Integrated WOZ Machine (IWM)

Device Specification

Features

- * Backwards-compatible with 16 sector Disk][controller
- * Use of 7M (or 8 mHz) to minimize sampling error rate
- * Fast mode using 2 uS bit cells
- * Asynchronous mode with pollable handshake registers

General Description

The IWM is an integration of the Disk II floppy disc interface. When the IWM is reset, it becomes a controller compatible with the current Disk II interface in its operation with currently supported Apple II and ///software. In addition the IWM has extensions including a status register, mode register, and mutiple modes of operation. The IWM provides an asynchronous mode which relaxes the precise software/hardware timing required in synchronous mode, a fast mode with a data rate twice that of Disk II, and an optional 1 second one-shot timer to hold the enable outputs low.

The IWM is a peripheral device that connects to a host data bus. the [redacted] device generates and receives serial GCR encoded data A programmable digital one shot is used for serial data recovery. The IWM generates buffered drive enables and phase line control outputs.

Packaging and Pin Assignment

The IWM is packaged in a standard 28 pin, 6ØØ mil plastic DIP .

PHASEØ	1	28	PHASE1
PHASE2	2	27	PHASE3
AØ	3	26	Vcc
A1	4	25	Q3
A2	5	24	FCLK
A3	6	23	/RESET
/DEV	7	22	RDDATA
WRDATA	8	21	SENSE
/WRREQ	9	2Ø	/ENBL1
DØ	1Ø	19	/ENBL2
D1	11	18	D7
D2	12	17	D6
D3	13	16	D5
GND	14	15	D4

Technical Description

The primary purpose of the IWM is to allow a microprocessor to read and write serial [redacted] The IWM may be controlled by setting state bits and reading or writing registers. Setting a state bit and accessing a register is done simultaneously. The registers are the mode register, the status register, the write-handshake register, the read data register, and the write data register. [redacted]

8MHZ The data format is an 8 bit nibble with the MSB set. The MSB of the 8 bit data nibble is shifted in or written out first. A bit is transferred every bit cell time. The bit cell time defaults to 4 uS (set to 2 uS in fast mode). Therefore the data rate is one nibble every 32 uS (16 uS in fast mode). [redacted]

The IWM is put into the write state by a transition from the write protect sense state to the write load state. In the synchronous mode, the time of that transition and every 8 Q3 periods (4 uS) thereafter, until L7 is cleared, marks the beginning of a write window. The duration of the write window is 4 periods of the Q3 input signal (2 uS). The the data written at the last write access occurring within this write window will load the shift register with the data to be shifted out. If the next write access has not occured 32 uS (64 Q3 periods) after a load, the write will be extended in multiples of 4 uS (8 Q3 periods) until another write access, and zeros will be shifted out.

[redacted] which would then be 64 and 80 of the Q3 clock input periods in duration, respectively, and the bit cell timings, 8 Q3 periods per bit cell time in slow mode.

[redacted]. The buffer register may be written at any time during the write state. Only the data last written into the buffer register, before the contents of the buffer register is transfered to the write shift register, is used.

In asynchronous mode CLK is used to generate the bit cell timings. In fast mode the CLK clock is equivalent to the clock input on FCLK. In slow mode CLK is equivalent to the clock input on FCLK divided by two. Therefore, in 7M and slow mode the bit cell time will be 28 FCLK clock input periods in duration, in 8M and slow mode the cell time will be 32 periods, and in 8M and fast mode the cell time will be 16 periods. In asynchronous mode the write shift register is loaded every 8 bit cell times starting seven CLK periods after the write state begins.

An underrun occurs when data has not been written to the buffer register between the time the write-handshake bit indicates an empty buffer and the time the buffer is transferred to the write shift-register. If an underrun occurs in asynchronous mode /WRREQ will be disabled (set to a TTL high state) and the /underrun flag will be set to zero. This occurrance can be detected by reading the write-handshake register before clearing state bit L7. Clearing state bit L7 will reset the /underrun flag.

When L6 and L7 are both zero the IWM is in the read state. When reading serial data, a falling transition within a bit cell window is considered to be a one, and no falling transition within a bit cell window is considered to be a zero. The receive data input on RDDATA is synchronized internally with the CLK clock. The synchronized falling transition is then discriminated to the nearest bit cell window using the 7/8 mHz FCLK clock signal in fast mode and the FCLK signal divided by two in slow mode. A digital one-shot data recovery scheme is used. Every falling transition establishes the bit cell windows, used by the data seperator in the IWM to recover the following bits, until another falling transition is received.

In the read state the data is shifted into the LSB of the shift register, and the shift register shifts data from LSB to MSB. A full data nibble is considered to be shifted in when a one is shifted into the MSB. When a full data nibble is shifted into the internal shift register, the data will be latched by the read data register and the shift register will be cleared to all zeros so that it will then be ready to shift in the next data word.

In the synchronous mode the shift register is readable in any intermediate state with this exeception : when a one is shifted into the MSB, the shift register will appear, to the data bus, to be stalled for a period of two bit times plus four CLK periods. This is to allow the host processor time to poll the MSB to determine when data is valid. In asynchronous mode the data register will latch the shift register when a one is shifted into the MSB and will be cleared 14 FCLK periods (about 2 uS) after a valid data read takes place (a valid data read being defined as both /DEV being low and D7 (the msb) outputting a one from the data register for at least one FCLK period).

Read data bit cell windows

mode		Nclks	period	data	notes
slow	7M	7-2Ø	FCLK/2	1	2.Ø-5.71+ uS
		21-34		Ø1	6.Ø-9.71+ uS
		35-48		ØØ1	1Ø.Ø-13.71+ uS
					window is 28 clks
slow	8M	8-23	FCLK/2	1	
		24-39		Ø1	
		4Ø-55		ØØ1	
fast	7M	7-2Ø	FCLK	1	window is 14 clks
		21-34		Ø1	
		35-48		ØØ1	
fast	8M	8-23	FCLK	1	1.Ø-2.875+ uS
		24-39		Ø1	3.Ø-4.75+ uS
		4Ø-55		ØØ1	5.Ø-6.875+ uS
					window is 16 clks

The table above shows how the data seperator in the IWM discriminates between ones and zeros when reading. Nclks is the number of clock periods between falling transitions of the internally synchronous version of RDDATA. The clock period is either that of the FCLK input or that of the FCLK input divided by two in slow mode. Each falling transition resets the read data windows for subsequent data to be relative to that transition. The data patterns noted above are the bit patterns that are shifted in as a result of the transitions and the absence of transitions in their respective windows.

[redacted]. In port operation the MSB can be used to continously clock data into external registers. The MSB will be cleared at least six FCLK periods before being set. Except in port operation, in asynchronous mode the latch mode bit should be set (for reliability in clearing the data register after a read).

Data written to the IWM is sampled by the the zero to one transition of the logical OR of Q3 and /DEV . In asynchronous mode the Q3 input may be tied low.

Signal and Bus operation Description

1. Vcc — +5 volt supply

   GND — Ground reference and negative supply

2. Device Control Signals

   A1-A3 — These three inputs select one of the 8 bits in the state register to be updated.

   AØ — The data input to the state bit selected by A1-A3. The state to which the addressed state bit is set by an operation will select the register to be accessed by that operation.

   Also the /READ input. A low on this input enables the IWM to send the register selected by the state onto the data bus.

   DØ-D7 — The bidirectional data bus .

   /DEV — Active low device enable. The falling edge of /DEV latches information on AØ-A3. The rising edge of the logical function (Q3 OR /DEV) qualifies write register data.

   FCLK — Clock input for the serial data logic; either 7 or 8 MHz.

   Q3 — 2.Ø Mhz clock input used to qualify the timing of the serial data being written out in the synchronous mode.

   /RESET — Active low system reset input. When asserted, this signal places all IWM outputs in their inactive state, and sets the state and the modes to their defaults.

3. Inputs (2)

   RDDATA — The serial data input. The falling transition of each pulse is synchronized by the IWM.

   SENSE — An input to the IWM that can be polled via the status register.

4. Outputs (8)

WRDATA

The serial data output. A transition occurs on this output for each one bit.

/ENBL1 , /ENBL2

Programmable buffered output lines.
No more than one enable may be low at any time. If an enable is low than Motor-On is true.
If the 1-second on board timer is enabled then the selected one will stay low for about 1 second after it is programmed high.

/WRREQ

This signal is a programmable buffered output line.

PHASEØ-3

These are programmable output lines.
A true TTL logic "1" (2.4 volts) can be maintained even while driving two darlington inputs in parallel.

Register Description

State Register

This is an 8-bit write-only pseudo-register. The bits in this register are individually addressed by A3,A2,A1 . The data on AØ is latched into the addressed state bit by /DEV low. All eight state bits are reset to Ø by /RESET low.

Not only do the state bits control certain chip functions and outputs, the setting of two of the state bits L6 and L7, and Motor-On, internally select which register is to be selected and whether the operation is to be a read or a write. If an operation occurs that changes the state of one of these bits to a new state, that new state will select the register to be accessed during that operation and whether that operation is to be a read or a write.

Address	Name	Function
Ø		A 1 in this bit will drive PHASEØ to a high state.
1		PHASE1
2		PHASE2
3		PHASE3
4	LMotor-On	A 1 on LMotor-On sets the enable selected below low
5	Drive-Sel	A 1 on this bit selects /ENBL2; a Ø selects /ENBL1
6	L6	(see description below)
7	L7	(see description below)

The state bits L7 and L6, and Motor-On, select which register is available to be read or written. Other registers are read during any operation in which AØ is a zero. A register is written when both L6 and L7 are set or are being set to 1 and AØ is a one.

The combination of L7 and Motor-On and /underrun enables /WRREQ low.

L7	L6	Motor-On	register operation selected	State Name
Ø	Ø	Ø	read all ones	
Ø	Ø	1	read data register	Read
Ø	1	x	read status register	Write-Protect Sense
1	Ø	x	read write-handshake register	Write
1	1	Ø	write mode register	Mode Set
1	1	1	write data register	Write Load

Mode register (a write only register)

All eight mode bits are reset to Ø by /RESET low.

bit	function
LSB Ø	1 = latch mode (should be set in asynchronous mode)
1	Ø = synchronous handshake protocol; 1 = asynchronous
2	Ø = 1-second on board timer enable; 1 = timer disable
3	Ø = slow mode; 1 = fast mode (2 uS bit cell timing)
4	Ø = 7MHz; 1 = 8MHz (7 or 8 mHz clock descriptor)
5	1 = test mode; Ø = normal operation
6	1 = MZ-reset
MSB 7	reserved for future expansion

In latch mode the msb of the read data is latched internally during /DEV low (this internally latched msb is then used for the determination of a valid data read).

