

## **Handy Appendix 2**

### **Hardware Addresses**

Original 21-Dec-87 Dave Needle  
6-Mar-89 Rev K  
27-Mar-89 Rev K-1  
12-Apr-89 Rev K-2  
17-Apr-89 Rev L  
22-May-89 Rev L-1  
30-May-89 Rev L-2  
16-June-89 Rev L-3  
7-July-89 Rev M (DevCon)

## 1. Basic Address Ranges

The 6502 CPU has an address range of 64K bytes. The upper 1K byte of address space is shared by the RAM, the ROM, Suzy hardware and Mikey hardware. This upper 1K byte of space is divided as follows:

FFFE, FFFF = CPU Interrupt Vector	.....	
FFFC, FFFD = CPU Reset Vector		\ RAM or ROM
FFFA, FFFB = CPU NMI Vector	.....	/
FFF9 = Memory Map Control	.....	Mikey and Suzy hardware
FFF8 = Reserved for future hardware	.....	Mikey and Suzy hardware
FE00 thru FFF7 = ROM Space		
FD00 thru FDFF = Mikey Space		
FC00 thru FCFF = Suzy Space		

The above listed address ranges (except for FFF8 and FFF9) can, under control of the CPU, have RAM overlayed on them. These overlays are controlled by the bits in the hardware register at FFF9. Both Mikey and Suzy accept a write at those addresses but only Mikey responds to a read.

The remaining 63k bytes are RAM.

These address ranges affect only CPU accesses from Mikey. The video and refresh generators in Mikey and the sprite engine in Suzy 'see' the entire 64K byte range as RAM.

\*\*\*\*\*  
**BIG NOTE !!!!** We are done messing around with these addresses. Do not change them anymore !!!

\*\*\*\*\*

## 1.1 Legend

(W), (R), (R/W) indicate write only, read only, and read/write capability of a particular address. Please follow the rules. If an address is read only, DON'T write to it. If an address is write only, DON'T read it.

The reset condition is shown with bit 7 on the left (B7.....B0). X indicates unchanged by reset.

(U) indicates that it is unsafe to read or write this address without first checking (at a different address) to see if it is available. Please follow the rules. A description of UNSAFE is in the hardware specification.

(CPU), (ENG), (SCB), indicate the usual source of data for this particular address. CPU = Mikey, ENG = sprite engine, SCB = DMA from the RAM based sprite control block.

## 1.2 Memory Map Control

FFF9 = MAPCTL, Memory Map Control, (R/W)Mikey reset = 0,0,0,0,0,0,0,0  
(W) Suzy reset = x,x,x,x,x,x,0  
(Only bit 0 is implemented)

B7 = sequential disable. If set, the CPU will always use full cycles (5 ticks min), never a sequential cycle (4 ticks).

B6, B5, B4 = reserved for future use

B3 = FFFA -> FFFF, Vector Space

B2 = FE00 -> FFF7, ROM Space

B1 = FD00 -> FDFF, Mikey Space

B0 = FC00 -> FCFF, Suzy Space

All 8 bits are set to 0 at reset. Any address space bit that is set to a 1 will cause its related address space to access RAM instead of the hardware or ROM normally accessed.

## 2. Suzy Addresses

FC00 -> FC7F = Sprite Control Block, (R/W) (U) reset = x,x,x,x,x,x,x,x

Even addresses are the LSB. Any CPU write to an LSB will set the MSB to 0.

