Western Digital WD1770/1772 Floppy Disk Controller

Programming information from the data sheets for the
Western Digital 1770-00 and 1772-00 floppy disk drive controllers.

By David Gahris
(dgahris@eece.maine.edu),
Typed and edited on 10 Mar 1995.

Converted to PDF format by LST of LaserForce
(lst@laserforce.org)
on 20 Jan 2001.

Unless I note otherwise, this file applies to both the 1770-00 and the 1772-00, and I do not deal with other chips in the 1770 family, such as the 1773-00 or the 1770-02. I call the 1770 and the 1772 collectively the 177x. Except as noted, "clock cycles" are CPU clock cycles at exactly 1 MHz.

1. REGISTERS

The names are self-explanatory. All registers are 8 bits wide. After the CPU writes to any register, it cannot read that register for 16 cycles.

r0 (*write*) - *Command Register* - When the 177x is busy, it ignores CPU writes to this register UNLESS the new command is a force interrupt.

r0 (*read*) - *Status Register* - The value in this register depends on the previous 177x command. If the 177x receives a Force Interrupt command while it is executing another command, the FDDC will clear the Busy bit (Status bit 0: see later in this paragraph), and leave all other Status bits unchanged. If the 177x receives a *Force Interrupt* command while it is not executing a command, the 177x will update the entire *Status Register* as though it had just executed a Type I command. (For an explanation of Type I, Type II, Type III, and Type IV commands, see the section on commands.) When the CPU is connected (directly or indirectly) to the 177x's interrupt output, it is a bad idea to check the *Busy* bit because a CPU read of the *Status Register* clears the 177x's interrupt output. After the CPU writes to the *Command Register*, it should not attempt to read the *Busy* bit for 24 cycles. The CPU should not follow a *Command Register* write with a read of Status bits 1-7 until 32 clock cycles have elapsed.

Bits:

Bit 7 - *Motor On*. This bit is high when the drive motor is on, and low when the motor is off.

Bit 6 - *Write Protect*. This bit is not used during reads. During writes, this bit is high when the disk is write protected.

Bit 5 - *Spin-up / Record Type*. For Type I commands, this bit is low during the 6-revolution motor spin-up time. This bit is high after spin-up. For Type II and Type III commands, Bit 5 low indicates a normal data mark. Bit 5 high indicates a deleted data mark.

Bit 4 - *Record Not Found*. This bit is set if the 177x cannot find the track, sector, or side which the CPU requested. Otherwise, this bit is clear.

Bit 3 - *CRC Error*. This bit is high if a sector CRC on disk does not match the CRC which the 177x computed from the data. The CRC polynomial is x^16+x^12+x^5+1. If the stored CRC matches the newly calculated CRC, the CRC Error bit is low. If this bit and the Record Not Found bit are set, the error was in an ID field. If this bit is set but Record Not Found is clear, the error was in a data field.

Bit 2 - *Track Zero / Lost Data*. After Type I commands, this bit is 0 if the mechanism is at track zero. This bit is 1 if the head is not at track zero. After Type II or III commands, this bit is 1 if the CPU did not respond to Data Request (Status bit 1) in time for the 177x to maintain a continuous data flow. This bit is 0 if the CPU responded promptly to Data Request.

Bit 1 - *Index / Data Request*. On Type I commands, this bit is high during the index pulse that occurs once per disk rotation. This bit is low at all times other than the index pulse. For Type II and III commands, Bit 1 high signals the CPU to handle the data register in order to maintain a continuous flow of data. Bit 1 is high when the data register is full during a read or when the data register is empty during a write. "Worst case service time" for Data Request is 23.5 cycles.

Bit 0 - *Busy*. This bit is 1 when the 177x is busy. This bit is 0 when the 177x is free for CPU commands.

r1 (r/w) - *Track Register* - The outermost track on the disk is numbered 0. During disk reading, writing, and verifying, the 177x compares the Track Register to the track number in the sector ID field. When the 177x is busy, it ignores CPU writes to this register. The highest legal track number is 240.

r2 (r/w) - *Sector Register* - During disk reading and writing, the 177x compares the Sector Register to the sector number in the sector ID field. When the 177x is busy, it ignores CPU writes to this register. Valid sector numbers range from 1 to 240, inclusive.

r3 (r/w) - *Data Register*

2. COMMANDS

The 177x accepts 11 commands. Western Digital divides these commands into four categories, labeled I,II, III, and IV.