If the 1-second timer bit is a zero then the enable (/ENBL1 or /ENBL2) selected by Drive-Sel will be held low for 2^23 +-1ØØ FCLK periods (about 1 second) after the LMotor-On state bit is reset to zero. If the latch mode bit is set the timer is not guarantied to count up to 2^23. Motor-On is synonomous with either /ENBL1 or /ENBL2 being low.

Fast mode selects a bit cell time of 2 uS instead of 4 uS. The 7/8 mHz descriptor indicates whether the input clock (FCLK) is to be divided by 7 or 8 to provide 1 uS internal timings.

When the test mode bit is a 1, device operation is unspecified, except that status register bit 5 can always be read and that the mode register can always be set.

Status register (a read-only register)

bit	function
Ø-4	same as mode register
5	1 = either /ENBL1 or /ENBL2 is currently active (low)
6	1 = MZ (reset to Ø by /RESET and MZ-reset)
7	1 = SENSE input high; Ø = SENSE input low

The MZ bit is reserved for compatibility with future products and should always be read as a zero.

Handshake Register (a read only register)

bit	function
Ø-5	reserved for future use (currently read as ones)
6	1 = write state (cleared to Ø if a write underrun occurs)
7	1 = write data buffer register ready for data

Data Register

The operation of the data register depends on the setting of state bits L6 and L7 and on the synchronous mode bit. With L6 and L7 clear, the data register operates as a read data register. With L7 set the data register operates in the write state as a write data buffer.

Maximum Ratings

supply voltage	-Ø.3 to +7.Ø V
input voltage	-Ø.3 to +7.Ø V
storage temperature	-35 to +125 degrees C
operating temperature	Ø to +7Ø degrees C (ambient)

DC characteristics

sym	parameter	min	max	units	notes
Vcc	supply voltage	4.75	5.25	Volts	
Icc	supply current	—	2ØØ	mA	5
Vil	Input low	—	Ø.8	V	3
Vih	Input high	2.Ø	—	V	4
Il	input leakage	—	1ØØ	uA	1
Vol	TTL output low	—	.4	V	
Voh	TTL output high	2.4	—	V	
Ioh	source current at Voh	.32	—	mA	2
Iol	sink current at Vol	3.2	—	mA	2

Notes

1. Inputs.

   The inputs have static protection. All IWM inputs and bidirectional lines in the input mode are high impedance except as noted below:

   WPROT and /RDDATA : pullup to VCC of 1ØK ohms nominal
   (source current of 8Ø to 6ØØ uA at Ø.4 to 2.8 Volts)

2. Outputs Ioh and Iol apply to DØ thru D7 and WRDATA. The following output lines have special drive capabilities, noted below.

/ENBL1, /ENBL2:	Sink current of at least 5.Ø mA at Vol Source current of at least 4Ø uA at Voh
/WRREQ:	Sink current of at least 1Ø.Ø mA at Vol Source current of at least 4Ø uA at Voh
PHASEØ-3:	Source current of at least 1.Ø mA at Voh Source current of Ø.5 mA when pulled down to 3.Ø V. Sink current of at least 2.4 mA at Vol

3. TTL Vil is also referred to as a zero.
4. TTL Vih is also referred to as a one.
5. at 5.25 V over full operating temperature range.

AC characteristics

sym	parameter	min	max	units	notes
tas	AØ-A3 to /DEV, fe	4Ø	—	nS	setup
tah	/DEV, re to AØ-A3 invalid	-1		nS	addr hold
tds	data to (Q3 OR /DEV), re	5Ø	—	nS	setup
tdh	(Q3 OR /DEV), re to data	1Ø	—	nS	data hold
tda	/DEV, fe to data out	—	2ØØ	nS	access, 1.
tdsl	/DEV low	2ØØ	—	nS	6.
tdsh	/DEV high	45Ø	—	nS	
tde	/DEV to /ENBLx or /WRREQ	—	5ØØ	nS	2.
tdph	/DEV to PHASEx	—	5ØØ	nS	2.
trdh	RDDATA high time	3ØØ	—	nS	5.
tckh	FCLK high time	5Ø	2ØØ	nS	
tckl	FCLK low time	5Ø	2ØØ	nS	
tckp	FCLK period	12Ø	143	nS	
tckpt	FCLK period, no timer	12Ø	5ØØ	nS	11.
tqdsl	Q3, re to /DEV, fe	1	1ØØ	nS	7.
tqdsh	Q3, re to /DEV, re	1	1ØØ	ns	7.
tq3h	Q3 high	26Ø	3ØØ	nS	7.
tq3l	Q3 low	19Ø	—	nS	7.
tdmsbh	DØ-6 valid to D7 re	5Ø	—	nS	8.
tres	/RESET low time	5ØØ	—	nS	1Ø.
trwrh	/RESET to /WRREQ high	—	3ØØ	nS	
tsj	sampling jitter	—	12.5	nS	4.
tckwr	write clock, re to WRDATA	—	5ØØ	nS	9.
twrj	write data jitter	—	62.5	nS	9.
cp	pin capacitance	—	15	pF	

notes

1. Load = 13Ø pF and 8 LS TTL loads
2. Load = 1ØØ pf and rated maximum current
3. fe = falling edge (TTL high to low)
   re = rising edge
4. tsj is the uncertainty window in sampling the asynchronous input RDDATA and synchronizing it internally with CLK at any constant Vcc and temperature.
5. trdh and trdl must be at least twice the period of CLK to be properly synchronized.
6. the time between 2 succesive /DEV selects will be greater than 2 CLK periods, and in synchronous mode will be no less than 1 Q3 period. /DEV may be held low indefinitly.
7. These apply to the synchonous mode only. In other modes Q3 may be held low indefinitly.
8. If, when /DEV is low, data on DØ-7 is changing to a word with D7 high, the data on DØ-6 must become valid before the rising edge of D7 .
9. tckwr is the time from FCLK, re, in asynchronous mode, or Q3, re, in synchronous mode, to changes in the output WRDATA, driving a load of 1ØØ pF.
   twrj is the change in tckwr from edge to edge of WRDATA at any constant Vcc and temperature.
10. 1Ø. for test purposes tres must be at least 24 times tckp .
11. tckpt is max FCLK period with 1-second timer disabled.

This specification is confidential to Apple and contains proprietary information.

changes from earlier specs

A∅-3 pinout corrected
/ENBL1 and /ENBL2 pinout corrected
tdmsbh specified
latch mode added (change to breadboard also)

---

changes from rev#11 spec (4/16/82)

D∅-7 pinout changed to facilitate IC layout
write jitter specified

---

changes from rev#12 spec (5/5/82)

Address setup time changed to 4∅ nS

---

changes from rev#14 spec (6/4/82)

LMotor-On different from Motor-On
Phase lines sink 2.4 mA
tres test condition added
underrun and other nomenclatured cleared up

---

changes from rev#17 spec (8/17/82)

---

read data stall time changed to 4 CLKs
relaxed AC and DC characteristics

rhn ∅2/∅5/82
bcs ∅1/11/82
ws 1∅/2∅/81
woz 1978,79

4. Outputs (8)

WRDATA

The serial data output. A transition occurs on this output for each one bit.

/ENBL1 , /ENBL2

Programmable buffered output lines.
No more than one enable may be low at any time. If an enable is low than Motor-On is true.
If the 1-second on board timer is enabled then the selected one will stay low for about 1 second after it is programmed high.

/WRREQ

This signal is a programmable buffered output line.

PHASEØ-3

These are programmable output lines.
A true TTL logic "1" (2.4 volts) can be maintained even while driving two darlington inputs in parallel.

Register Description

State Register

This is an 8-bit write-only pseudo-register. The bits in this register are individually addressed by A3,A2,A1 . The data on AØ is latched into the addressed state bit by /DEV low. All eight state bits are reset to Ø by /RESET low.

Not only do the state bits control certain chip functions and outputs, the setting of two of the state bits L6 and L7, and Motor-On, internally select which register is to be selected and whether the operation is to be a read or a write. If an operation occurs that changes the state of one of these bits to a new state, that new state will select the register to be accessed during that operation and whether that operation is to be a read or a write.

Address	Name	Function
Ø		A 1 in this bit will drive PHASEØ to a high state.
1		PHASE1
2		PHASE2
3		PHASE3
4	LMotor-On	A 1 on LMotor-On sets the enable selected below low
5	Drive-Sel	A 1 on this bit selects /ENBL2; a Ø selects /ENBL1
6	L6	(see description below)
7	L7	(see description below)

The state bits L7 and L6, and Motor-On, select which register is available to be read or written. Other registers are read during any operation in which AØ is a zero. A register is written when both L6 and L7 are set or are being set to 1 and AØ is a one.

The combination of L7 and Motor-On and /underrun enables /WRREQ low.

L7	L6	Motor-On	register operation selected	State Name
Ø	Ø	Ø	read all ones	
Ø	Ø	1	read data register	Read
Ø	1	x	read status register	Write-Protect Sense
1	Ø	x	read write-handshake register	Write
1	1	Ø	write mode register	Mode Set
1	1	1	write data register	Write Load

Mode register (a write only register)

All eight mode bits are reset to Ø by /RESET low.

bit	function
LSB Ø	1 = latch mode (should be set in asynchronous mode)
1	Ø = synchronous handshake protocol; 1 = asynchronous
2	Ø = 1-second on board timer enable; 1 = timer disable
3	Ø = slow mode; 1 = fast mode (2 uS bit cell timing)
4	Ø = 7MHz; 1 = 8MHz (7 or 8 mHz clock descriptor)
5	1 = test mode; Ø = normal operation
6	1 = MZ-reset
MSB 7	reserved for future expansion

In latch mode the msb of the read data is latched internally during /DEV low (this internally latched msb is then used for the determination of a valid data read).

If the 1-second timer bit is a zero then the enable (/ENBL1 or /ENBL2) selected by Drive-Sel will be held low for 2^23 +-1ØØ FCLK periods (about 1 second) after the LMotor-On state bit is reset to zero. If the latch mode bit is set the timer is not guarantied to count up to 2^23. Motor-On is synonomous with either /ENBL1 or /ENBL2 being low.

Fast mode selects a bit cell time of 2 uS instead of 4 uS. The 7/8 mHz descriptor indicates whether the input clock (FCLK) is to be divided by 7 or 8 to provide 1 uS internal timings.

When the test mode bit is a 1, device operation is unspecified, except that status register bit 5 can always be read and that the mode register can always be set.

Status register (a read-only register)

bit	function
Ø-4	same as mode register
5	1 = either /ENBL1 or /ENBL2 is currently active (low)
6	1 = MZ (reset to Ø by /RESET and MZ-reset)
7	1 = SENSE input high; Ø = SENSE input low

The MZ bit is reserved for compatibility with future products and should always be read as a zero.