		(address)	
L0	(ENG)	00,01	TMPADRL,H      Temporary address
H0	(ENG)	02,03	TILTACUML,H      Accumulator for tilt value
L1	(CPU)	04,05	HOFFL,H      Offset to H edge of screen
H1	(CPU)	06,07	VOFFL,H      Offset to V edge of screen
L2	(CPU)	08,09	VIDBASL,H      Base Address of Video Build Buffer
H2	(CPU)	0A,0B	COLLBASL,H      Base Address of Coll Build Buffer
L3	(ENG)	0C,0D	VIDADRL,H      Current Video Build Address
H3	(ENG)	0E,0F	COLLADRL,H      Current Collision Build Address
L4	(SCB)	10,11	SCBNEXTL,H      Address of Next SCB
H4	(SCB)	12,13	SPRDLINEL,H      Start of Sprite Data Line Address
L5	(SCB)	14,15	HPOSSTRTL,H      Starting Hpos
H5	(SCB)	16,17	VPOSSTRTL,H      Starting Vpos
L6	(SCB)	18,19	SPRHISIZL,H      H Size
H6	(SCB)	1A,1B	SPRVSIZL,H      V Size
L7	(ENG)	1C,1D	STRETCHL,H      H Size Adder
H7	(ENG)	1E,1F	TILTL,H      H Position Adder
L8	(ENG)	20,21	SPRDOFFL,H      Offset to Next Sprite Data Line
H8	(ENG)	22,23	SPRVPOSL,H      Current Vpos
L9	(CPU)	24,25	COLLOFFL,H      Offset to Collision Depository
H9	(ENG)	26,27	VSIZACUML,H      Vertical Size Accumulator
L10	(CPU)	28,29	HSIZOFFL,H      Horizontal Size Offset
H10	(CPU)	2A,2B	VSIZOFFL,H      Vertical Size Offset
L11	(ENG)	2C,2D	SCBADRL,H      Address of Current SCB
H11	(ENG)	2E,2F	PROCADRL,H      Current Spr Data Proc Address
	(ENG)	30 thru 4F reserved	
	(ENG)	50,51	Do Not Use-----
H4	(ENG)	52	MATHD (12)
H4	(ENG)	53	MATHC (13)
L5	(ENG)	54	MATHB (14)
L5	(ENG)	55	MATHA (15)
H5	(ENG)	56	MATHP (16)
H5	(ENG)	57	MATHN (17)
	(ENG)	58 thru 5F	Do Not Use \ These are used for MATH
L8	(ENG)	60	MATHH (20)
L8	(ENG)	61	MATHG (21)
H8	(ENG)	62	MATHF (22)
H8	(ENG)	63	MATHE (23)
	(ENG)	64 thru 6B	Do Not Use \ operations. The meanings
L11	(ENG)	6C	MATHM (2C)
L11	(ENG)	6D	MATHL (2D)
H11	(ENG)	6E	MATHK (2E)
H11	(ENG)	6F	MATHJ (2F)
	(ENG)	70 thru 7F	Do Not Use / of the letters are
			/ explained in the spec.

FC80 = SPRCTL0, Sprite Control Bits 0, (W) (U) reset = x,x,x,x,x,x,x,x

B7,B6 = bits/pixel-1 (1,2,3,4)

B5 = H flip, 0 = not flipped

B4 = v flip, 0 = not flipped

B3 = reserved

B2,B1,B0 = Sprite Type

7 -> shadow

6 -> exclusive-or, shadow

5 -> non-collidable

4 -> normal

3 -> boundary

2 -> boundary-shadow

1 -> background-no collision

0 -> background, shadow

(old

B2,B1,B0 = Sprite Type

7 -> normal

6 -> exclusive-or

5 -> non-collidable

4 -> shadow

3 -> boundary-shadow

2 -> boundary

1 -> background-no collision

0 -> background

)

FC81 = SPRCTL1, Sprite Control Bits 1, (W) (U) reset = x,x,x,x,x,x,x,x

B7 = literal attribute, 0=normal, 1=totally literal

B6 = Sizing algorithm choice, 0=add (algo 4), 1=shifter (algo 3)  
(SET IT TO ZERO !!!, algo 3 is broke)

B5,B4 = Reloadable depth.

0=none.

1=Hsize,Vsize.

2=Hsize,Vsize,Stretch.

3=Hsize,Vsize,Stretch,Tilt.

B3 = Palette re-load. 0=reload the palette, 1=use existing palette.

B2 = Skipsprite. 1=skip this sprite, 0=use this sprite.

B1 = Start drawing up (1=up, 0=down)

B0 = Start drawing left (1=left, 0=right)

FC82 = SPRCOLL, Sprite Collision Number, (W) (U)

reset = x,x,x,x,x,x,x,x

B7,B6 = 0

B5 = dont collide. 1=dont collide with this sprite.

B4=0

B3,B2,B1,B0=number

FC83 = SPRINIT. Sprite Initialization Bits, (W) (U)

B7 = fc1, B6 = fc2, B5 = fc3, B4 = reserved  
B3 = ac1, B2 = ac2, B1 = ac3, B0 = ac4

reset = x,x,x,x,x,x,x,x

Set to 'F3' after at least 100ms after power up and before any sprites are drawn.

FC84 -> FC87 = not yet allocated

FC88 = SUZYHREV. Suzy Hardware Revision (R)  
='01'

FC89 = SUZYSREV. Suzy Software Revision (W)  
no actual register is implemented

FC8A -> FC8F = not yet allocated

FC90 = SUZYBUSEN. Suzy Bus Enable FF, (W) reset = 0  
B0 = Suzy Bus Enable, 0=disabled

FC91 = SPRGO. Sprite Process Start Bit (W) reset = 0  
B0 = Sprite process enabled, 0=disabled.

Write a 1 to start the process, at completion of process this bit will be reset to 0.  
Either setting or clearing this bit will clear the Stop At End Of Current Sprite bit.  
B1 is reserved  
B2= enable everon detector. 1=enabled.

FC92 = SPRSYS. System Control Bits. (R/W)

(write) reset = x,x,x,x,x,x,x

B7 = Signmath. 0 = unsigned math, 1 = signed math.

B6 = OK to accumulate. 0=do not accumulate, 1=yes, accumulate.

