Chapter 19

Floppy Disk Controller

Without some "mass storage" device such as a floppy disk, even the largest computer would still be just a toy. The SK68K can use either a floppy disk or a hard disk; the floppy disk interface is right on the main SK68K board.

19-1. Discussion

Although we tend to look in awe at the floppy disk circuitry of any computer, actually it is a very simple circuit - because the most complex parts are hidden in a dedicated IC known as the FDC or Floppy Disk Controller.

Fig. 19-1 shows the circuitry. The heart of the circuit is the WD1772 FDC which handles most of the real work of the interface. On the CPU side, the FDC connects to the lower eight bits of the data bus, to address bits A1 and A2, to the R/$\overline{W}$ line, to the $\overline{RESET}$ line, to the $\overline{I/O4}$ select line from the address decoder, and to the 8 MHz clock signal CLK8.

The $\overline{I/O4}$ line comes from U34 in the address decoder (Fig. 10-1). It goes low whenever the 68000 accesses any address in the range from $FE0100 through FE013F, and thus selects the FDC whenever the 68000 does any read or write to any address in this range. But since only the lower eight bits of the data bus go to the FDC, only odd addresses in this range can actually be used for data transfer.

Internally, the CPU-side of the FDC is organized into six *registers* - six groups of flip-flops which store a byte-sized number. Two of these registers can be both written into and read by the 68000; another two can be read but not written, and the last two can be written into but not read. The 68000 selects which register it is reading or writing by (a) making the R/$\overline{W}$ line high for reading or low for writing, and (b) putting the appropriate bit pattern on address lines A1 and A2. These two bits are controlled by