2.1. Type I

Type I commands are *Restore*, *Seek*, *Step*, *Step In* and *Step Out*.

The following table is a bit map of the values to store in the *Command Register*.

Command	Bit 7	B6	B5	B4	B3	B2	B1	Bit 0
Restore	0	0	0	0	h	V	r1	r0
Seek	0	0	0	1	h	V	r1	r0
Step	0	0	1	u	h	V	r1	r0
Step in	0	1	0	u	h	V	r1	r0
Step out	0	1	1	u	h	V	r1	r0

Flags:

u (*Update Track Register*) - If this flag is set, the 177x will update the track register after executing the command. If this flag is cleared, the 177x will not update the track register.

h (*Motor On*) - If the value of this bit is 1, the controller will disable the motor spin-up sequence. Otherwise, if the motor is off when the chip receives a command, the chip will turn the motor on and wait 6 revolutions before executing the command. At 300 RPM, the 6-revolution wait guarantees a one-second start time. If the 177x is idle for 9 consecutive disk revolutions, it turns off the drive motor. If the 177x receives a command while the motor is on, the controller executes the command immediately.

V (*Verify*) - If this flag is set, the head settles after command execution. The settling time is 15 000 cycles for the 1772 and 30 000 cycles for the 1770. The FDDC will then verify the track position of the head. The 177x reads the first ID field it finds and compares the track number in that ID field against the Track Register. If the track numbers match but the ID field CRC is invalid, the 177x sets the CRC Error bit in the status register and reads the next ID field. If the 177x does not find a sector with valid track number AND valid CRC within 5 disk rotations, the chip sets the Seek Error bit in the status register.

r (*Step Time*) - This bit pair determines the time between track steps according to the following table:

r1	r0	1770	1772
0	0	6000 CPU clock cycles	6000 cycles

0	1	12000 cycles	12000 cycles
1	0	20 000 cycles	2000 cycles
1	1	30 000 cycles	3000 cycles

Restore

If the FDDC receives this command when the drive head is at track zero, the chip sets its Track Register to $00 and ends the command. If the head is not at track zero, the FDDC steps the head he head arrives at track 0. The 177x then sets its Track Register to $00 and ends the command. If the chip's track zero input does not activate after 255 step pulses AND the V bit is set in the command word, the 177x sets the Seek Error bit in the status r

Seek command causes the 177x to step the head to the desired track number and update the Track Register.

Step

The 177x

the r flag. [Transcriber's Note: Western Digital assumes in this paragraph that disks do not have more than 77 tracks.]

Step out

The 177x issues one step pulse in the direction toward Track 0 and waits one step time according to the r flag.

2.2. Type II

Type II commands are *Read Sector* *Write Sector*.

Comm	Bit 7	B6		B4	B3		B1	Bit 0
	1	0		m	h		0	0
	1	0		m	h		P	a0

Flags:

m *Multiple Sectors*) If this bit = 0, the 177x reads or writes ("accesses") only one sector. If this bit = 1, the 177x sequentially accesses sectors u - sector command will end prematurely when the CPU loads a Force Interrupt command into the

(*Motor On* - If the value of this bit is 1, the controller will disable the motor spi -up sequence.

wait 6 revolutions before executing the command. At 300 RPM, the 6 revolution wait guarantees a one second start time. If the 177x is idle for 9 consecutive disk revolutions, it turns off the drive motor.

If the 177x receives a command while the motor is on, the controller executes the command immediately.

E *Settling Delay*) If this flag is set, the head settles before command execution. The settling time is

(*Write Precompensation* - On the 1770 02 and 1772-
precompensation. The FDDC delays or advances the write bit stream by one- ighth of a cycle according to the following table :

Previous		Current bit sending	to be sent	Precompensation
	1	1		Early
X		1	1	
0	0		1	Early
	0	0		Late

Programmers typically enable precompensation on the innermost tracks
and bit density is maximal. A 1 value for this flag disables write precompensation.

a0 ( ) -
will write a deleted data mark.