B5 = dont collide. 1=dont collide with any sprites.

B4 = Vstretch. 1=stretch the v. 0=Don't play with it, it will grow by itself.  
B3 = Lefthand, 0=normal handed

B2 = Clear the 'unsafeAccess' bit. 1=clear it. 0=no change.

B1 = Stop at end of current sprite, 1= request to stop. Continue sprite processing by  
setting the Sprite Process Start Bit. Either setting or clearing the SPSB will clear this  
stop request.

B0 = has no effect

(read) reset = 0,0,0,x,x,0,x,0

B7 = Math in process

B6 = Mathbit. If mult, 1=accumulator overflow. If div, 1=div by zero attempted.

B5 = Last carry bit.

B4 = Vstretch.

B3 = Lefthand.

B2 = UnsafeAccess. 1=Unsafe Access was performed.

B1 = Stop at end of current sprite, 1= request to stop.

B0 =Sprite process was started and has neither completed nor been stopped.

FC93 -> FCAF = not yet allocated

FCB0 = JOYSTICK, Read Joystick and Switches(R)

If Lefthand=1

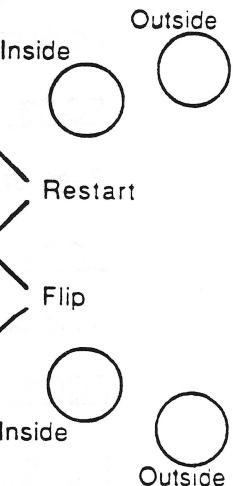
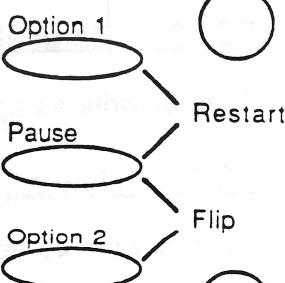
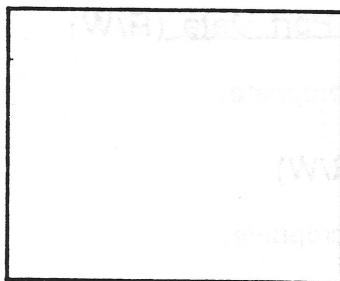
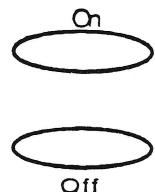
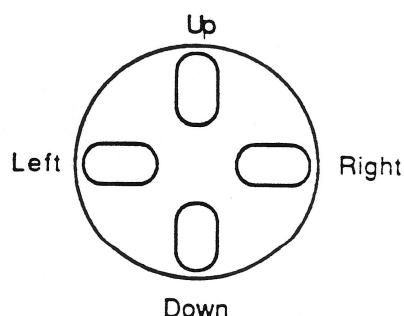
B7 = Joy Up  
B6 = Joy Down  
B5 = Joy Left  
B4 = Joy Right  
B3 = Option 1 (was Re-Start)  
B2 = Option 2 (was Pause)  
B1 = Inside (was Right Fire)  
B0 = Outside (was Left Fire)

If Lefthand=0

(Down)  
(Up)  
(Right)  
(Left)  
Option 1 (was Re-Start)  
Option 2 (was Pause)  
Inside (was Right Fire)  
Outside (was Left Fire)

FCB1 = SWITCHES, Read Other Switches (R)

B7 = 0  
B6 = 0  
B5 = 0  
B4 = 0  
B3 = 0  
B2 = Cart1 I/O Inactive  
B1 = Cart0 I/O Inactive  
B0 = Pause (was Flablude)



This Picture shows the 'Normal' orientation of the unit, which according to our previous definition is the 'LeftHand' orientation. Please realize that this means that the normal setup is 'Flip' = off, and 'LeftHand' = on.

The current agreement is that 'Pause' and 'Option 1' pressed together will result in the patter of little rubber feet around the house. Also, 'Pause' and "Option 2" pressed together will be used to indicate a user request for the 'Flip' function.

FCB2,FCB3 = RCART(R/W)

Read or write 8 bits of data.

FCB2 uses 'CART0/' as the strobe, FCB3 uses 'CART1/' as the strobe.

FCB4-> FCBF = not yet allocated

FCC0 = LEDS(W)reset=0,0,0,0,0,0,0,0

A '0' in a bit will turn its corresponding LED on.

FCC1 = Reserved

FCC2 = Parallel Port Status(R/W)reset=0,0,x,x,x,x,x,x  
(write)

B7 = direction, 0=input, 1=output

B6 = paper out, used to signal the device at the other end of the parallel cable.  
(read)

B7 = data available.

When port is in input mode, B7=1 indicates that a character has arrived and not yet been read by the CPU. This bit will be cleared when the CPU reads the parallel port data.

