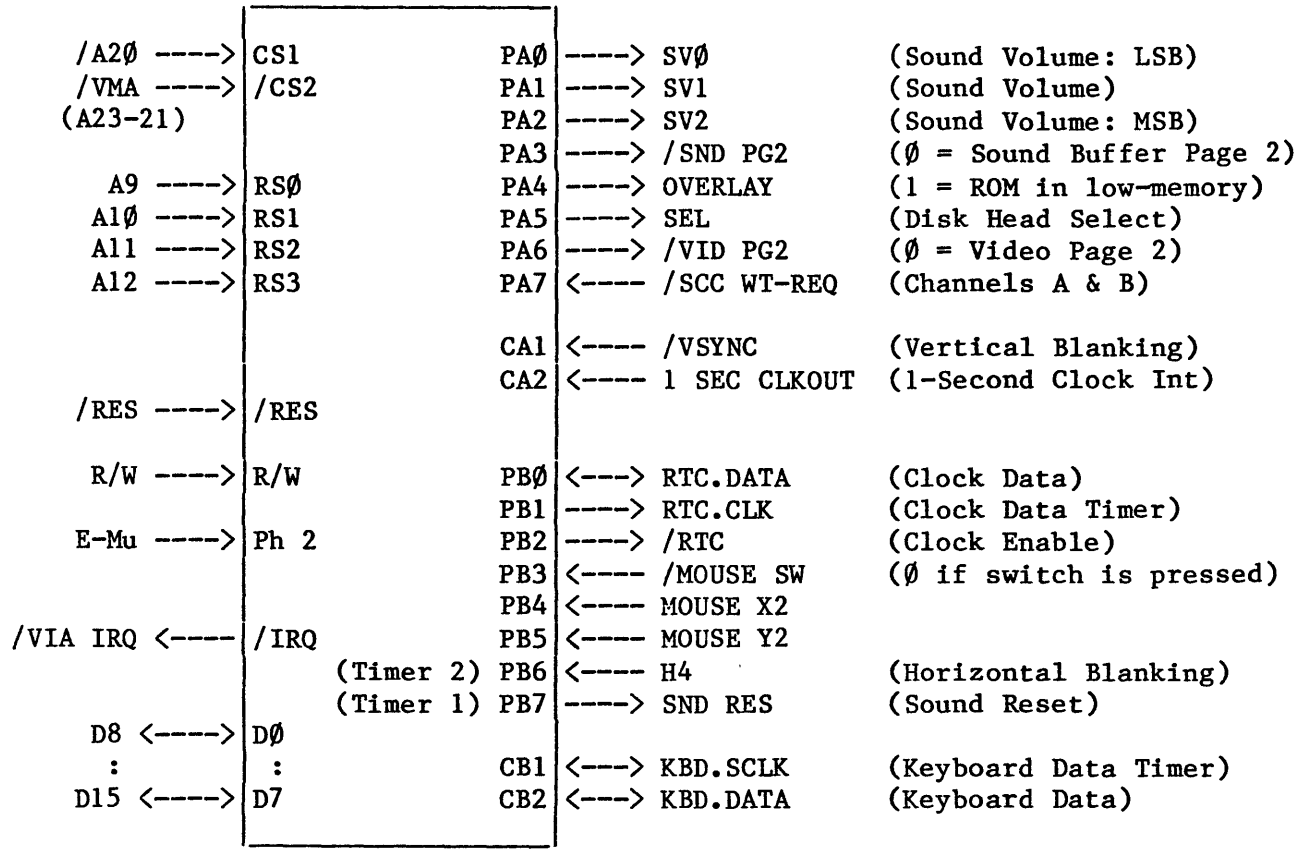


7.3 Some Useful IWM Addresses

These are addresses which use A12-A9 to set individual bits in the IWM State Register.