Read Sector

The controller waits for a sector ID field that has the correct track number, sector number, and CRC.

of the CRC. If the controller does not find a sector with correct ID field and address mark within 5 disk

sector into the data register. If there is a CRC error at the end of the data field, th
Error bit in the Status Register and ends the command regardless of the state of the "m" flag.

Write Sector

The 177x waits for a sector ID field with the correct track number, sector number, and CRC. The 177x then counts off 22 bytes
before the end of this 22 byte delay, the 177x ends the command. Assuming that the CPU has heeded the 177x's data request, the controller writes 12 bytes of zeroes. The 177x then w
or deleted Data Address Mark according to the a0 flag of the command. Next, the 177x writes the byte

end of the sector. After the 177x writes the last byte, it calculates and writes the 16 bit CRC. The chip then writes one $ff byte. The 177x interrupts the CPU 24 cycles after it writes the second byte of the CRC.

2.3.

Read Address, , and *Write Track*

d		B6	B5		B3	B2		Bit 0
Read Address		1	0		h	E		0
Read Track		1	1		h	E		0
Write Track		1	1		h	E		0

Flags:

h (*Motor On*) - If the value of this bit is 1, the controller will disable the motor spin-up sequence. Otherwise, if the motor is off when the chip receives a command, the chip will turn the motor on and wait 6 revolutions before executing the command. At 300 RPM, the 6-revolution wait guarantees a one- second start time. If the 177x is idle for 9 consecutive disk revolutions, it turns off the drive motor. If the 177x receives a command while the motor is on, the controller executes the command immediately.

E (*Settling Delay*) - If this flag is set, the head settles before command execution. The settling time is 15 000 cycles for the 1772 and 30 000 cycles for the 1770.

P (*Write Precompensation*) - On the 1770-02 and 1772-00, a 0 value in this bit enables automatic write precompensation. The FDDC delays or advances the write bit stream by one-eighth of a cycle according to the following table :

Previous bits sent		Current bit sending	Next bit to be sent	Precompensation
X	1	1	0	Early
X	0	1	1	Late
0	0	0	1	Early
1	0	0	0	Late

Programmers typically enable precompensation on the innermost tracks, where bit shifts usually occur and bit density is maximal. A 1 value for this flag disables write precompensation.

Read Address

The 177x reads the next ID field it finds, then sends the CPU the following six bytes via the Data Register :

Byte #	Meaning
1	Track
2	Side
3	Sector
4	Sector length code
5	CRC byte 1
6	CRC byte 2

Sector length code	Sector length
0	128
1	256
2	512
3	1024

The 177x copies the track address into the Sector Register. The chip sets the CRC Error bit in the status register if the CRC is invalid.

Read Track

splices or noise may cause the chip to look for an address mark[1]. The chip may read gap bytes incorrectly during write-splice time because of synchronization.

Write Track

This command is the means of formatting disks. The drive head must be over the correct track BEFORE the CPU issues the *Write Track* command. Writing starts with the leading edge of the first index pulse which the 177x finds. The 177x stops writing when it encounters the next index pulse. The 177x sets the Data Request bit immediately after receiving the *Write Track* command, but does not start writing until the CPU loads the Data Register. If the CPU does not send the 177x a byte within three byte times after the first index pulse, the 177x ends the command. The 177x will write all data values from $00 to $f4 (inclusive) and from $f8 to $ff (inclusive) unaltered. Data values $f5, $f6, and $f7, however, have special meanings. The value $f5 means to write an $a1 to the disk. The $a1 value which the 177x writes to the disk will lack an MFM clock transition between bits 4 and 5. This missing clock transition indicates that the next normally written byte will be an address mark. In addition, a Data Register value of $f5 will reset the 177x's CRC generator. A Data Register value of $f6 will not reset the CRC generator but will write a pre-address-mark value of $c2 to the disk. The written $c2 will lack an MFM clock transition between bits 3 and 4. A Data Register value of $f7 will write a two-byte CRC to the disk.

2.4. Type IV

The Type IV command is Force Interrupt.