When in output mode, B7=1 indicates that an output character is in the data register and awaiting acknowledgement from the device on the other end of the parallel cable. This bit is cleared when that other device accepts the character.

In addition, this bit is cleared whenever the direction of the port is changed.

B6 = Hello Handy

This bit is connected to the 'Busy' pin of the parallel port. It is intended as a signal line FROM the device on the other end of the parallel cable.

FCC3 = Parallel Port Data (R/W)

Read or write as appropriate.

FCC4 = Howie (R/W)

Read or write as appropriate.

FCC5 = Reserved

FCC6 -> FCCF = not yet allocated

FCD0 -> FCFF = not yet allocated

### 3. Mikey Addresses (R/W)

Timers are reset to 0

FD00 -> FD03 = Timer channel 0 and Hcount  
FD04 -> FD07 = Timer channel 1 and mag0a (read current state of TAPE0 in b7)  
FD08 -> FD0B = Timer channel 2 and Vcount  
FD0C -> FD0F = Timer channel 3 and mag0b  
FD10 -> FD13 = Timer channel 4 and serial rate  
FD14 -> FD17 = Timer channel 5 and mag1a (read current state of TAPE1 in b7)  
FD18 -> FD1B = Timer channel 6  
FD1C -> FD1F = Timer channel 7 and mag1b

FD00 = TIM0BKUP, HTIMBKUP, Timer 0 backup value  
FD01 = TIM0CTLA, HTIMCTL0, Timer 0 static control  
FD02 = TIM0CNT, Timer 0 current count  
FD03 = TIM0CTLB, Timer 0 dynamic control  
FD04 = TIM1BKUP, MAGA, Timer 1 backup value  
FD05 = TIM1CTLA, Timer 1 static control  
FD06 = TIM1CNT, Timer 1 current count  
FD07 = TIM1CTLB, Timer 1 dynamic control  
FD08 = TIM2BKUP, VTIMBKUP, Timer 2 backup value  
FD09 = TIM2CTLA, Timer 2 static control  
FD0A = TIM2CNT, Timer 2 current count  
FD0B = TIM2CTLB, Timer 2 dynamic control  
FD0C = TIM3BKUP, MAGB, Timer 3 backup value  
FD0D = TIM3CTLA, Timer 3 static control  
FD0E = TIM3CNT, Timer 3 current count  
FD0F = TIM3CTLB, Timer 3 dynamic control  
FD10 = TIM4BKUP, BAUDBKUP, Timer 4 backup value  
FD11 = TIM4CTLA, Timer 4 static control  
FD12 = TIM4CNT, Timer 4 current count  
FD13 = TIM4CTLB, Timer 4 dynamic control  
FD14 = TIM5BKUP, MAGC, Timer 5 backup value  
FD15 = TIM5CTLA, Timer 5 static control  
FD16 = TIM5CNT, Timer 5 current count  
FD17 = TIM5CTLB, Timer 5 dynamic control  
FD18 = TIM6BKUP, Timer 6 backup value  
FD19 = TIM6CTLA, Timer 6 static control  
FD1A = TIM6CNT, Timer 6 current count  
FD1B = TIM6CTLB, Timer 6 dynamic control  
FD1C = TIM7BKUP, MAGD, Timer 7 backup value  
FD1D = TIM7CTLA, Timer 7 static control  
FD1E = TIM7CNT, Timer 7 current count  
FD1F = TIM7CTLB, Timer 7 dynamic control

xxx0 = backup value (count+1)  
xxx1 = control bits:  
    B7=Enable interrupt  
    B6=Reset timer done (see xxx3, bit 3)  
    B5=Unused in timers 0,2,4,6. Magmode in timers 1,3,5,7.  
    B4=Enable reload  
    B3=Enable count  
    B2,B1,B0=clock select  
        0=1us  
        1=2us  
        2=4us  
        3=8us  
        4=16us  
        5=32us  
        6=64us  
        7=linking  
xxx2 = Timer current value  
xxx3 = Other control bits  
    B7,B6,B5,B4=unused  
    B3=Timer done. It is set on time out, reset with the reset timer done bit (xxx1,  
        B6).  
    B2=Last clock.  
    B1=Borrow in.  
    B0=Borrow out.

Audio are reset to 0, all are read/write

FD20 -> FD27 = Audio channel 0, links from timer 7  
FD28 -> FD2F = Audio channel 1, links from audio timer 0  
FD30 -> FD37 = Audio channel 2, links from audio timer 1  
FD38 -> FD3F = Audio channel 3, links from audio timer 2

### FD20 = 8 bit, 2's Complement Volume Control

00000000 is zero volume

00000001 is the same volume as 11111111, but opposite phase.

### FD21 = Shift Register Feedback Enable

B7=feedback bit 11

B6=feedback bit 10

B5=feedback bit 5

B4=feedback bit 4

B3=feedback bit 3

B2=feedback bit 2

B1=feedback bit 1

B0=feedback bit 0

### FD22 = Audio Output Value

In normal mode, shift reg 0 = 0: contains 2's complement of volume register.