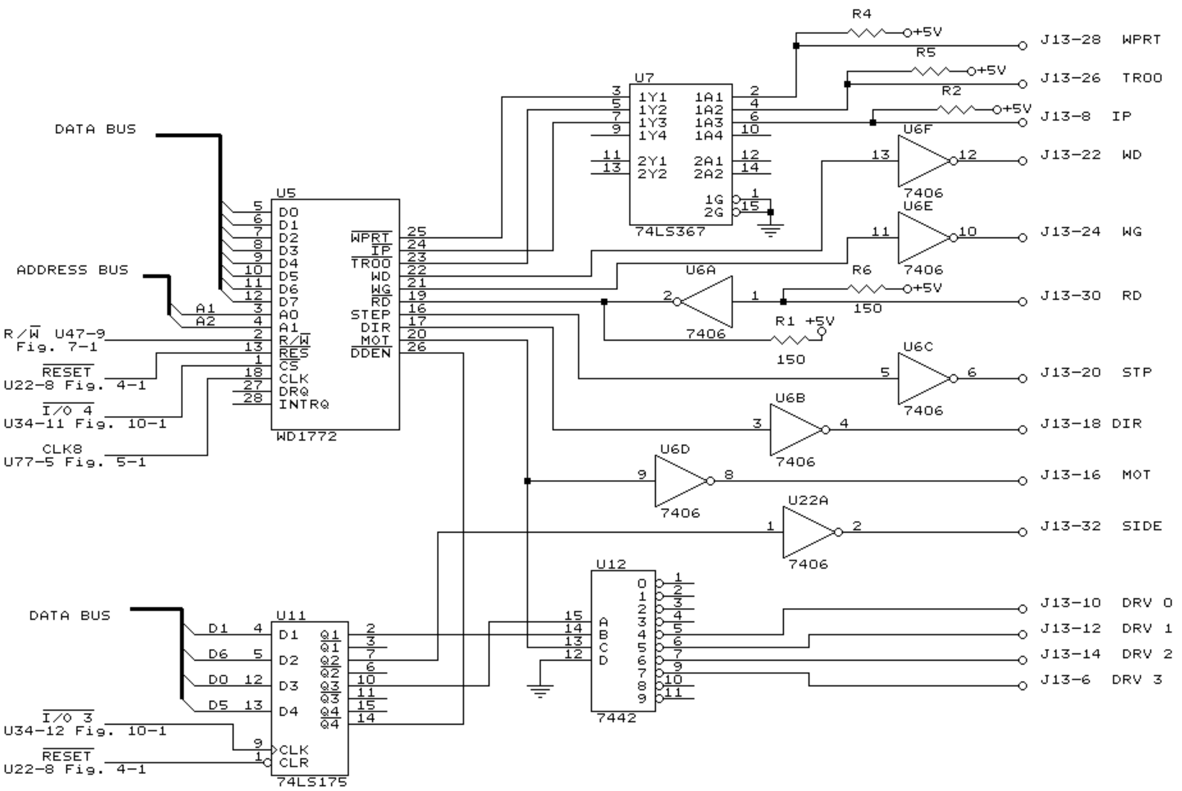

Fig. 19-1. Floppy disk controller.

choosing which address (in the range from $FE0100 through FE013F) the program accesses. For example, writing to location $FE0103 makes R/$\overline{W}$ low and puts the bits 01 on A2 and A1 respectively. (Remember that the 3 at the end of $FE0103 is the bit pattern 0011; the middle two bits of this pattern are A2 and A1, respectively.)

These six registers therefore appear to the 68000 at four address locations as follows:

$FE0101	Control register	Write only
$FE0101	Status register	Read only
$FE0103	Track register	Read / Write
$FE0105	Sector register	Read / Write
$FE0107	Write data register	Write only
$FE0107	Read data register	Read only

Each of the registers has a specific job:

To tell the FDC what to do, the 68000 (or, to be more exact, its program) puts a command into the Control register. The command might tell it to move to a specific track, to read or write a specific sector, or to format a track. (Read Appendix D about disk organization if you are not familiar with how data is stored on a floppy disk.)

To tell the 68000 (or its program) what is going on, the FDC puts status information into the Status register. Each bit in the status register has a function, such as to indicate that the FDC is busy, that it is waiting for data, that the disk is write-protected, or that an error has occurred.

The 68000 tells the FDC where to read or write by placing the track and sector number into the Track and Sector registers.

Finally, the 68000 places data to be written on the disk into the Write data register, or reads data from the Read data register.

Although the FDC IC handles most of the housekeeping involved with reading and writing floppy disks, there are two jobs it does not handle - choosing one out of several drives, or choosing a specific side of a double-sided disk. In addition, the FDC needs an additional input to tell it whether to use single or double density on the disk. All three of these jobs are handled by U11, a quad latch.

As shown in Fig. 19-1, the four data inputs of U11 connect to bits 0, 1, 5, and 6 of the data bus, while its CLK or clock input connects to I/O3. Like I/O4, I/O3 comes from U34 in the address decoder, but this signal is pulsed whenever the 68000 reads or writes to any address in the range of $FE00C0 through FE00FF. More specifically, any time that the 68000 stores a byte into location $FE00C1, U11 is clocked and any data that is on bits 0, 1, 5, or 6 of the data bus is stored into U11. The outputs of U11 are then used as follows: bits 0 and 1 go to U12, a decoder which selects one of four drives; bit 5 is sent to the DDEN pin of the FDC IC to choose either single or double density; bit 6 goes through U22a to the disk drives to select the side.

Finally, let's look at the connections to the disk drives themselves. Disk drives connect to the SK68K computer through a 34-wire flat cable which plugs into J13. Although not shown in Fig. 19-1, all of the odd pins of that connector are grounded, while the even pins carry signals to and from the drives. (On the connector, all the odd pins are on one side of the connector, while in the flat cable the odd-numbered wires alternate with the even-numbered wires. This means that between any two even-numbered signal wires there is always a ground wire which provides some shielding and isolation between signal pins.)

When there is more than one floppy drive, all the drives share the same 34-conductor cable and are connected in parallel. The only signals which are not shared by all drives are the DRV 0 through DRV 3 wires, which are used to select a specific drive. For example, to access drive 0, the SK68K places a low on the DRV 0 line while keeping the other three DRV lines high. When a command is then sent to the four drives, only the one selected drive actually obeys the command.