Force Interrupt

Programmers use this command to stop a multiple-sector read or write command or to ensure Type I status in the Status Register. The format of this command is %1101(I3)(I2)00. If flag I2 is set, the 177x will acknowledge the command at the next index pulse. If flag I3 is set, the 177x will immediately stop what it is doing and generate an interrupt. If neither I2 nor I3 are set, the 177x will not interrupt the CPU, but will immediately stop any command in progress. After the CPU issues an immediate interrupt command ($d8), it MUST write $d0 (*Force Interrupt*, I2 clear, I3 clear) to the Command Register in order to shut off the 177x's interrupt output. After any *Force Interrupt* command, the CPU must wait 16 cycles before issuing any other command. If the CPU does not wait, the 177x will ignore the previous *Force Interrupt* command. Because the 177x is microcoded, it will acknowledge *Force Interrupt* commands only between micro-instructions.

[1] *Transcriber's Note: I do not know how the programmer can tell that the AM detector has found an address mark.*

3. INTERRUPTS

The 177x interrupts the CPU:

- each time it finishes a command
- if a *Force Interrupt* condition is true or
- if the FDDC tries to write to a write-protected disk

The 177x's interrupt output pin resets when the CPU reads the status register or writes to the command register, except after a *Force Interrupt* (immediate) command. See the section on the *Force Interrupt* command.

4. DISK FORMATTING

The 177x formats disks according to the IBM 3740 or System/34 standard. See the *Write Track* command for the CPU formatting method. The recommended physical format for 256-byte sectors is as follows :

Number of Bytes	Value of Byte	Comments
60	$4e	Gap 1 and Gap 3. Start and end of index pulse
12	$00	Gap 3. Start of bytes repeated for each sector.
3	$a1	Gap 3. Start of ID field. See section on *Write Track* command.
1	$fe	ID address mark
1	track #	$00 through $4c (0 through 76)
1	side #	0 or 1
1	sector #	$01 through $10 (1 through 16)
1	length code	See section on *Read Address* command.
2	CRC	End of ID field. See section on *Write Track* command.

22	4^e	Gap 2.
12	$00	Gap 2. During *Write Sector* commands the drive starts writing at the start of this.
3	$a1	Gap 2. Start of data field. See section on *Write Track* command.
1	$fb	data address mark
256	data	Values $f5, $f6, and $f7 invalid. See section on *Write Track* command. IBM uses $e5.
2	CRC	End of data field. See section on *Write Track* command.
24	$4e	Gap 4. End of bytes repeated for each sector. During *Write Sector* commands the drive stops writing shortly after the beginning of this.
668	$4e	Continue writing until the 177x generates an interrupt. The listed byte count is approximate.

Variations in the recommended formats are possible if the following requirements are met:

1. Sector size must be 128, 256, 512, or 1024 bytes.
2. All address mark indicators ($a1) must be 3 bytes long.
3. The $4e section of Gap 2 must be 22 bytes long. The $00 section of Gap 2 must be 12 bytes long.
4. The $4e section of Gap 3 must be at least 24 bytes long. The $00 section of Gap 3 must be at least 8 bytes long.
5. Gaps 1 and 4 must be at least 2 bytes long. These gaps should be longer to allow for PLL lock time, motor speed variations, and write splice time.

The 177x does not require an Index Address Mark.

5. MISCELLANEOUS

The chip steps the drive head in the same direction it last stepped unless the command changes the direction. Each step pulse takes 4 cycles. The 177x begins outputting a direction signal to the drive 24 cycles before the first stepping pulse. The 177x is designed for a head data rate of 250 kilobits per second. If the CPU does not give the 177x a byte by the time the 177x needs to write to the disk, the 177x writes a zero byte instead. The minimum cycle time of the raw head data input is 3 CPU clock cycles. The head data output has a cycle time of 4, 6, or 8 CPU clock cycles. The head data output pulse width is .82 clock cycles for an early bit, .69 cycles for a normal bit, and .57 cycles for a late bit. See the Commands section for information on write precompensation. The 1772-02 is 100% software-compatible with the 1772-00. During DMA between the 177x and CPU RAM, the DRQ (Data Request) pin of the 177x controls data transfer. The DRQ pin is high when the Data Request bit in the Status Register is high, and the DRQ pin is low when the Data Request bit is low.