In normal mode, shift reg 0 = 1: contains value of volume register.

In integrate mode, shift reg 0 = 0: subtract volume register from output.

In integrate mode, shift reg 0 = 1: add volume register to output.

note that there is hardware clipping at max and min (ff,00).

### FD23 = Lower 8 Bits of Shift Register

### FD24 = Audio Timer Backup Value

### FD25 = Audio Control Bits

B7=feedback bit 7

B6=reset timer done

B5=enable integrate mode

B4=enable reload

B3=enable count

B2,B1,B0=clock select

7='linking'

6=64 us

5=32 us

4=16 us

3=8 us

2=4 us

1=2 us

0=1 us

### FD26 = Audio Counter

Value of the current count at the time it is read

### FD27 = Other Audio Bits

B7=shift register bit 11  
B6=shift register bit 10  
B5=shift register bit 9  
B4=shift register bit 8  
B3=for Glenn to know and you to find out  
B2=last clock state (0->1 causes count)  
B1=borrow in (1 causes count)  
B0=borrow out (count=0 and borrow in)

### FD40 -> FD4F = not yet allocated

### FD50 = MSTEREO(\*R/W) reset = 0,0,0,0,0,0,0,0

\*Note: Write-only on Howards  
A '1' in the bit disables the indicated audio connection.

B7=Audio Channel 3 -> Left Ear  
B6=Audio Channel 2 -> Left Ear  
B5=Audio Channel 1 -> Left Ear  
B4=Audio Channel 0 -> Left Ear  
B3=Audio Channel 3 -> Right Ear  
B2=Audio Channel 2 -> Right Ear  
B1=Audio Channel 1 -> Right Ear  
B0=Audio Channel 0 -> Right Ear

### FD51 -> FD7F = not yet allocated

### FD80 = INTRST. Interrupt Poll 0, (R/W) reset = 0,0,0,0,0,0,0,0

Read is a poll, write will reset the int that corresponds to a set bit.

### FD81 = INTSET. Interrupt Poll 1, (R/W) reset = 0,0,0,0,0,0,0,0

Read is a poll, write will set the int that corresponds to a set bit.

B7=timer 7  
B6=timer 6  
B5=timer 5  
B4=serial interrupt  
B3=timer 3  
B2=timer 2 (vertical line counter)  
B1=timer 1  
B0=timer 0 (horizontal line timer)

### FD82 -> FD83 = not yet allocated

### FD84 = MAGRDY0, Mag Tape Channel 0 ready bit,(R) reset = x

B7=edge (1 ) Reset upon read.

### FD85 = MAGRDY1, Mag Tape Channel 1 ready bit,(R) reset = x

B7=edge (1 ) Reset upon read.

FD86 = AUDIN, Audio In,(R) reset = b7,0,0,0,0,0,0,0  
B7=Audio in comparator

FD87 = SYSCTL1, Control Bits (W) reset = x,x,x,x,x,x,1,0  
B1=power (1=on)  
B0=Cart Address Strobe (also counter reset) (was MotorOn)

FD88 = MIKEYHREV, Mikey Hardware Revision(R)  
= '01'

FD89 = MIKEYSREV. Mikey Software Revision(W)  
no actual register is implemented

FD8A = IODIR, Mikey Parallel I/O Data Direction (W)

reset = 0,0,0,0,0,0,0

8 bits I/O direction corresponding to the 8 bits at FD8B  
0=input, 1=output

FD8B = IODAT, Mikey Parallel Data(sort of a R/W)  
8 bits of general purpose I/O data

Note that some lines are used for several functions, please read the spec.  
Also note that only the lines that are set to input are actually valid for reading.  
---READ THE SPEC---

FD8C = SERCTL, Serial Control Register, (R/W) reset = 0,0,0,0,0,0,0,0  
(write)

B7=TXINTEN transmitter interrupt enable  
B6=RXINTEN receive interrupt enable  
B5=0 (for future compatibility)  
B4=PAREN xmit parity enable (if 0, PAREVEN is the bit sent)  
B3=RESETERR reset all errors  
B2=TXOPEN 1 = open collector driver, 0=TTL driver  
B1=TXBRK send a break (for as long as the bit is set)  
B0=PAREVEN send/recv even parity  
(read)

B7=TXRDY transmitter buffer empty  
B6=RXRDY receive character ready  
B5=TXEMPTY transmitter totally done  
B4=PARERR received parity error  
B3=OVERRUN received overrun error  
B2=FRAMERR received framing error  
B1=RXBRK break received (24 bit periods)  
B0=PARBIT 9th bit

FD8D = SERDAT, Serial Data, (R/W) reset = x,x,x,x,x,x,x,x

FD8E -> FD8F = not yet allocated

FD90 = SDONEACK, Suzy Done Acknowledge, (W) reset = 0 (not acked)

FD91 = CPUSLEEP, CPU Bus Request Disable(W)

A write of '0' to this address will reset the CPU bus request flip flop. The setting of the flip flop is described in the hardware specification.