The signal wires can be grouped into three groups: read or write data, control signals to the drives, and status signals returned by the drives.

The two data lines are RD for Read Data, and WD for Write Data. Although data on the data bus travels in parallel, the FDC IC contains internal shift registers which convert it to and from serial data, which is then sent one bit at a time to or from the drive. Clock pulses are also added by the FDC to make sure that the data on the disk is properly timed and can be correctly read back. In single density, a clock bit is inserted between every data bit; in double density, clock bits are only inserted between consecutive zeroes. By cutting down on the clock bits, double-density operation can store more data bits on each track of the disk.

There are a number of control signals which are sent to the drive to control its operation. We have already discussed the four DRV signals, which select a particular drive, and the SIDE signal which selects the side of the disk. Other control signals include WG, the Write Gate signal, which goes low to tell the drive to switch from reading to writing; STP (step) which tells it to move the head from one track to the next; DIR which tells it which direction to move the head in; and MOT, which tells it to turn the motor on.

By way of explanation, MOT actually controls all drives, so even drives not selected (with a DRV signal) turn their motors on and rotate the disk. The STP signal is so named because floppy drives use a *stepper* motor to move the head back and forth between tracks. A stepper motor has a ratchet-like motion which is so aligned that the head settles over the correct track when the motor clicks into the next position. In this way the precise positioning of the head over a track is handled by the motor without any external help.

Finally, each drive sends back three status signals, all usually derived from photo-electric sensors in the drive. A drive sends back a low on the WPRT (write protect) line if it senses that the write-protect notch on the disk is covered with tape. It sends back a low on the TR00 (track 00) line when it senses that the head has been stepped all the way to track 0, the outermost track on the disk. (Note that the FDC has no easy way of knowing which track the head is positioned over - it keeps track of the head position by sensing track 0, and then keeping a count of the STP step pulses as the head moves in and out.) The IP or Index Pulse signal carries a low pulse every time that a disk rotates so that the beginning of the track is under its head. The drive senses the beginning of the track by sending a beam of light through a small hole in the disk called the *index hole*; once every revolution of the disk, the light is sensed and generates the IP pulse.

The FDC uses the index pulse in several ways. In normal operation, the presence of the pulse signal tells the FDC that there is a disk in the drive and the door is closed (otherwise the disk would not turn). While formatting the disk, it tells the FDC where to begin and end each track. When the FDC encounters a disk error, it also uses the IP signal to count down how many times it retries the operation before it gives up.

19-2. Construction

Install the following components:

U5	WD1772 floppy disk controller, and its socket
U11	74LS175 quad latch, and its socket
U12	7442 BCD-to-decimal decoder, and its socket
U7	74LS367 hex bus driver, and its socket
U6	7406 open-collector hex inverter, and its socket

R1, R2, R4, R5, and R6	150-ohm 1/4-watt resistors
J13	a 34-pin header
C6	0.1 µF disk ceramic capacitor

19-3. Testing

Most 3-1/2" or 5-1/4" floppy disk drives will work with the SK68K, but a double-sided drive (and, if possible, an 80-track drive) is preferable as it will hold more data. In the PC world, these drives are normally identified as 360K for 40-track drives, or 720K for 80-track drives; drives labelled 1.2 meg or 1.44 meg are not suitable.

Note also that the SK*DOS disk operating system is normally supplied on 80-track double-sided disks unless you specify otherwise. An 80-track drive can read 40-track disks by a process called double-stepping, where the drive takes two steps of 1/96" to move the 1/48" spacing between tracks on a normal 40-track disk; SK*DOS automatically tries double-stepping when it encounters a disk error, so reading a 40-track disk in an 80-track drive is totally invisible to the user. But the reverse is not true - although SK*DOS can write a 40-track disk in an 80-track drive, it often happens that this makes the disk unreadable on a 40-track drive. The reason is that the tracks on an 80-track disk are not just closer together, they are also narrower. When an 80-track drive writes on a 40-track disk, it does not write over the full track width and some of the original 40-track data may still remain around the edges of the track. When a 40-track drive subsequently reads the disk, it reads the new data as well as some of the old. Depending on the exact track positioning and other factors, it may then misread the disk and fail.