Handshake Register (a read only register)

bit	function
Ø-5	reserved for future use (currently read as ones)
6	1 = write state (cleared to Ø if a write underrun occurs)
7	1 = write data buffer register ready for data

Data Register

The operation of the data register depends on the setting of state bits L6 and L7 and on the synchronous mode bit. With L6 and L7 clear, the data register operates as a read data register. With L7 set the data register operates in the write state as a write data buffer.

Maximum Ratings

supply voltage	-Ø.3 to +7.Ø V
input voltage	-Ø.3 to +7.Ø V
storage temperature	-35 to +125 degrees C
operating temperature	Ø to +7Ø degrees C (ambient)

DC characteristics

sym	parameter	min	max	units	notes
Vcc	supply voltage	4.75	5.25	Volts	
Icc	supply current	—	2ØØ	mA	5
Vil	Input low	—	Ø.8	V	3
Vih	Input high	2.Ø	—	V	4
Il	input leakage	—	1ØØ	uA	1
Vol	TTL output low	—	.4	V	
Voh	TTL output high	2.4	—	V	
Ioh	source current at Voh	.32	—	mA	2
Iol	sink current at Vol	3.2	—	mA	2

Notes

1. Inputs.

   The inputs have static protection. All IWM inputs and bidirectional lines in the input mode are high impedance except as noted below:

   WPROT and /RDDATA : pullup to VCC of 1ØK ohms nominal
   (source current of 8Ø to 6ØØ uA at Ø.4 to 2.8 Volts)

2. Outputs Ioh and Iol apply to DØ thru D7 and WRDATA. The following output lines have special drive capabilities, noted below.

/ENBL1, /ENBL2:	Sink current of at least 5.Ø mA at Vol Source current of at least 4Ø uA at Voh
/WRREQ:	Sink current of at least 1Ø.Ø mA at Vol Source current of at least 4Ø uA at Voh
PHASEØ-3:	Source current of at least 1.Ø mA at Voh Source current of Ø.5 mA when pulled down to 3.Ø V. Sink current of at least 2.4 mA at Vol

3. TTL Vil is also referred to as a zero.
4. TTL Vih is also referred to as a one.
5. at 5.25 V over full operating temperature range.

AC characteristics

sym	parameter	min	max	units	notes
tas	AØ-A3 to /DEV, fe	4Ø	—	nS	setup
tah	/DEV, re to AØ-A3 invalid	-1		nS	addr hold
tds	data to (Q3 OR /DEV), re	5Ø	—	nS	setup
tdh	(Q3 OR /DEV), re to data	1Ø	—	nS	data hold
tda	/DEV, fe to data out	—	2ØØ	nS	access, 1.
tdsl	/DEV low	2ØØ	—	nS	6.
tdsh	/DEV high	45Ø	—	nS	
tde	/DEV to /ENBLx or /WRREQ	—	5ØØ	nS	2.
tdph	/DEV to PHASEx	—	5ØØ	nS	2.
trdh	RDDATA high time	3ØØ	—	nS	5.
tckh	FCLK high time	5Ø	2ØØ	nS	
tckl	FCLK low time	5Ø	2ØØ	nS	
⊗ tckp	FCLK period	12Ø	143	nS	
tckpt	FCLK period, no timer	12Ø	5ØØ	nS	11.
tqdsl	Q3, re to /DEV, fe	1	1ØØ	nS	7.
tqdsh	Q3, re to /DEV, re	1	1ØØ	ns	7.
tq3h	Q3 high	26Ø	3ØØ	nS	7.
tq3l	Q3 low	19Ø	—	nS	7.
tdmsbh	DØ-6 valid to D7 re	5Ø	—	nS	8.
tres	/RESET low time	5ØØ	—	nS	1Ø.
trwrh	/RESET to /WRREQ high	—	3ØØ	nS	
tsj	sampling jitter	—	12.5	nS	4.
tckwr	write clock, re to WRDATA	—	5ØØ	nS	9.
twrj	write data jitter	—	62.5	nS	9.
cp	pin capacitance	—	15	pF	

notes

1. Load = 13Ø pF and 8 LS TTL loads
2. Load = 1ØØ pf and rated maximum current
3. fe = falling edge (TTL high to low)
   re = rising edge
4. tsj is the uncertainty window in sampling the asynchronous input RDDATA and synchronizing it internally with CLK at any constant Vcc and temperature.
5. trdh and trdl must be at least twice the period of CLK to be properly synchronized.
6. the time between 2 succesive /DEV selects will be greater than 2 CLK periods, and in synchronous mode will be no less than 1 Q3 period. /DEV may be held low indefinitly.
7. These apply to the synchonous mode only. In other modes Q3 may be held low indefinitly.
8. If, when /DEV is low, data on DØ-7 is changing to a word with D7 high, the data on DØ-6 must become valid before the rising edge of D7 .
9. tckwr is the time from FCLK, re, in asynchronous mode, or Q3, re, in synchronous mode, to changes in the output WRDATA, driving a load of 1ØØ pF.
   twrj is the change in tckwr from edge to edge of WRDATA at any constant Vcc and temperature.
1Ø. for test purposes tres must be at least 24 times tckp .
11. tckpt is max FCLK period with 1-second timer disabled.

This specification is confidential to Apple and contains proprietary information.

changes from earlier specs

A∅-3 pinout corrected
/ENBL1 and /ENBL2 pinout corrected
tdmsbh specified
latch mode added (change to breadboard also)

---

changes from rev#11 spec (4/16/82)

D∅-7 pinout changed to facilitate IC layout
write jitter specified

---

changes from rev#12 spec (5/5/82)

Address setup time changed to 4∅ nS

---

changes from rev#14 spec (6/4/82)

LMotor-On different from Motor-On
Phase lines sink 2.4 mA
tres test condition added
underrun and other nomenclatured cleared up

---

changes from rev#17 spec (8/17/82)

---

read data stall time changed to 4 CLKs
relaxed AC and DC characteristics

rhn ∅2/∅5/82
bcs ∅1/11/82
ws 1∅/2∅/81
woz 1978,79

DRAWING NUMBER 343-0041-A | SH 1 OF 14

REV	ZONE	ECO #	REVISION	APPD	DATE
A		C288	INITIAL RELEASE		
A		F115	CHANGED TEST PROGRAM LIMITS	BOB BAILEY	8/16/83
A		F221	PAGE 2: ADDED WARNING SECTION	BOB BAILEY	2/22/84

TOLERANCES
UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES
DECIMALS .X ± ——
.XX ± ——
.XXX ± ——
ANGLES XX.X ± ——
FRACTIONS ± ——
DIMENSIONS IN PARENTHESIS ARE IN MILLIMETERS

DRAWN BY A. BADER DATE 1/83
CHECKED BY DATE 1/83
APPROVED BY DATE 1/83
RELEASED BY DATE 2/

apple computer inc.

TITLE
IC, INTEGRATED,
WOZ MACHINE (IWM)

SIZE A | DRAWING NUMBER 343-0041-A

MATERIAL: ——
FINISH: ——
NEXT ASSY.
SCALE: ——
SHEET 1 OF 14

Integrated WOZ Machine (IWM)

Device Specification

Features

- Backwards-compatible with 16 sector Disk][controller
- Use of 7M (or 8 mHz) to minimize sampling error rate
- Fast mode using 2 uS bit cells
- Asynchronous mode with pollable handshake registers

WARNING

The IWM will not work reliably in synchronous mode with the 6502 using Apple II disk software. It will work with the CMOS version done by Western Design Center (NCR supplies this part as NCR65C02).

During the load accumulator instruction, if the D7 output of the IWM changes to 1 just before the end of the microprocessor read window, the NEGATIVE flag may be set but A[7] remain 0. This results in recognition of a "valid" byte. The accumulator will contain the byte except A[7] will be incorrect The difference between the 6502 and the 65C02 is that the latter uses regenerative feedback to force the latched input data to valid logic levels after the read window ends. This guarantees the NEGATIVE flag and A[7] will be equal after the load accumulator instruction.

The 6502 may be used if A[7] is not used in the de-nibblizing process. A simple fix to the current routines is to force A[7] to a 1 by ORA #$80 after reading the nibble.

General Description

The IWM is an integration of the Disk II floppy disc interface. When the IWM is reset, it becomes a controller compatible with the current Disk II interface in its operation with currently supported Apple II and /// software. In addition, the IWM has extensions including a status register, mode register, and multiple modes of operation. The IWM provides an asynchronous mode, a fast mode with a data rate twice that of Disk II, and an optional 1 second one-shot timer to hold the enable outputs low.

The IWM is a peripheral device that connects to a host data bus. The device generates and receives serial GCR encoded data. A programmable digital one shot is used for serial data recovery. The IWM generates buffered drive enables and phase line control outputs.

Packaging and Pin Assignment

The IWM is packaged in a standard 28 pin, 600 mil plastic DIP.

```
              +--------u--------+
PHASE 0       | 1             28|     PHASE 1
PHASE 2       | 2             27|     PHASE 3
A0            | 3             26|     Vcc
A1            | 4             25|     Q3
A2            | 5             24|     FCLK
A3            | 6             23|     /RESET
/DEV          | 7             22|     RDDATA
WRDATA        | 8             21|     SENSE
/WRREQ        | 9             20|     /ENBL1
D0            |10             19|     /ENBL2
D1            |11             18|     07
D2            |12             17|     D6
D3            |13             16|     D5
GND           |14             15|     D4
              +-----------------+
```

Technical Description

The primary purpose of the IWM is to allow a microprocessor to read and write serial GCR (group code) encoded data. The IWM may be controlled by setting state bits and reading or writing registers. Setting a state bit and accessing a register is done simultaneously. The registers are the mode register, the status register, the write-handshake register, the read data register, and the write data register. The modes selected by the mode register include synchronous or asynchronous mode and slow or fast mode.

The data format is an 8 bit nibble with the MSB set. The MSB of the 8 bit data nibble is shifted in or written out first. A bit is transferred every bit cell time. The bit cell time defaults to 4 uS (set to 2 uS in fast mode). Therefore the data rate is one nibble every 32 uS (16 uS in fast mode). When writing data out, a one is written as a transition on the WRDATA output at a bit cell boundary time and a zero is written as no transition.

The IWM is put into the write state by a transition from the write protect sense state to the write load state. In the synchronous mode, the time of that transition and every 8 Q3 periods (4 uS) thereafter, until L7 is cleared, marks the beginning of a write window. The duration of the write window is 4 periods of the Q3 input signal (2 uS). The data written at the last write access occurring within this write window will load the shift register with the data to be shifted out. If the next write access has not occured 32 uS (64 Q3 periods) after a load, the write will be extended in multiples of 4 uS (8 Q3 periods) until another write access, and zeros will be shifted out.