FD92 = DISPCTL, Video Bus Request Enable, (W) reset = 0

B7,B6,B5,B4 = 0  
B3 = color, 1=color, 0=monochrome  
B2 = fourbit, 1=4 bit mode, 0=2 bit mode  
B1 = 1=flip, 0=normal  
B0 = 1=enable video DMA, 0=disable

FD93 = PBKUP, Magic 'P' count, (W) reset = x,x,x,x,x,x,x  
INT(((line time - .5us) / 15) \* 4) -1

At 60 Hz, 'PBKUP' = 41. (x'29')

FD94 ->FD95= DISPADRL,H . Start Address of Video Display, (W)  
reset = x,x,x,x,x,x,x,x

DISPADRH (FD95) is upper 8 bits of display address.

DISPADRL (FD94) is lower 8 bits of display address with the bottom 2 bit ignored by the hardware. The address of the upper left corner of a display buffer must always have '00' in the bottom 2 bits.

ED96 -> FD9B = not yet allocated

ED9C = Mtest0 ,(W) reset = 0,0,0,0,0,0,0,0

B7 = ATcnt16  
B6 = ATtest  
B5 = XCLKEN  
B4 = UARTturbo  
B3 = ROMsel  
B2 = ROMtest  
B1 = Mtest  
B0 = CPUtest

ED9D = Mtest1 ,(W) reset = x,0,0,0,0,0,0,0

B7 = not used  
B6 = Pcnt16  
B5 = REFcnt16  
B4 = VIDtrig  
B3 = REFtrig  
B2 = VIDdmaDIS  
B1 = REFfast  
B0 = REFdis

ED9E = Mtest2 ,(W) reset = x,x,x,0,0,0,0,0

B7 ->B5 not used  
B4 = Vstrobe  
B3 = Vzero  
B2 = H120  
B1 = Hzero  
B0 = Vblankef

FDA0 -> FDAF =Green color map, (R/W) reset = x,x,x,x,x,x,x,x  
B3->B0=green

FDB0 -> FDBF =Blue and Red color map, (R/W) reset = x,x,x,x,x,x,x,x  
B7->B4=blue  
B3->B0=red

FDA0 = GREEN0	FDA8 = GREEN8
FDA1 = GREEN1	FDA9 = GREEN9
FDA2 = GREEN2	FDAA = GREENA
FDA3 = GREEN3	FDAB = GREENB
FDA4 = GREEN4	FDAC = GREENC
FDA5 = GREEN5	FDAD = GREEND
FDA6 = GREEN6	DAE = GREENE
FDA7 = GREEN7	FDAF = GREENF

FDB0 = BLUERED0	FDB8 = BLUERED8
FDB1 = BLUERED1	FDB9 = BLUERED9
FDB2 = BLUERED2	FDBA = BLUEREDA
FDB3 = BLUERED3	FDBB = BLUEREDB
FDB4 = BLUERED4	FDBC = BLUEREDC
FDB5 = BLUERED5	FDBD = BLUEREDD
FDB6 = BLUERED6	FDBE = BLUEREDE
FDB7 = BLUERED7	FDBF = BLUEREDF

FDC0 -> FDCF = not yet allocated

FDD0 -> FDFF = not yet allocated

## Address Summary ... System

FE00 thru FFF7 = ROM Space

FFF8 = Reserved for future hardware

FFF9 = Memory Map Control, Mikey reset= 0,0,0,0,0,0,0, Suzy reset = x,x,x,x,x,x,x,0

FFFA, FFFB = CPU NMI Vector

FFFC, FFFD = CPU Reset Vector

FFFE, FFFF = CPU Interrupt Vector

## Address Summary ... Suzy (dont forget FFF9)

FC00 -> FC7F = Sprite Control Block, (R/W) (U) reset = x,x,x,x,x,x,x,x

Even addresses are the LSB. Any CPU write to an LSB will set the MSB to 0.