The next task is to make sure that the jumpers on the drive are properly set. As described above, selecting one out of a possible four drives is done by the four DRV lines. Although all four lines go to each drive, only one DRV signal is actually used by any single drive. Fig. 19-2 shows a simplified diagram of some of the jumpers which control drive selection on a typical floppy drive. As you can see, the four DRV signals appear on pins 6, 10, 12, and 14 of the 34-pin connector, and go to a set of jumpers usually called DS0

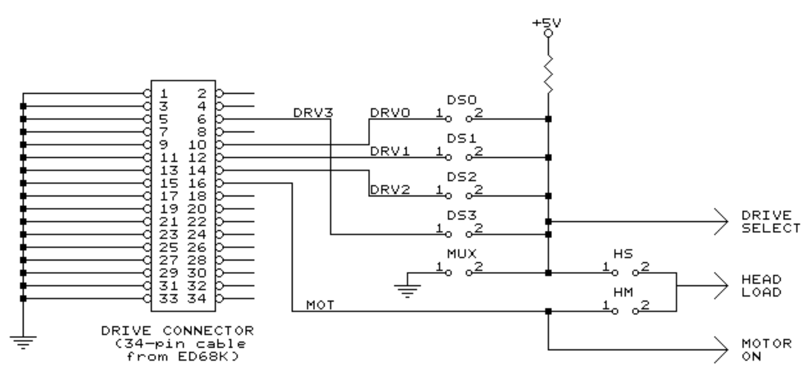

Fig. 19-2. Drive select circuitry in a typical drive.

through DS3 (although sometimes they are labelled DS1 through DS4). When one of these jumpers is installed, then this DRV signal goes to the 'drive select' line on the drive to enable the drive.

To use a single drive with the SK68K, you must make sure that it has only the DS0 jumper installed. In some drives, the connections are made through a shorting plug installed in an IC socket; in this case you may have to break three of the four connections on the shorting plug (or install a small DIP switch instead of the shorting plug). In other cases, you may have to move a jumper from one of the other positions into DS0. (Many floppy drives currently sold for use in IBM PCs or clones have the DS1 jumper installed, since the drive selection in these systems is done by flipping wires in the 34-pin cable, rather than by moving jumpers on the drive.) If you install more than one drive, place each DS jumper in a different position, starting with DS0 for your main drive.

In addition to moving the DS jumpers, you should also remove the MUX (or MX) and HM jumpers if installed, and place a jumper in the HS position. MUX is used to permanently select a drive for those computers which do not provide DRV signals, while the HM and HS jumpers control the 'head load' signal in those drives that have a solenoid which moves the read / write head against the disk. Installing the HM (head with motor) jumper would bring the head against the disk as soon as the motor turns on, whereas installing the HS (head with select) jumper only brings the head against the disk if the drive is selected. (Many modern drives do not have a head load solenoid, in which case this item does not apply.)

One last item on the drive(s) concerns small resistor packs installed in sockets. In order to minimize noise, each of the wires bringing a signal from the computer to the disk drive must be properly terminated with a resistor. Drives usually have small resistor packs which provide the terminations. But to avoid overloading the signals when more than one drive is installed, only the one drive at the very end of the cable should have its resistor pack installed. If there is no terminating resistor pack, or if there are two or more, the disk system may be unreliable, so check each drive in your system. (If you are using used drives, they may be missing the resistor packs altogether.)

Once this is completed, connect the four-pin power plug from the power supply to the drive, and connect the 34-pin cable between the computer and disk drive. Pin 1 of J13 is toward the back of the board, and the pin 1 end of the connector is marked on most disk drives with a small notch between pins.