In synchronous mode, Q3 clock input is used internally to generate the 32 and 4Ø uS timings, which would then be 64 and 3Ø of the Q3 clock input periods in duration, respectively, and the bit cell timings, 8 Q3 periods per bit cell time in slow mode.

In asynchronous mode the write shift register is buffered and, when the buffer is empty, the IWM sets the MSB of the write-handshake register to a one to indicate that the next data nibble can be written to the buffer. The buffer register may be written at any time during the write state. Only the data last written into the buffer register, before the contents of the buffer register is transfered to the write shift register, is used.

In asynchronous mode CLK is used to generate the bit cell timings. In fast mode the CLK clock is equivalent to the clock input on FCLK. In slow mode CLK is equivalent to the clock input on FCLK divided by two. Therefore, in 7M and slow mode the bit cell time will be 28 FCLK clock input periods in duration, in 8M and slow mode the cell time will be 32 periods, and in 8M and fast mode the cell time will be 16 periods. In asynchronous mode the write shift register is loaded every 8 bit cell times starting seven CLK periods after the write state begins.

An underrun occurs when data has not been written to the buffer register between the time the write-handshake bit indicates an empty buffer and the time the buffer is transferred to the write shift-register. If an underrun occurs in asynchronous mode /WRREQ will be disabled (set to a TTL high state) and the /underrun flag will be set to zero. This occurrance can be detected by reading the write-handshake register before clearing state bit L7. Clearing state bit L7 will reset the /underrun flag.

When L6 and L7 are both zero the IWM is in the read state. When reading serial data, a falling transition within a bit cell window is considered to be a one, and no falling transition within a bit cell window is considered to be a zero. The receive data input on RDDATA is synchronized internally with the CLK clock. The synchronized falling transition is then discriminated to the nearest bit cell window using the 7/8 mHz FCLK clock signal in fast mode and the FCLK signal divided by two in slow mode. A digital one-shot data recovery scheme is used. Every falling transition establishes the bit cell windows, used by the data seperator in the IWM to recover the following bits, until another falling transition is received.

In the read state the data is shifted into the LSB of the shift register, and the shift register shifts data from LSB to MSB. A full data nibble is considered to be shifted in when a one is shifted into the MSB. When a full data nibble is shifted into the internal shift register, the data will be latched by the read data register and the shift register will be cleared to all zeros so that it will then be ready to shift in the next data word.

In the synchronous mode the shift register is readable in any intermediate state with this exeception : when a one is shifted into the MSB, the shift register will appear, to the data bus, to be stalled for a period of two bit times plus four CLK periods. This is to allow the host processor time to poll the MSB to determine when data is valid. In asynchronous mode the data register will latch the shift register when a one is shifted into the MSB and will be cleared 14 FCLK periods (about 2 uS) after_a valid data read takes place (a valid data read being defined as both /DEV being low and D7 (the msb) outputting a one from the data register for at least one FCLK period).

Read data bit cell windows

mode		Nclks	period	data	notes
slow	7M	7-2Ø	FCLK/2	1	2.Ø-5.71+ uS
		21-34		Ø1	6.Ø-9.71+ uS
		35-48		ØØ1	1Ø.Ø-13.71+ uS
					window is 28 clks
slow	8M	8-23	FCLK/2	1	
		24-39		Ø1	
		4Ø-55		ØØ1	
fast	7M	7-2Ø	FCLK	1	window is 14 clks
		21-34		Ø1	
		35-48		ØØ1	
fast	8M	8-23	FCLK	1	1.Ø-2.875+ uS
		24-39		Ø1	3.Ø-4.75+ uS
		4Ø-55		ØØ1	5.Ø-6.875+ uS
					window is 16 clks

The table above shows how the data separator in the IWM discriminates between ones and zeros when reading. Nclks is the number of clock periods between falling transitions of the internally synchronous version of RDDATA. The clock period is either that of the FCLK input or that of the FCLK input divided by two in slow mode. Each falling transition resets the read data windows for subsequent data to be relative to that transition. The data patterns noted above are the bit patterns that are shifted in as a result of the transitions and the absence of transitions in their respective windows.

In port operation, which is asynchronous mode true and latch mode false with /DEV held low indefinitely, read data will appear and change as if the IWM were being continually read. In port operation the MSB can be used to continously clock data into external registers. The MSB will be cleared at least six FCLK periods before being set. Except in port operation, in asynchronous mode the latch mode bit should be set (for reliability in clearing the data register after a read).

Data written to the IWM is sampled by the zaro to one transition of the logical OR of Q3 and /DEV . In asynchronous mode the Q3 input may be tied low.

Signal and Bus operation Description

1. Vcc — +5 volt supply
   GND — Ground reference and negative supply

2. Device Control Signals

   A1-A3 — These three inputs select one of the 8 bits in the state register to be updated.

   AØ — The data input to the state bit selected by A1-A3. The state to which the addressed state bit is set by an operation will select the register to be accessed by that operation.

   Also the /READ input. A low on this input enables the IWM to send the register selected by ıe state onto the data bus.

   DØ-D7 — The bidirectional data bus .

   /DEV — Active low device enable. The falling edge of /DEV latches information on AØ-A3. The rising edge of the logical function (Q3 OR /DEV) qualifies write register data.

   FCLK — Clock input for the serial data logic; either 7 or 8 MHz.

   Q3 — 2.Ø Mhz clock input used to qualify the timing of the serial data being written out in the synchronous mode.

   /RESET — Active low system reset input.
   When asserted, this signal places all IWM outputs in their inactive state, and sets the state and the modes to their defaults.

3. Inputs (2)

   RDDATA — The serial data input. The falling transition of each pulse is synchronized by the IWM.

   SENSE — An input to the IWM that can be polled via the status register.

4. Outputs (8)

WRDATA

The serial data output. A transition occurs on this output for each one bit.

/ENBL1 , /ENBL2

Programmable buffered output lines.
No more than one enable may be low at any time. If an enable is low than Motor-On is true.
If the 1-second on board timer is enabled then the selected one will stay low for about 1 second after it is programmed high.

/WRREQ

This signal is a programmable buffered output line.

PHASEØ-3

These are programmable output lines.
A true TTL logic "1" (2.4 volts) can be maintained even while driving two darlington inputs in parallel.

Register Description

State Register

This is an 8-bit write-only pseudo-register. The bits in this register are individually addressed by A3,A2,A1 . The data on AØ is latched into the addressed state bit by /DEV low. All eight state bits are reset to Ø by /RESET low.

Not only do the state bits control certain chip functions and outputs, the setting of two of the state bits L6 and L7, and Motor-On, internally select which register is to be selected and whether the operation is to be a read or a write. If an operation occurs that changes the state of one of these bits to a new state, that new state will select the register to be accessed during that operation and whether that operation is to be a read or a write.

Address	Name	Function
Ø		A 1 in this bit will drive PHASEØ to a high state.
1		PHASE1
2		PHASE2
3		PHASE3
4	LMotor-On	A 1 on LMotor-On sets the enable selected below low
5	Drive-Sel	A 1 on this bit selects /ENBL2; a Ø selects /ENBL1
6	L6	(see description below)
7	L7	(see description below)

The state bits L7 and L6, and Motor-On, select which register is available to be read or written. Other registers are read during any operation in which AØ is a zero. A register is written when both L6 and L7 are set or are being set to 1 and AØ is a one.

The combination of L7 and Motor-On and /underrun enables /WRREQ low.

L7	L6	Motor-On	register operation selected	State Name
Ø	Ø	Ø	read all ones	
Ø	Ø	1	read data register	Read
Ø	1	x	read status register	Write-Protect Sense
1	Ø	x	read write-handshake register	Write
1	1	Ø	write mode register	Mode Set
1	1	1	write data register	Write Load

Mode register (a write only register)

All eight mode bits are reset to Ø by /RESET low.

bit	function
LSB Ø	1 = latch mode (should be set in asynchronous mode)
1	Ø = synchronous handshake protocol; 1 = asynchronous
2	Ø = 1-second on board timer enable; 1 = timer disable
3	Ø = slow mode; 1 = fast mode (2 uS bit cell timing)
4	Ø = 7MHz; 1 = 8MHz (7 or 8 mHz clock descriptor)
5	1 = test mode; Ø = normal operation
6	1 = MZ-reset
MSB 7	reserved for future expansion

In latch mode the msb of the read data is latched internally during /DEV low (this internally latched msb is then used for the determination of a valid data read).

If the 1-second timer bit is a zero then the enable (/ENBL1 or /ENBL2) selected by Drive-Sel will be held low for 2^23 +-1ØØ FCLK periods (about 1 second) after the LMotor-On state bit is reset to zero. If the latch mode bit is set the timer is not guaranteed to count up to 2^23. Motor-On is synonomous with either /ENBL1 or /ENBL2 being low.

Fast mode selects a bit cell time of 2 uS instead of 4 uS. The 7/8 mHz descriptor indicates whether the input clock (FCLK) is to be divided by 7 or 8 to provide 1 uS internal timings.

When the test mode bit is a 1, device operation is unspecified, except that status register bit 5 can always be read and that the mode register can always be set.

Status register (a read-only register)

bit	function
∅-4	same as mode register
5	1 = either /ENBL1 or /ENBL2 is currently active (low)
6	1 = MZ (reset to ∅ by /RESET and MZ-reset)
7	1 = SENSE input high; ∅ = SENSE input low

The MZ bit is reserved for compatibility with future products and should always be read as a zero.

Handshake Register (a read only register)

bit	function
∅-5	reserved for future use (currently read as ones)
6	1 = write state (cleared to ∅ if a write underrun occurs)
7	1 = write data buffer register ready for data

Data Register

The operation of the data register depends on the setting of state bits L6 and L7 and on the synchronous mode bit. With L6 and L7 clear, the data register operates as a read data register. With L7 set the data register operates in the write state as a write data buffer.

Maximum Ratings

supply voltage	-Ø.3 to +7.Ø V
input voltage	-Ø.3 to +7.Ø V
storage temperature	-35 to +125 degrees C
operating temperature	Ø to +7Ø degrees C (ambient)

DC characteristics

sym	parameter	min	max	units	notes
Vcc	supply voltage	4.75	5.25	Volts	
Icc	supply current	—	60	mA	5
Vil	Input low	—	Ø.8	V	3
Vih	Input high	2.Ø	—	V	4
Il	input leakage	—	1ØØ	uA	1
Vol	TTL output low	—	.4	V	
Voh	TTL output high	2.4	—	V	
Ioh	source current at Voh	.32	—	mA	2
Iol	sink current at Vol	3.2	—	mA	2

Notes

1. Inputs.

The inputs have static protection. All IWM inputs and bidirectional lines in the input mode are high impedance except as noted below:

WPROT and /RDDATA : pullup to VCC of 1ØK ohms nominal
(source current of 8Ø to 6ØØ uA at Ø Volts with VCC = 5.25V

2. Outputs Ioh and Iol apply to DØ thru D7 and WRDATA. The following output lines have special drive capabilities, noted below.