State Register Bit	IWM Function		Macintosh Address	
0	Phase 0:	Low	$DFE1FF	
		High	$DFE3FF	
1	Phase 1:	Low	$DFE5FF	
		High	$DFE7FF	
2	Phase 2:	Low	$DFE9FF	
		High	$DFEBFF	
3	Phase 3:	Low	$DFEDFF	
		High	$DFEFFF	
4	Motor:	Off	$DFF1FF	(disables all drives)
		On	$DFF3FF	(enables selected drive)
5	Select:	Drive 1	$DFF5FF	(selects internal drive)
		Drive 2	$DFF7FF	(selects external drive)
6	Disk Q6:	Low	$DFF9FF	(called L6 in IWM document)
		High	$DFFBFF	
7	Disk Q7:	Low	$DFFDFF	(called L7 in IWM document)
		High	$DFFFFF	

8. VIA



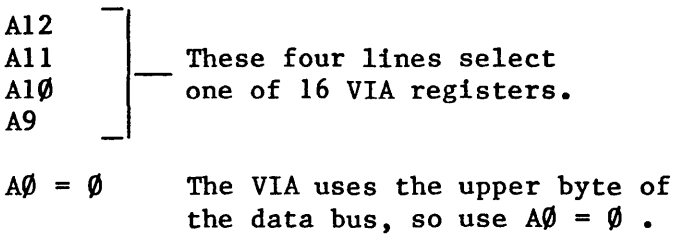
VIA (6522)
U15D

8.1 Address Decoding to Activate VIA

Device Addressed	Address Lines A23	A22	A21	A2Ø	A19	But Do Not Use	Address Range (Exceptions)
VIA (/VMA=Ø, /A2Ø=1)	1	1	1	Ø	X	111ØØ: PHASE READ on bus	$EØØØØØ-$EFFFFF (But do not use $EØØØØØ-$E7FFFF)

(Note: X indicates "don't care": either 1 or Ø)

8.2 Further VIA Address Decoding



8.3 Some Useful VIA Addresses

These are addresses which use A12-A9 to select individual VIA registers.

Input or Output Register A		$EFFFFE	(Do NOT use I-O Register A with Handshake: $EFE3FE)
Input or Output Register B		$EFE1FE	
Data Direction Register A		$EFE7FE	(Ø-bits indicate inputs,
Data Direction Register B		$EFE5FE	while 1's are outputs)
Timer 1 Counter:	Low Byte	$EFE9FE	(Associated with PB7)
	High Byte	$EFEBFE	
Timer 1 Latch:	Low Byte	$EFE9FE	
	High Byte	$EFEBFE	
Timer 2 Counter:	Low Byte	$EFF1FE	(Down-counter; may be
	High Byte	$EFF3FE	associated with PB6)
Shift Register		$EFF5FE	(Shifts data into or out of VIA on CB2, clocked by Ph 2, Timer 2, or CB1)
Auxiliary Control Register		$EFF7FE	
Peripheral Control Register		$EFF9FE	
Interrupt Flag Register		$EFFBFE	
Interrupt Enable Register		$EFFDFE	

8.4 Macintosh-Specific Information about VIA Registers

8.4.1 Port A Input, Output, and Data Direction Registers

Port A I-O Reg. Bit	VIA Line Name	Port A Data Direction	Computer Signal Name	(Comments)
7	PA7	<---- Input <----	/SCC WT-REQ	(Channels A & B)
6	PA6	----> Output ---->	/VID PG2	(Ø = Video Page 2)
5	PA5	----> Output ---->	SEL	(Disk Head Select)
4	PA4	----> Output ---->	OVERLAY	(1 = ROM in low-memory)
3	PA3	----> Output ---->	/SND PG2	(Ø = Sound Buffer Page 2)
2	PA2	----> Output ---->	SV2	(Sound Volume: MSB)
1	PA1	----> Output ---->	SV1	(Sound Volume)
Ø	PAØ	----> Output ---->	SVØ	(Sound Volume: LSB)

