This document describes the method of interfacing to the 1772 through the DMA CHANNEL on the ATARI ST.

Before any data transfers can begin to or from the floppy, the DMA fifo must be cleared and the DMA address must be set up. To clear the fifo write a 190 to ff8606 followed by 90 to ff8606. Next set up the DMA address as follows:

ff860d is the least significant 8 bits of the address
ff860b is the middle 8 bits of the address
ff8609 is the most significant 8 bits of the address

Addressing the 1772

The 1772 requires two writes to access the registers. The first write (to ff8606) selects the 1772 register. The second access is to the 1772 register.

The addresses for the registers are (in hex):
1. 80 - This selects the command/status register.
2. 82 - This selects the track register.
3. 84 - This selects the sector register.
4. 86 - This selects the data register.

Each of the above values must be or-ed with 100 (hex) if the direction of transfer is from memory to floppy disk. Also a value of ff must be written to 43e to prevent the TOS from changing the value in ff8606. (If ff is in 43e this prevents TOS from modifying the value in ff8606. THIS BYTE MUST BE SET BACK TO 00 AS SOON AS THE DISK OPERATION IS COMPLETED!!)

Seeks

To seek to a track the 1772 first receives the destination track in the data register (86) and then is issued the command seek command (17).

For Example:

```
move.w #$86,ffff8606   ;select the data register
move.w #$4f,ffff8604   ;write destination track (last track on drive)
move.w #$80,ffff8606   ;select command register
nop
nop
                       ;delay because of the 3usec cycle time of
                       ;the 1772 register access
move.w #$17,ffff8604    ;this is a seek with verify with the correct
                       ;step rate
```

The 1772 will generate an interrupt when it has finished the seek. The interrupt can also be polled at fffa01 bit 5. If this bit is a zero then the 1772 is finished. To check for errors read from ff8604. this will clear the interrupt bit.

Transferring Data

To write to the diskette, first set up the DMA address. Next clear the DMA fifo then write 190 to ff8606 followed by 1 to ff8604. The 1 sets an upper limit of 512 bytes that can be transferred. (The maximum 512 byte blocks that can be transferred is ff.) Write the sector (1 - 9) number in the sector register. For a write in tracks 00-27 use a6 for the write command, and a4 for tracks 28-4f. Use 84 (hex) for the read command. Do not use any of the read/write multiple sectors commands since these require a Force Interrupt command which performs much slower than re-executing the read or write command.

Format Track Command

The write track command (f6 for tracks 00 - 27 and f4 for tracks 28 - 4f) writes data to the drive beginning with the index pulse and ending with the index pulse. This requires about 1a00 bytes to fill a drive running at 3%. By using the existing format command each track will have 9 sectors of 512 bytes per sector. The fourth byte of the Id-Field is used by the 1772 to count the number of bytes to transfer and to determine where the CRC is in the data field. Changing a parameter in the Id-Field can be incompatible with the TOS. Valid track numbers are 0-4f, valid sector numbers are 1-9. The side number is ignored. (The side number can be read by issuing the READ ADDRESS command 3 times without clearing the DMA fifo or changing the DMA pointer.)

The entire track can be written as one long sector and then read back using the read track command. (Using this method does not allow any use of the 1772 error checking capability.) When doing this, the first bytes to written after the index pulse should be 00. Allow at least 12 bytes of 00 for lock on time and 3 bytes of f5's for byte synchronization. (The f5's generate a1's with a missing clock pulse to allow the bytes to be aligned.)