/ENBL1, /ENBL2: Sink current of at least 5.Ø mA at Vol
Source current of at least 4Ø uA at Voh

/WRREQ: Sink current of at least 1Ø.Ø mA at Vol
Source current of at least 4Ø uA at Voh

PHASE 0, PHASE 2
PHASE 3 Source current of at least 1.Ø mA at Voh
Source current of Ø.5 mA when pulled down to 3.Ø V.
Sink current of at least 2.4 mA at Vol

PHASE 1: Source current of at least 1.0 mA at Voh
Source current of 0.5 mA when pulled down to 3.0V
Sink current of at least 12.4 mA at Vol

3. TTL Vil is also referred to as a zero.

4. TTL Vih is also referred to as a one.

5. At 5.25 V over full operating temperature range.

AC characteristics

sym	parameter	min	max	units	notes
tas	AØ-A3 to /DEV, fe	4Ø	—	nS	setup
tah	/DEV, re to AØ-A3 invalid	-1		nS	addr hold
tds	data to (Q3 OR /DEV), re	5Ø	—	nS	setup
tdh	(Q3 OR /DEV), re to data	1Ø	—	nS	data hold
tda	/DEV, fe to data out	—	180	nS	access, 1.
tdsl	/DEV low	2ØØ	—	nS	6.
tdsh	/DEV high	45Ø	—	nS	
tde	/DEV to /ENBLx or /WRREQ	—	5ØØ	nS	2.
tdph	/DEV to PHASEx	—	5ØØ	nS	2.
trdh	RDDATA high time	3ØØ	—	nS	5.
tckh	FCLK high time	5Ø	2ØØ	nS	
tckl	FCLK low time	5Ø	2ØØ	nS	
tckp	FCLK period, with timer	12Ø	143	nS	
tckpt	FCLK period, no timer	12Ø	5ØØ	nS	11.
tqdsl	Q3, re to /DEV, fe	1	200	nS	7.
tqdsh	Q3, re to /DEV, re	1	200	ns	7.
tq3h	Q3 high	26Ø	3ØØ	nS	7.
tq3l	Q3 low	19Ø	—	nS	7.
tdmsbh	DØ-6 valid to D7 re	5Ø	—	nS	8.
tres	/RESET low time	5ØØ	—	nS	1Ø.
trwrh	/RESET to /WRREQ high	—	3ØØ	nS	
tsj	sampling jitter	—	10	nS	4.
tckwr	write clock, re to WRDATA	—	5ØØ	nS	9.
twrj	write data jitter	—	15	nS	9.
cp	input pin capacitance	—	15	pF	

notes

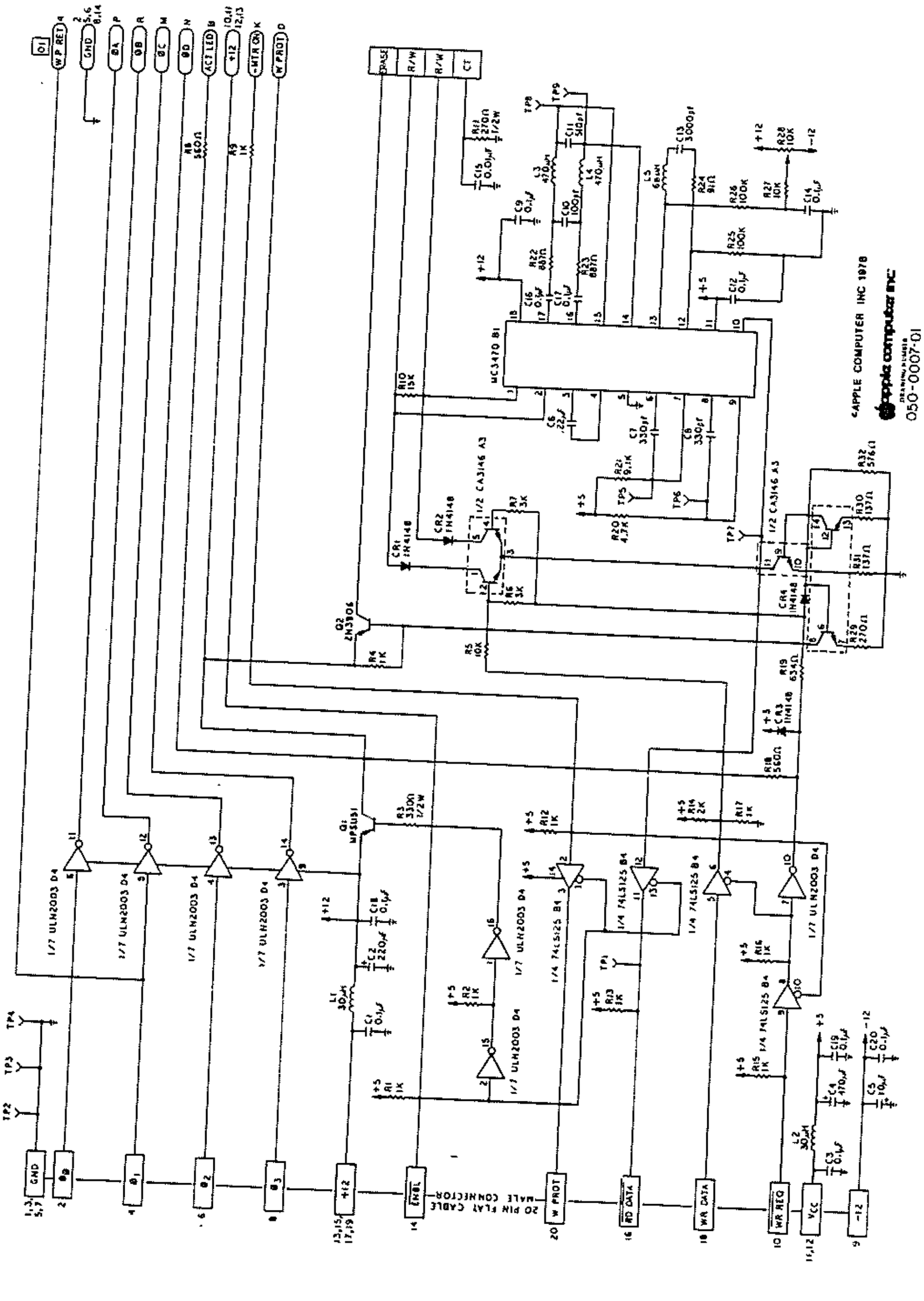
1. Load = 13Ø pF and 8 LS TTL loads
2. Load = 1ØØ pf and rated maximum current
3. fe = falling edge (TTL high to low)
   re = rising edge
4. tsj is the uncertainty window in sampling the asynchronous input RDDATA and synchronizing it internally with CLK at any constant Vcc and temperature.
5. trdh and trdl must be at least twice the period of CLK to be properly synchronized.
6. the time between 2 succesive /DEV selects will be greater than 2 CLK periods, and in synchronous mode will be no less than 1 Q3 period. /DEV may be held low indefinitly.
7. These apply to the synchonous mode only. In other modes Q3 may be held low indefinitly.
8. If, when /DEV is low, data on DØ-7 is changing to a word with D7 high, the data on DØ-6 must become valid before the rising edge of D7 .
9. tckwr is the time from FCLK, re, in asynchronous mode, or Q3, re, in synchronous mode, to changes in the output WRDATA, driving a load of 1ØØ pF.
   twrj is the change in tckwr from edge to edge of WRDATA at any constant Vcc and temperature.
1Ø. for test purposes tres must be at least 24 times tckp .
11. jckpt is max FCLK period with 1-second timer disabled.

This specification is confidential to Apple and contains proprietary information.

```
changes from earlier specs
----------------------
  AØ-3 pinout corrected
  /ENBL1 and /ENBL2 pinout corrected
  tdmsbh specified
  latch mode added (change to breadboard also)
----------------------
changes from rev#11 spec (4/16/82)
----------------------
  DØ-7 pinout changed to facilitate IC layout
  write jitter specified
----------------------
changes from rev#12 spec (5/5/82)
----------------------
  Address setup time changed to 4Ø nS
----------------------
changes from rev#14 spec (6/4/82)
----------------------
  LMotor-On different from Motor-On
  Phase lines sink 2.4 mA
  tres test condition added
  underrun and other nomenclatured cleared up
----------------------
chaoges from rev#17 spec (8/17/82)
----------------------
  read data stall time changed to 4 CLKs
  relaxed AC and DC characteristics
```

rh̄n Ø2/Ø5/82
bcs Ø1/11/82
ws 1Ø/2Ø/81
woz 1978,79

CIRCUIT SCHEMATIC: DISK II ANALOG BOARD



Reading

Reads occur with MOTOR-ON=1 and L6=0, L7=0. The read logic is not reset and therefore, one byte must be read in before ~~it~~ random data is flushed out.

The input signal from RDDATA is synchronized to FCLOCK. Negative transitions are detected and generate a pulse of 1 FCLOCK period. These pulses generate 1's which are fed into the read data shift register. If another negative transition does not occur within a specified period, a 0 is fed into the read data shift register. However, if a negative transition does occur it shifts in a 1 and resets the timer. Group code has 2 important rules: 1) the leading bit (MSB) is 1 and, 2) there is a maximum of 2 consecutive zeroes. Therefore, all legal bit sequences can be analyzed by the patterns 1, 01, and 001 where the leading bit (MSB) is shown on the right.

After a negative transition a 0 is inserted into the shift register if the next negative transition is 24 FCLOCKS or more later. If it is less than 24 FCLOCKS another 1 is ~~shifted~~ inserted and the timer reset. If a 0 is inserted another 0 will be inserted if the next negative transition does not occur for 16 more FCLOCKS. Thus, if negative transitions of RDDATA are spaced by

0 - 23 FCLOCKS	DATA = 11
24 - 39 FCLOCKS	01
40 - 55 FCLOCKS	001

The read data shift register shifts each time a 0 or 1 is fed to it. Once the MSB has shifted through to the far end the shift register resets to all zeroes in order to receive the next byte. It will begin shifting in new bits when it is fed a 1.