		(address)	
L0	(ENG)	00,01	TMPADRL,H      Temporary address
H0	(ENG)	02,03	TILTACUML,H      Accumulator for tilt value
L1	(CPU)	04,05	HOFFL,H      Offset to H edge of screen
H1	(CPU)	06,07	VOFFL,H      Offset to V edge of screen
L2	(CPU)	08,09	VIDBASL,H      Base Address of Video Build Buffer
H2	(CPU)	0A,0B	COLLBASL,H      Base Address of Coll Build Buffer
L3	(ENG)	0C,0D	VIDADRL,H      Current Video Build Address
H3	(ENG)	0E,0F	COLLADRL,H      Current Collision Build Address
L4	(SCB)	10,11	SCBNEXTL,H      Address of Next SCB
H4	(SCB)	12,13	SPRDLINEL,H      Start of Sprite Data Line Address
L5	(SCB)	14,15	HPOSSTRTL,H      Starting Hpos
H5	(SCB)	16,17	VPOSSTRTL,H      Starting Vpos
L6	(SCB)	18,19	SPRHISIZL,H      H Size
H6	(SCB)	1A,1B	SPRVSIZL,H      V Size
L7	(ENG)	1C,1D	STRETCHL,H      H Size Adder
H7	(ENG)	1E,1F	TILTL,H      H Position Adder
L8	(ENG)	20,21	SPRDOFFL,H      Offset to Next Sprite Data Line
H8	(ENG)	22,23	SPRVPOSL,H      Current Vpos
L9	(CPU)	24,25	COLLOFFL,H      Offset to Collision Depository
H9	(ENG)	26,27	VSIZACUML,H      Vertical Size Accumulator
L10	(CPU)	28,29	HSIZOFFL,H      Horizontal Size Offset
H10	(CPU)	2A,2B	VSIZOFFL,H      Vertical Size Offset
L11	(ENG)	2C,2D	SCBADRL,H      Address of Current SCB
H11	(ENG)	2E,2F	PROCADRL,H      Current Spr Data Proc Address
		(ENG) 30 thru 4F reserved	
		(ENG) 50,51	Do Not Use-----
H4	(ENG)	52	MATHD (12) \
H4	(ENG)	53	MATHC (13) \
L5	(ENG)	54	MATHB (14) \
L5	(ENG)	55	MATHA (15) \
H5	(ENG)	56	MATHP (16) \
H5	(ENG)	57	MATHN (17) \
		(ENG) 58 thru 5F	Do Not Use \ These are used for MATH
L8	(ENG)	60	MATHH (20) \ operations. The meanings
L8	(ENG)	61	MATHG (21) \ of the letters are
H8	(ENG)	62	MATHF (22) \ explained in the spec.
H8	(ENG)	63	MATHE (23) /

(ENG) 64 thru 6B Do Not Use /  
L11 (ENG) 6C MATHM (2C) /  
L11 (ENG) 6D MATHL (2D) /  
H11 (ENG) 6E MATHK (2E) /  
H11 (ENG) 6F MATHJ (2F) /  
(ENG) 70 thru 7F Do Not Use-----

FC80 = SPRCTL0, Sprite Control Bits 0, (W) (U) reset = x,x,x,x,x,x,x,x  
FC81 = SPRCTL1, Sprite Control Bits 1, (W) (U) reset = x,x,x,x,x,x,x  
FC82 = SPRCOLL, Sprite Collision Number, (W) (U) reset = x,x,x,x,x,x,x  
FC83 = SPRINIT, Sprite Initialization Bits, (W) (U) reset = x,x,x,x,x,x,x  
FC84 -> FC87 = not yet allocated  
FC88 = SUZYHREV, Suzy Hardware Revision (R)  
FC89 = SUYSREV, Suzy Software Revision (W)  
FC8A -> FC8F = not yet allocated  
FC90 = SUZBUSEN, Suzy Bus Enable FF, (W) reset = 0  
FC91 = SPRGO, Sprite Process Start Bit (W) reset = 0  
FC92 = SPRSYS, System Control Bits, (R/W)  
FC93 -> FCAF = not yet allocated  
FCB0 = JOYSTICK, Read Joystick and Switches(R)  
FCB1 = SWITCHES, Read Other Switches (R)  
FCB2,FCB3 = RCART(R/W)  
FCB4-> FCBF = not yet allocated  
FCC0 = LEDS(W)reset=0,0,0,0,0,0,0  
FCC1 = Reserved  
FCC2 = Parallel Port Status(R/W)reset=0,0,x,x,x,x,x  
FCC3 = Parallel Port Data (R/W)  
FCC4 = Howie (R/W)  
FCC5 = Reserved  
FCC6 -> FCCF = not yet allocated  
FCD0 -> FCFF = not yet allocated