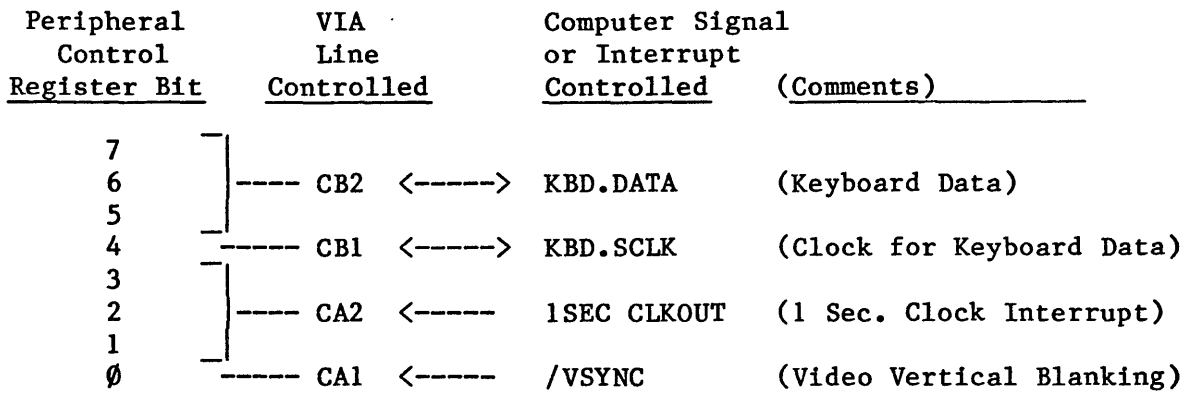
Port A Data Direction Byte: $7F

8.4.2 Port B Input, Output, and Data Direction Registers

Port B I-O Reg. Bit	VIA Line Name	Port B Data Direction	Computer Signal Name	(Comments)
7 (Tmr1)	PB7	----> Output ------>	SND RES	(Sound Reset)
6 (Tmr2)	PB6	<---- Input <------	H4	(Horizontal Blanking)
5	PB5	<---- Input <------	MOUSE Y2	
4	PB4	<---- Input <------	MOUSE X2	
3	PB3	<---- Input <------	/MOUSE SW	(Ø if switch is pressed)
2	PB2	----> Output ------>	/RTC	(Clock Enable)
1	PB1	----> Output ------>	RTC.CLK	(Clock Data Timer)
Ø	PBØ	<---> In or Out <-->	RTC.DATA	(Clock Data)

Port B Data Direction Byte,
when data is coming in from clock: $86
when data is going out to clock: $87

8.4.3 Control Registers



Peripheral Control Register Bit	VIA Line Controlled		Computer Signal or Interrupt Controlled	(Comments)
7				
6	CB2	<----->	KBD.DATA	(Keyboard Data)
5				
4	CB1	<----->	KBD.SCLK	(Clock for Keyboard Data)
3				
2	CA2	<-----	1SEC CLKOUT	(1 Sec. Clock Interrupt)
1				
Ø	CA1	<-----	/VSYNC	(Video Vertical Blanking)

8.4.4 Interrupt Flag and Enable Registers

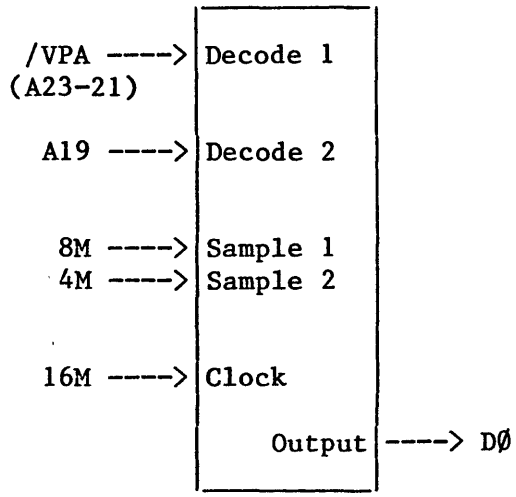
Interrupt Flag Reg. Bit	VIA Function Flagged	Computer Signal Flagged	(Comments)
7	/IRQ (any enabled VIA interrupts)	Also sets /VIA IRQ (/IPLØ)	
6	Timer 1 (PB7)	SND RESET	(Sound Timer)
5	Timer 2		
4	CB1	KBD.SCLK	(Clock for Keyboard Data)
3	CB2	KBD.DATA	(Keyboard Data)
2	Shift Register	Eight bits of KBD.DATA Shifted	
1	CA1	/VSYNC	(Video Vertical Blanking)
Ø	CA2	1SEC CLKOUT	(1 Sec. Clock Interrupt)

The Interrupt Enable Register is arranged just like the Interrupt Flag Register except that bit 7 is "Set/Clear":

Bit 7 Value	Meaning of Values In Bits 6 Through Ø
1	Each 1 enables the corresponding interrupt
Ø	Each Ø enables the corresponding interrupt

9. PHASE READ

Note: PHASE READ is one of the functions of the TSG. It allows the programmer to determine (usually at power-up) whether the computer's high-frequency timing signals are correctly in phase, or out of phase and needing correction.