In asynchronous mode the read data shift register parallel loads the read data register once the MSB=1 reaches the end of the shift register. The full byte is now available for reading. Port mode should always be used with asynchronous mode. In port mode the MSB read on D7 will be stable when read because the MSB is latched at the beginning of the read data command. This insures D7 will meet the set-up and hold requirements of the processor. Furthermore, if D7 is read as 1 then D0-D6 will be valid and stable. Once D7 is read as 1 by the processor the MSB of the read data register will reset 14 FCLOCKS (1.75 usec) later. This time is not affected by fast/slow mode.

In synchronous mode the read data register is loaded each time the read data shift register shifts. However, once the MSB=1 arrives at the far end of the shift register the read data register will not re-load for 2 data shifts + 4 FCLOCKS (8 FCLOCKS in slow mode). This is to ensure that the correct byte is in the read data register long enough to be seen by the processor but not long enough so as not to be seen as a valid byte twice. The rising edge of D7 is delayed so that if the processor sees D7=1 it is guaranteed that D0-D6 will have been correctly written into a processor register.

Bob Bailey 29 November 1983

Asynchronous writes are entered with motor-on=1 going through the sequence set L6, set L7. This causes the pre-write pre-set pulse followed by the release reset of all write logic. The write data command is generated which loads data from the Data Bus to the write data buffer. This buffer in turn is parallel loaded into the write data shift ~~register.~~ Eight +/- 1/2 FCLOCKS after set L7 the parallel load ends. After 8 more FCLOCKS the MSB of the byte to be written (which is 1) will cause the WRDATA output to toggle from 1 to 0. Subsequently the write data shift register shifts and the WRDATA output toggles if the shifted bit is 1. The shift and toggle are separated by 8 FCLOCKS. After all 8 bits are shifted the write data shift register is loaded. This parallel load ends 8 x 16 FCLOCKS after the last parallel load ended. ¶ The handshake bit is set by the end of the parallel load and reset by the write data command. If it is 1 it means the write data buffer is available since its contents have been loaded into the write data shift register. If the handshake bit is 0 the write data buffer is not available since data has been written into the buffer but has not yet completed loading into the write data shift register.

no ¶ The handshake is read by the read status command which is prompted by CLR L6. This is part of a polling loop so the processor can poll until the handshake is 1 indicating the write data buffer is available. Then SET L7 will cause the write data command ~~and~~ to write a byte to the write data buffer. ¶ The underrun bit indicates ~~that~~ whether or not a new byte has been written into the write data buffer before the end the parallel load. If no new data is available an underrun occurs and the /WRREQ output will go to 1 before the next transition ~~or~~ of WRDATA occurs. This disables the write head before the old byte is rewritten. The underrun bit is read by the read status command. It is cleared while exiting the write mode by CLR L7.

Synchronous writes are entered with motor-on=1 through the sequence set L6, set ~~L6~~ L7. Unlike asynchronous write which uses FCLOCK to generate the time base, synchronous writes are done using Q3. Thus to have 2us bit cells instead of 4us requires asynchronous mode. The data to be written must be brought in with ~~the~~ set L7 which generates the write data command or during the next Q3 as is the case in false reads. The data will be parallel loaded from the write data buffer to the write data shift register. This will end 4-5 Q3 periods after the first time L7 is set. Two Q3 periods later the MSB of the data byte (which is required to be 1) will cause the WRDATA output to toggle from 1 to 0. The shifting and toggling are each updated every 8 Q3 periods. Each time a new byte is written the write data shift register goes into the load mode but the shift and toggle time bases are unaffected. It is required that a write device occurs every 8 Q3 periods. If a write data command does not occur, zeroes will be shifted out and not transitions of WRDATA will occur. Exit by CLR L7, CLR L6.

Bob Bailey

29 November 1983

Commands

Commands generate control line pulses which are only as wide as DEV. These control reading and writing of registers. They are decoded by L6, L7, and DMOTOR-ON. DMOTOR-ON includes the 1 second timer when enabled. When L6 or L7 are changed the decoder uses the new state. Thus, during a DEV the address A0-A3 is asserted, ~~L6~~ L6 or L7 is changed, the decode is made, a command is generated, data is read into the Data Bus D0-D7 or written from the Data Bus to a register. A0 must be 0 to enable a read command. This corresponds to clearing a state. A0 must be 1 to enable a write command. This corresponds to setting a state.

The DMOTOR-ON input to the decoder comes from the same line that enables the selected drive motor but it has some added delay so it changes when the device is de-selected. This insures that DMOTOR-ON is always stable during DEV, only one command will occur. For example, if L4 is set MOTOR-ON becomes 1, but DMOTOR-ON will remain 0 during that DEV and be 1 throughout the next DEV. Do not use the transition of L4 to make a command occur during that DEV, but consider the motor to be either on or off and L6, L7 to the transitioning inputs.

(ENABLE 1, ENABLE 2)

Bob Bailey

29 November 1983

SOFTWARE CONTROL OF THE DISK II OR IWM CONTROLLER

PREPARED BY

NORMAN LEUNG

APRIL 26, 1984

REVISION 1, MAY 10, 1984

SOFTWARE CONTROL OF THE DISK II OR IWM CONTROLLER

Each of the eight expansion slots of the Apple II computer has the exclusive use of sixteen memory locations for I/O control. These memory locations can be used as software switches. A software routine can instruct the interface card in a particular slot to perform a predefined hardware task by toggling a software switch. Whenever the Apple II addresses one of the sixteen I/O locations allocated to a particular slot, the signal on pin 41 of that slot, called DEVICE SELECT/, switches to the active low state. This signal, in conjunction with the four low-order address lines (A0 - A3), can be used to enable logic in the peripheral card to perform a particular task. The following table illustrates the memory space for the sixteen I/O locations for each expansion slot.

TABLE 1: PERIPHERAL CARD I/O SPACE

SLOT	LOCATIONS
0	$C080 - $C08F
1	$C090 - $C09F
2	$C0A0 - $C0AF
3	$C0B0 - $C0BF
4	$C0C0 - $C0CF
5	$C0D0 - $C0DF
6	$C0E0 - $C0EF
7	$C0F0 - $C0FF

One common method of accessing peripheral card I/O control softswitches through software is to use the Apple's (6502) indexed addressing mode. For example, a LDA $C080, X instruction can be used to access softswitch 0 in any slot, if the index register X is loaded with a value equal to the slot number times sixteen.

Sixteen peripheral I/O addresses are used to control the functions of the disk II controller. The following table illustrates the functions of the sixteen software switches.

TABLE 2: DISK II CONTROLLER SOFTSWITCHES

ADDRESS	FUNCTION
$C080, X	PHASE 0 OFF
$C081, X	PHASE 0 ON
$C082, X	PHASE 1 OFF
$C083, X	PHASE 1 ON
$C084, X	PHASE 2 OFF
$C085, X	PHASE 2 ON
$C086, X	PHASE 3 OFF
$C087, X	PHASE 3 ON
$C088, X	TURN MOTOR OFF
$C089, X	TURN MOTOR ON
$C08A, X	SELECT DRIVE 1
$C08B, X	SELECT DRIVE 2
$C08C, X	Q6L
$C08D, X	Q6H
$C08E, X	Q7L
$C08F, X	Q7H

The index register X has the value of slot number times 16. The last four addresses have the following functions.

Q6	Q7	FUNCTION
L	L	READ
H	L	SENSE WRITE PROTECT OR PREWRITE STATE
L	H	WRITE
H	H	WRITE LOAD

In general, any valid 6502 instruction can be used to access the above softswitch address, except a load instruction is used to read a byte of encoded data from the controller and a store instruction is used to write a byte of encoded data to the controller. Below are typical examples demonstrating the use of the disk II controller softswitches. It is assumed that both Q6 and Q7 are low at the beginning of the read/write examples.

SELECT DRIVE

```
LDA    $C08A, X            SELECT DRIVE 1
LDA    $C08B, X            SELECT DRIVE 2
```

The hardware design of the controller allows only one drive to be selected at one time. A LDA $C08A, X instruction will select drive 1 and deselect drive 2. A LDA $C08B, X instruction will select drive 2 and deselect drive 1.

MOTOR ON

```
LDA    $C089, X            TURN MOTOR ON
LDA    $C088, X            TURN MOTOR OFF
```

It should be noted that there is only one interface signal (ENABLE/), going from the controller to each floppy disk drive, which is used to enable the drive's read/write function and to turn on the motor. Both the select drive and the motor on instructions must be executed in order to activate the ENABLE/ signal of a particular drive. A typical program will select the drive first and then turn on the motor at a later time. After the completion of the motor on instruction, the program should wait at least 1 second for the motor to come up to speed, before read/write functions can be performed reliably.

The disk II controller hardware will keep the ENABLE/ signal to its active low state for approximately one second after the execution of the motor off instruction, therefore read/write can be performed reliably within this period. To be on the safe side, the program should verify that the motor is spinning by monitoring the change in data pattern read from the drive. This delay in turning off the motor facilitate rapid and repeat access to the same drive.

SENSE WRITE PROTECT

```
LDA    Q6H, X     WRITE PROTECT SENSE MODE
LDA    Q7L, X     READ CONTROLLER STATUS REG.
BMI    WRPROT     BRANCH IF BIT 7 OF STATUS REG. IS HIGH
```

The above instruction will load the content of the controller's status register into the accumulator. Bit 7 of the status register is the write protect flag. A "one" in bit 7 of the status register indicates that a write protected diskette is inserted into the drive. A BMI instruction will check the write protect flag. The program will branch to the WRPROT address label if the write protect flag is set.

READ A DATA BYTE

```
                LDA    Q7L, X       MAKE SURE IN READ MODE
      LOOP      LDA    Q6L, X       READ THE BYTE
                BPL    LOOP         STAY IN THE LOOP IF THE M.S. BIT IS LOW
                 .
                 .
                 .
                 .
      LP        LDA    Q6L, X       READ ANOTHER BYTE
                BPL    LP           SAY IN THE LOOP IF THE M.S. BIT IS LOW
                 .
                 .
```

NOTE: THERE SHOULD BE NO PAGE CROSSING FOR THE BPL INSTRUCTIONS.

The LDA Q7L, X instruction makes sure that the controller is in read mode. The LDA Q6L, X instruction loads the contents of the controller's data shift register into the accumulator. Since the Apple GCR code requires that the most significant bit of every encoded data byte is high, the BPL instruction will force the program to stay in a two instruction loop until the M.S. bit of the controller data shift register is high. At the beginning of every byte time, the controller internal logic will clear the data shift register. As data bits are shifted into the data shift register, the M.S. bit of the register is high, if and only if a full byte of data is assembled in the register. The data byte will stay in the register for a little more than 7 us. Therefore, it is important to make sure that the BPL instruction does not cross the page boundary. This is necessary to ensure that the execution time of the two instruction loop (LDA, BPL) is no more than 7 us.