## Address Summary ... Mikey (dont forget FFF9)

FD00 -> FD03 = Timer channel 0 and Hcount  
FD04 -> FD07 = Timer channel 1 and mag0a (read current state of TAPE0 in b7)  
FD08 -> FD0B = Timer channel 2 and Vcount  
FD0C -> FD0F = Timer channel 3 and mag0b  
FD10 -> FD13 = Timer channel 4 and serial rate  
FD14 -> FD17 = Timer channel 5 and mag1a (read current state of TAPE1 in b7)  
FD18 -> FD1B = Timer channel 6  
FD1C -> FD1F = Timer channel 7 and mag1b  
FD20 -> FD27 = Audio channel 0, links from timer 7  
FD28 -> FD2F = Audio channel 1, links from audio timer 0  
FD30 -> FD37 = Audio channel 2, links from audio timer 1  
FD38 -> FD3F = Audio channel 3, links from audio timer 2  
FD20 = 8 bit, 2's Complement Volume Control  
FD21 = Shift Register Feedback Enable  
FD22 = Audio Output Value  
FD23 = Lower 8 Bits of Shift Register  
FD24 = Audio Timer Backup Value  
FD25 = Audio Control Bits  
FD26 = Audio Counter  
FD27 = Other Audio Bits  
FD40 -> FD4F = not yet allocated  
FD50 = MSTEREO(R/W) reset = 1,1,1,1,1,1,1,1  
FD51 -> FD7F = not yet allocated  
FD80 = INTRST, Interrupt Poll 0, (R/W) reset = 0,0,0,0,0,0,0,0  
FD81 = INTSET, Interrupt Poll 1, (R/W) reset = 0,0,0,0,0,0,0,0  
FD82 -> FD83 = not yet allocated  
FD84 = MAGRDY0, Mag Tape Channel 0 ready bit,(R) reset = x  
FD85 = MAGRDY1, Mag Tape Channel 1 ready bit,(R) reset = x  
FD86 = AUDIN, Audio In,(R) reset = b7,0,0,0,0,0,0,0  
FD87 = SYSCTL1, Control Bits.(W) reset = x,x,x,x,x,x,1,0  
FD88 = MIKEYHREV, Mikey Hardware Revision(R)  
FD89 = MIKEYSREV, Mikey Software Revision(W)  
FD8A = IODIR, Mikey Parallel I/O Data Direction (W)  
FD8B = IODAT, Mikey Parallel Data(sort of a R/W)  
FD8C = SERCTL, Serial Control Register,(R/W) reset = 0,0,0,0,0,0,0,0  
FD8D = SERDAT, Serial Data ,(R/W) reset = x,x,x,x,x,x,x  
FD8E -> FD8F = not yet allocated  
FD90 = SDONEACK, Suzy Done Acknowledge, (W) reset = 0 (not acked)  
FD91 = CPUSLEEP, CPU Bus Request Disable(W)  
FD92 = DISPCtl, Video Bus Request Enable, (W) reset = 0  
FD93 = PBKUP, Magic 'P' count, (W) reset = x,x,x,x,x,x,x  
FD94 -> FD95= DISPADRl,H , Start Address of Video Display, (W)  
FD96 -> FD9B = not yet allocated  
FD9C = Mtest0 ,(W) reset = 0,0,0,0,0,0,0  
FD9D = Mtest1 ,(W) reset = x,0,0,0,0,0,0  
FD9E = Mtest2 ,(W) reset = x,x,x,0,0,0,0,0  
FDA0 -> FDAF =Green (mono) color map, (R/W) reset = x,x,x,x,x,x,x  
B3->B0=green  
FDB0 -> FDDBF =Blue and Red color map, (R/W) reset = x,x,x,x,x,x,x  
B7->B4=blue B3->B0=red  
FDC0 -> FDCF = not yet allocated  
FDD0 -> FDFF = not yet allocated

## How Bit Assignments

**SPRGO** \$FC91 7 6 5 4 3 2 1 0  
\* These are the SPRGO flag definitions  
**EVER\_ON** \$4 |  
**SPRITE\_GO** \$1 |

	\$FCB0	7	6	5	4
* These are the JOYSTICK bit definitions					
JOY_DOWN	\$80				
JOY_UP	\$40				
JOY_RIGHT	\$20				
JOY_LEFT	\$10				
RESTART_BUTTON	\$8				
PAUSE_BUTTON	\$4				
RIGHT FIRE BUTTON	\$2				
LEFT FIRE BUTTON	\$1				

**SWITCHES** \$FCB1 7 6 5 4 3 2 1  
 These are the SWITCHES bit definitions  
**CART STROBE** \$2  
**FLABLODE BUTTON** \$1

**RCART** **\$FCB2** 7 6 5 4 3 2 1

MAPCTL	\$FFF9	7	6	5	4	3	2	1
* These are the MAPCTL flag definitions								
VECTOR_SPACE	\$8							
ROM SPACE	\$4							
MIKEY_SPACE	\$2							
SUZY SPACE	\$1							

TIM_CONTROLA	\$1						
* Here's the TIM_CONTROLA control bits							
ENABLE_INT	\$80						
RESET_DONE	\$40						
ENABLE_RELOAD	\$10						
ENABLE_COUNT	\$8						
AUD_CLOCK_MASK	\$7						
AUD_LINKING	\$7						
AUD_64	\$6						
AUD_32	\$5						
AUD_16	\$4						
AUD_8	\$3						
AUD_4	\$2						
AUD_2	\$1						
AUD_1	\$0						