Finally, insert a blank disk into whichever is drive 0 and turn on the power. Once the HUMBUG program is running, examine the disk drive to check that the motor is off and the drive select LED on the face of the drive is off. If either of these is on, the 34-pin cable is most likely connected backward at one end.

Now type the HUMBUG command FD (floppy disk). Since the drive still contains a blank disk it cannot boot SK*DOS, but still the drive motor should start and the drive select LED should light. If the motor comes on but the LED does not, the DS jumpers may not be properly placed. Don't go on until the motor and LED behave properly.

Once you have made all these checks, reset the computer, remove the blank disk, place a write-protect tape strip on the write-protect notch of your SK*DOS system disk, and place it in the drive. Then type the FD command.

If all goes well, the disk drive should start, the drive select LED should go on, the head should go back and forth a few times, and in a few seconds the SK*DOS signon message should appear. Congratulations - your system is working and almost finished!

19-4. In Case Of Difficulty

If not, then a bit of debugging is in order. First check the type of disk drive - a single-sided drive cannot read a double-sided disk, a 40-track drive cannot read an 80-track disk. If you are really quick, try to count the disk revolutions per second (rps) - the disk should be turning at 5 rps, not 6. If it turns at 6 rps, it may be a 360 rpm 1.2 megabyte drive intended for an AT-type system; if so, all is still not lost since some of these drives have a jumper to select either 300 or 360 rpm.

If the drive seems correct, use HUMBUG's FM command to fill all of memory from address $0800 through $8000 with zeroes, and then try the FD command again. After about 15 seconds, reset the system and use the HA command to look at locations $0800 through 0900. If these locations still contain zeroes, then the disk system totally failed to read the disk. If an oscilloscope or logic probe is available, check that the IP and RD lines are normally high, but have negative-going pulses just after you type the FD command. If not, then the disk may be in the drive backward, the disk drive may still not be properly selected, or may be defective. An easy way to check is to remove the disk, turn off the power, manually (and very carefully - it might be useful to ask a knowledgeable person for help!) move the drive's head carriage a half-inch toward the center of the disk, and then try again. As soon as you type the FD command, the FDC should step the head carriage outward toward track 0; if not, then some of the control signals are either not getting to the drive, or else are being ignored.

If locations $800 and up are now nonzero, then *something* was read from the disk. Check whether the first few bytes at $0800 are 60 08 50 54 32; these are the very first bytes read from track 0 sector 1 of the disk. If this data is there, then the disk system is almost OK. The data read from this very first sector contains the first half of a program we call the *superboot*, and loads into locations $0800 through $08FF. Once loaded, HUMBUG's FD command then jumps to location $0800, and this program loads the very next sector (track 0 sector 2) into memory at location $0900, so check whether the area from $0900 through $09FF is now filled, or whether it still contains zeroes.

The combination of these two sectors is now supposed to load the SK*DOS.SYS system file into memory and jump to its beginning at location $1000. Look at locations $0805 and $0806; these two bytes should contain two nonzero numbers, which tell the program where on the disk to find the SK*DOS.SYS file.

Next, look at locations $1000 through 1100. If these locations still contain zeroes, then the SK*DOS file was not read. If the drive seems able to read the superboot program but fails to read SK*DOS.SYS, there may be a problem with the STP or DIR lines or circuitry so that the drive can read track 0 but not other tracks. This symptom would also appear if you were using a 40-track drive to read an 80-track disk, or a single-sided drive to read a double-sided disk.

Finally, look at memory locations in the range from $4000 to $5000 or so. If these are still zeroes, then an error may have occurred while reading the disk. A common problem is head alignment - the fact that the disk head may not be centered over a track. Try a different drive; another possibility is to copy your SK*DOS system disk on another system and try the copy. For example, SK*DOS disks can be duplicated using the Copy II PC Option Board on PC / XT clones. Alternatively, exchange your SK*DOS disk for a new one.