WRITE A DATA BYTE

```
                           LDA    Q6H, X       GO TO
                           LDA    Q7L, X       PREWRITE STATE
                           LDA    DATA
___________________________STA    Q7H, X       PARALLEL LOAD DATA INTO CONTROLLER
                           LDA    Q6L, X       CONTROLLER SHIFT DATA OUT SERIALLY
   EXECUTION TIME           .
OF THESE INSTRUCTIONS       .
   MUST BE EXACTLY          .
   32 CLOCK CYCLES          .
___________________________STA    Q6H, X       PARALLEL LOAD ANOTHER BYTE
                           LDA    Q6L, X       SHIFT OUT DATA
                            .
                            .
                            .
                           LDA    Q7L, X       OUT OF WRITE MODE
                           LDA    Q6L, X       TO READ MODE
```

NOTE: It is important to write a garbage byte (HEX FF) before turning off the write mode, so that the drive electronics has enough time to write the last valid data byte.

The first two instructions force the controller into the prewrite state. These are the same instructions to sense the write protect flag. It is important to execute these two instructions even the programmer does not want to sense the write protect flag. The STA instruction loads the contents of the accumulator into the controller's data shift register. The next instruction [LDA Q6L, X] causes the data in the register to shift out serially. Q6H and Q7H are the conditions required for parallel loading the data into the controller. Shifting out the data serially to the disk drive requires Q6L and Q7H. The first STA instruction sets Q7 high, because the conditions are Q6H and Q7L before executing this instruction. The conditions are Q6L and Q7H before the second STA instruction, therefore the second STA instruction sets Q6H.

The execution time of the instructions between the end of two consecutive parallel load instructions [STA] has to be exactly 32 clock cycles, otherwise invalid data will be written on the diskette. In order to calculate the execution time, it is important to note that the 6502 processor requires one additional execution cycle for branching or indexing operations crossing the page boundary. The program should switch the controller back to the read mode after all the data has been written.

WRITE SELF SYNC BYTE

```
                         LDA   Q6H, X
                         LDA   Q7L, X
                         LDA   #$FF
 ______________________  STA   Q7H, X      PARALLEL LOAD AUTO SYNC BYTE
                         ORA   Q6L, X      START TO SHIFT OUT AUTO SYNC BYTE
    EXECUTION TIME        .                ORA IS USED SO THAT LDA   #$FF IS NOT
 OF THESE INSTRUCTIONS    .                NEEDED TO WRITE THE NEXT SYNC CYCLE
    MUST BE EXACTLY       .
    40 CLOCK CYCLES
 ______________________  STA   Q6H, X      PARALLEL LOAD ANOTHER AUTO SYNC BYTE
                         ORA   Q6L, X      SHIFT OUT SYNC BYTE
    SELF SYNC BYTE        .
    40 CLOCK CYCLES       .
                          .
                         LDA   DATA
 ______________________  STA   Q6H, X      LOAD FIRST DATA BYTE
                         LDA   Q6L, X      SHIFT OUT DATA BYTE
      DATA BYTE           .
    32 CLOCK CYCLES       .
                          .
 ______________________  STA   Q6H, X
                         LDA   Q6L, X
                          .
                          .
                          .
                         LDA   Q7L, X      OUT OF WRITE MODE
                         LDA   Q6L, X      TO READ MODE
```

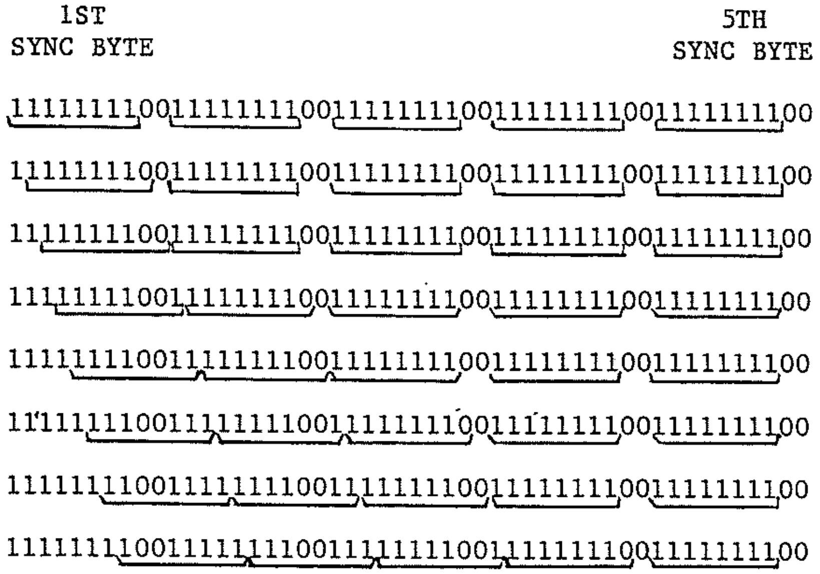
NOTE: Write a garbage byte (HEX FF) before turning off write mode.

The Apple GCR code uses a unique synchronization technique to determine the byte boundary. A self sync byte consists of eight bits of "1" and two bits of "0." The procedure to write a sync byte is the same as to write a data byte, except that the execution time of the instructions between the end of two consecutive parallel load sync byte instructions has to be exactly 40 us. During the write sync byte time, the processor loads eight bits of "1" into the controller. After shifting out 8 bits of "1", the controller hardware will shift out "0" until the next parallel load instruction. Since there are 40 us between two consecutive parallel load instructions and a 4 us bit time, 8 bits of "1" from the processor and 2 bits of "0" appended by the controller hardware are shifted out to the drive. It is necessary to write at least five self sync bytes at the beginning of both the address and data field.

READ SELF SYNC BYTE

Due to the Apple GCR code's unique synchronization technique, the controller hardware will determine the byte boundary automatically. The following is a brief description of the Apple synchronization technique.

FIGURE 1: SYNCHRONIZATION PROCESS



In the above diagram, each row of brackets represent what the controller will send out to the Apple II should the controller start reading at any given bit in the first self sync byte. The controller groups the self sync read data stream into 8-bit byte with a "1" in the most significant bit of each byte. Any "0" bit between bytes are dropped out. From the above diagram, it is shown that the controller is able to group the data at the correct byte boundary within five byte time after the beginning of the read. This is always true for any bit position to start the reading. Therefore, a minimum of five self sync bytes are required for the controller to sync on the read data. After the fifth self sync byte, the controller has established the byte boundary and is able to read the data following the sync bytes correctly. The "D5 AA 96" and "D5 AA AD" address mark sequences follows the self sync bytes in the address and data field respectively. It is not necessary to read and verify the sync byte. In order to read/write a sector, the program should look for the "D5 AA 96" sequence which are the address mark bytes for the address field.

The D5 and AA patterns are reserved for address mark. These patterns are not used to encode data. Therefore, byte syncronization for the address field is achieved by searching for the "D5 AA 96" sequence. Byte synchronization for the data field is done by looking for the "D5 AA AD" sequence.

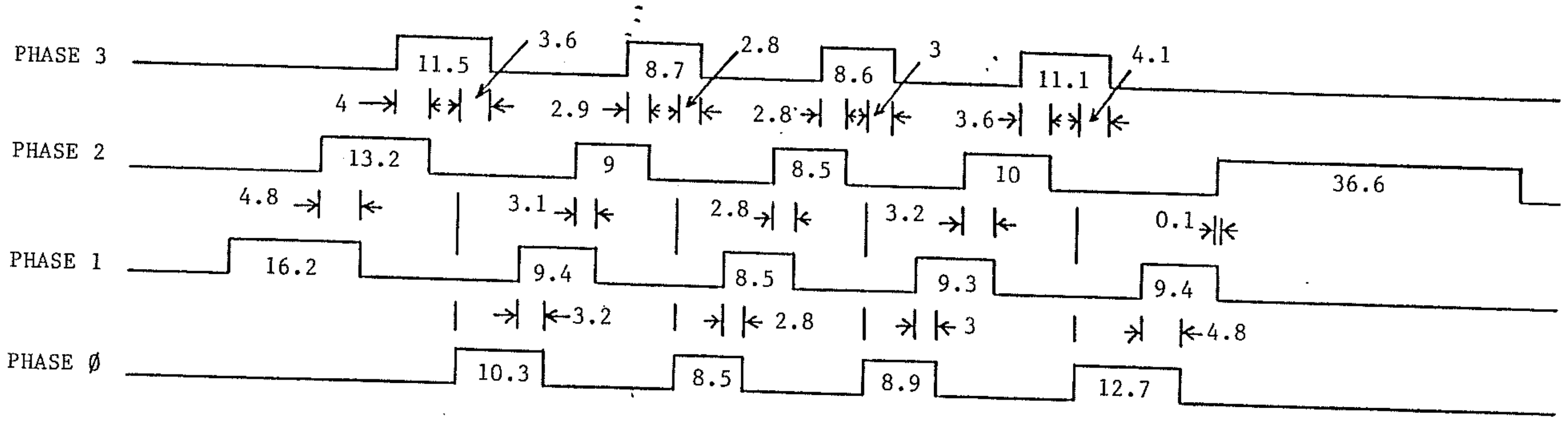
SEEK TO ANOTHER TRACK

The stepper motor in the Disk II is a four phase stepper motor. Eight I/O control softswitches are used to toggle the four phase on and off as shown in table 2. Two adjacent phases have to be activated in sequence in order to move the R/W head to the adjacent track. Activating the phases in ascending order (0, 1, 2, 3, 0, 1, ...) moves the head towards (inward) the center of the diskette. The head moves away (outward) from the center of the diskette when the phases are activated in descending order (3, 2, 1, 0, 3, 2, ...). All even numbered tracks are positioned under phase 0 and all odd numbered tracks are under phase 2. In order to step in a track, the phase 1 and then phase 2 have to be activated in sequence from an even numbered track, while the phase 3 and then phase 0 is activated in sequence from an odd numbered track. The phase 3 and then phase 2 sequence is used to step out a track from an even numbered track. For stepping out a track from an odd numbered track, the phase 1 and then phase 0 sequence is used. The spindle motor should be on for 150 ms before starting the seek operation. The following is an example to step in a track from an even numbered track.

```
 LDA    $C083, X    TURN ON PHASE 1
   .
   .
11.5 msec delay loop
   .
   .
   .
 LDA    $C085, X    TURN ON PHASE 2
   .
   .
 0.1 msec delay loop
   .
   .
   .
 LDA    $C084, X    TURN  OFF PHASE 1
   .
   .
36.6 msec delay loop
   .
   .
   .
 LDA    $C086, X    TURN OFF PHASE 2
```

The above programming example is used to illustrate the timing required to step in a track from an even numbered track. The user may use a indexed look up table for the parameter required for different delay loops. No matter how many tracks to step, the user has to allow the last phase to be on for 36.6 msec, because this timing includes the head settling time requirement (25 ms) of the drive. For long seek (step a number of tracks), two adjacent phases can overlap the phase on time in order to increase the torque of the stepper motor and to reduce the seek time. Since the timing between the phase ON/OFF time is critical, it is recommended that the user calls upon the SEEK routine in the Apple DOS for seeking. Figure 2 shows the waveforms of the phases to seek from track 0 to track 9.