PHASE READ Function
of the TSG
U3D

9.1 Address Decoding to Activate PHASE READ

Device Addressed	Address Lines A23	A22	A21	A2Ø	A19	But Do Not Use	Address Range (Exceptions)
PHASE READ (/VPA=Ø, A19=Ø)	1	1	1	X	Ø	111ØØ: VIA also on bus	$EØØØØØ-$E7FFFF and $FØØØØØ-$F7FFFF (But do not use $EØØØØØ-$E7FFFF)

(Note: X indicates "don't care": either 1 or Ø)

9.2 Further PHASE READ Address Decoding

When PHASE READ is addressed, the TSG puts phase information on the lowest data line, DØ . To read the lower data byte containing DØ, AØ = 1 would normally be used. However, in practice, you do a multiple-word read to collect phase information at the correct times (see section 9.3).

Note: Do not write to PHASE READ. While it will not actually damage the computer, writing to PHASE READ will cause bus contention.

9.3 Using PHASE READ

To read phase information, do a multiple-word read of three words, starting at address $F7FFFA

The lowest data bit (DØ) in each of the three words then contains the desired phase information, which can be interpreted as follows:

Number of Phase Words in which DØ=1	State of Timing
Ø	In Phase
1	In Phase
2	Out of Phase
3	Out of Phase

If the timing is out of phase, a word-access to the SCC (see section 6.3) will correct the situation by adjusting the phase.

1Ø. AUTO-VECTOR "READ" ADDRESSES

When servicing an interrupt, the CPU "reads" an address in the range $FFFFFØ - $FFFFFF . The exact address consists of ones on all address lines, except that address lines A3-A1 are determined by the three interrupt lines /IPL2-Ø:

Interrupting Device	Interrupt Line	Address Line A3	A2	A1	Address "Read"
VIA	/IPLØ	Ø	Ø	1	$FFFFF3
SCC	/IPL1	Ø	1	Ø	$FFFFF5
VIA + SCC (Transient: Retry)	/IPL1+Ø	Ø	1	1	$FFFFF7
Interrupt Switch	/IPL2	1	Ø	Ø	$FFFFF9
Int. + VIA	/IPL2+Ø	1	Ø	1	$FFFFFB
Int. + SCC	/IPL2+1	1	1	Ø	$FFFFFD
Int. + SCC + VIA	/IPL2+1+Ø	1	1	1	$FFFFFF

No device is activated, and any data "read" is ignored. The only response of the system is that device BMU1 sets the signal /VPA low. This in turn causes the CPU to set /VMA low and to jump through the appropriate auto-vector location in low memory.

When any address in the range $EØØØØØ-$FFFFFF is accessed, BMU1 sets /VPA low. and the CPU responds by setting /VMA low. However, the CPU does not do an auto-vector jump unless the address was "read" by the CPU in servicing an interrupt.

11. SOME USEFUL DECODING EQUATIONS

Note: while some device functions are selected by address lines directly, others are selected by Macintosh signals which are internally decoded as follows:

```
/ROMEN = Ø      when  OVERLAY=1 and A23=Ø and A22=Ø
            or when  OVERLAY=1 and A23=Ø and A22=1 and A21=Ø
            or when  OVERLAY=Ø and A23=Ø and A22=1
            or when  A23=1 and A22=Ø
            or when  A23=1 and A22=1 and A21=Ø

/RAMEN = Ø      when  OVERLAY=1 and A23=Ø and A22=1 and A21=1
            or when  OVERLAY=Ø and A23=Ø and A22=Ø
            or when  A23=1 and A22=Ø


/RAM READ = Ø  when  /RAMEN=Ø and /ROMEN=1 and R/W=1 and /DTACK=Ø

 RAM R/W  = Ø  when  /RAMEN=Ø and /ROMEN=1 and R/W=Ø and /DTACK=Ø


/SCC EN = Ø     when  /AS=Ø and A23=1 and A22=Ø

/SCC RD = Ø     when  /AS=Ø and A23=1 and A22=Ø and A21=Ø


/IWM = Ø        when  /AS=Ø and A23=1 and A22=1 and A21=Ø


/VPA = Ø        when  /AS=Ø and A23=1 and A22=1 and A21=1

/VMA = Ø        when  /VPA=Ø

/AØ = Ø         when  AØ=1
```

Phase Info on DØ when /VPA=Ø and A19=Ø