FIGURE 2: PHASE WAVEFORMS TO SEEK FROM TRACK 0 TO TRACK 9



NOTE: ALL THE NUMBERS SHOWN IN THE DIAGRAM ARE IN MILLISECONDS.

TO: Norman Leung   May 31, 1982
     Conrad Chen

From: Peter Baum

1) I think that the document "Software Control of Disk II/IWM Controller" should also contain specific examples in certain section. These examples should be included in the sections where the operation requires time dependent code. The reason I suggest this is that the 65C02 has a few operations that use a different number of cycles then their counterparts in the NMOS 6502. If the developer decides to use his own sequence of instructions, they may not work properly with the other processor. It will also make things easier in the future if we ever decide to use a processor that is similar to the 6502, but has a few operations with different cycle times. A specific example would make things easier on the programmer.

2) It has been brought to my attention by a developer that we don't follow the guidelines exactly for the write self sync byte operation in ProDOS. Here is what I was told:

```
              Last-1 self sync byte
          40<
              Last self sync byte
***--> 36<
              1st address mark byte ($D5)
          32<
              2nd address mark byte ($AA)
          32<
              3rd address mark byte ($AD)
***--> 36<
              1st data nibble
          32<
              2nd data nibble
```

I have not checked this myself because I am not sure where the routines are located in ProDOS. The developer didn't think this would affect the operation of ProDOS, but I think we should check into it just to make sure.

3) I am also unclear from the document where the address mark and data mark fit in. Nowhere in the document does it tell when to write these bytes out, but it does mention something about reading them under the section titled "Read self sync". Specifically, shouldn't this be covered in the section on writing the self sync byte? Currently this section shows how to write the self sync bytes, followed directly by the first data byte. Is that correct

Peter Baum
20525 Mariani Ave.
Cupertino, Ca. 95014

Apple Computer
MS 22-W

May 24, 1984

Differences Between the IWM and the Apple // Disk Controller

The IWM (Integrated Woz Machine) is a single chip (LSI) implementation of the Disk][Controller state machine, designed and patented by Steve Wozniak. It is not an exact replica of the state machine, and though all of Apple's disk operating systems (DOS 3.3, ProDOS) work with it, 3rd party developers will find that some of their software will not work. This document describes all the known differences between the IWM, currrently used in the Apple //c, and the state machine used in the Apple][and //e disk controller card. This document only exists to describe differences and is not intended as a specification for the IWM or Apple // Disk Controller. The correct method of using the disk interface is described in another document, "Software Control of the Disk][or IWM Controller", which is required reading before perusing this document. The document is distributed to registered developers under the auspices of Apple // Developer Technical Support. (For more information contact: Apple Computer at above address - Certification/Registration Program, Mail Stop 23AF). This document will be updated as more differences are uncovered.

The Differences

1) When reading the write protect switch only the most significant bit (MSB) should be tested for the status. On the Disk Controller other bits can be tested, but this will not work on the IWM.

2) The IWM will not work properly in an Apple // if it is placed into a mode where both Q6H and Q7H are set and the motor is off (Q4L). If the chip is placed in this mode at any time then all subsequent operations of the IWM may not work properly.

3) The IWM does not require the false-read cycle, which occurs on indexed write instructions such as STA \$C05E,X, to work properly. The Disk Controller state machine depends on the false read cycle during the STA Q7H,X instruction to store data properly into the shift register. This anamoly means that software which may work with the IWM will not work with the Disk Controller. This type of problem will generally appear if a store absolute instruction is used to set Q7H (STA \$C0EF), since there is no false-read cycle on store absolute operations.

Other Notes of Interest

The IWM has never been sold in a Disk Controller card for the Apple // and there are no plans to do this. All Disk][Controller Cards, sold with either the Disk][or the Duodisk, use the state machine.

In the Apple //c the 65C02 processor, manufactured by GTE and NCR, is used. Some of the instructions use different cycle times than the corresponding instructions in the NMOS 6502 used in the Apple //e and][+. If these instructions are used as part of the 32usec. or 40usec. loops required in the disk write routines, then the write routines will not work properly in both the Apple //e (or][+) and Apple //c.

Other documents and books which may prove helpful:

Beneath Apple DOS, by Don Worth and Pieter Lechner, from Quality Software

Understanding the Apple II, by Jim Sather, from Quality Software

Software Control of the IWM or Disk][Controller , Apple Computer*

The 6502 False-read cycles, Apple Computer*

* -- These documents are distributed to registered developers under the auspices of Apple // Developer Technical Support. (For more information contact: Apple Computer at above address - Certification/Registration Program, Mail Stop 23AF).

```
File:    iwm.infolist                                                   Page  1
Report:  iwm.infolist
Selection: Company Name is not equal to -

Ray Klein
Sunrise Software
10855 Sorrento Valley Rd.
San Diego
Ca.
92121
Larry Roddenberry
-Home w/ Steve Park (Hayes=work)
6448 C Parton Ct.
Norcross
Ga.
30092
David Krankshaw
Scarborough Systems
25 No. Broadway
Tarrytown
N.Y.
10591
```

SEE BACK

Iwm

Addendum to Software Control of the IWM or Disk][Controller

PREFACE

The purpose of this document is to briefly illustrate the instructions needed to operate the disk controller. The examples which are given show only the instructions that are needed to operate the disk drive and the timing sequence of these instruction. An example of executable code has not been given since the operation within the timing loops depends on other factors, such as the current disk function. The operations described include:

Selecting the Drive
Turning the Motor On
Sensing the Write Protect Switch
Reading a Data Byte
Writing a Data Byte
Writing a Self-Sync Byte
Reading a Self-Sync Byte
Seeking to Another Track

This document assumes that the reader has some knowledge of disk formats and operating systems. (A good primer for this type of information is the book 'Beneath Apple DOS'). For example, this document doesn't show how to format a disk, but assumes that the developer knows how a disk is formatted, including the address marks and address field format.

```
OURCE   FILE #01 =>IWM
000:                 1 *
000:                 2 *
000:                 3 * THIS ROUTINE WILL DETERMINE WHETHER A STATE MACHINE DISK II CONTROLLER
000:                 4 * OR AN IWM CONTROLLER IS INSTALLED IN THE SYSTEM.
000:                 5 * UPON EXIT FROM THE ROUTINE, Y=1 MEANS IWM CONTROLLER AND Y=0 MEANS
000:                 6 * STATE MACHINE DISK II CONTROLLER.
000:                 7 *
000:                 8 * ASSUME A MOTOR OFF INSTRUCTION [LDA $C088,X] HAS BEEN EXECUTED FOR
000:                 9 * TWO SECONDS BEFORE THE USER CALLS ON THIS ROUTINE.  OTHERWISE, A
000:                10 * TWO SECOND DELAY LOOP MUST BE ADDED AFTER THE FIRST MOTOR OFF
000:                11 * INSTRUCTION [LDA $C088,X]  AT THE BEGINNING OF THIS ROUTINE.
000:                12 *
000:                13 * THE ENABLED DISK DRIVE WILL CONTINUE TO BE ON FOR 1 SEC
000:                14 * AFTER EXIT FROM THIS ROUTINE.
000:                15 *
000:                16 *
---- NEXT OBJECT FILE NAME IS IWM.0
000:        1000    17           ORG   $1000
000:AE 41 10        18           LDX   SLOTX16       ;X REG=SLOT NO. X 16
003:BD 88 C0        19           LDA   $C088,X       ;MOTOR OFF
006:A0 00           20           LDY   #00           ;CLEAR REG. Y
008:BD 8D C0        21           LDA   $C08D,X       ;Q6H
00B:BD 8F C0        22           LDA   $C08F,X       ;Q7H, ADDRESS MODE REG.
00E:A9 04           23           LDA   #$04
010:9D 8F C0        24           STA   $C08F,X       ;DISABLE TIMER BIT IN MODE REG.
013:BD 8E C0        25           LDA   $C08E,X       ;Q7L, OUT OF WRITE MODE
016:BD 89 C0        26           LDA   $C089,X       ;MOTOR ON
019:48              27 LOOP      PHA                  ;10 MSEC DELAY LOOP
01A:68              28           PLA                  ;WAIT FOR THE MOTOR ON
01B:48              29           PHA                  ;SIGNAL TO BE ACTIVE IN
01C:68              30           PLA                  ;THE CONTROLLER CARD
01D:48              31           PHA
01E:68              32           PLA
01F:48              33           PHA
020:68              34           PLA
021:48              35           PHA
022:68              36           PLA
023:C8              37           INY
024:D0 F3   1019    38           BNE   LOOP          ;END OF DELAY LOOP
026:BD 8E C0        39           LDA   $C08E,X       ;Q7L, READ STATUS REG.
029:9D 88 C0        40           STA   $C088,X       ;MOTOR OFF
02C:29 1F           41           AND   #$1F          ;MASK 5 L.S. BITS
02E:C9 04           42           CMP   #$04          ;CHECK TIMER BIT
030:D0 09   103B    43           BNE   DISKII        ;DISK II CONTROLLER IF NOT EQ
032:C8              44 IWM       INY                  ;INCREMENT Y REG.
033:BD 8F C0        45           LDA   $C08F,X       ;Q7H, ADDRESS MODE REG.
036:A9 00           46           LDA   #$00
038:9D 8F C0        47           STA   $C08F,X       ;Q7H, RESTORE MODE REG.
03B:BD 8E C0        48 DISKII    LDA   $C08E,X       ;Q7L,
03E:BD 8C C0        49           LDA   $C08C,X       ;Q6L, RESTORE TO READ MODE
041:        1041    50 FIN       EQU   *
041:60              51 SLOTX16   DFB   $60
```

```
103B DISKII        ?1041 FIN           ?1032 IWM           1019 LOOP
1041 SLOTX16
* SUCCESSFUL ASSEMBLY := NO ERRORS
* ASSEMBLER CREATED ON 15-JAN-84 21:28
* TOTAL LINES ASSEMBLED    51
* FREE SPACE PAGE COUNT   89
```