COMMODORE 900 FLOPPY SPECIFICATION

Compiled by:

Benny Pruden

and

Frank Hughes

15 March 1985

TABLE OF CONTENTS

LIST OF FIGURES

1. System Overview

The Commodore 900's floppy subsystem provides for one or two high density floppy drives capable of storing 1.2 Mbytes of data. All digital and analog circuitry is included so that only a drive unit with motor control electronics is required. The subsystem makes heavy use of CBM LSI circuits to minimize the cost of the interface. The communication path for both data and commands is the DMA device in the Commodore 900. The data storage method used today is GCR (Group Coded Recording). The subsystem supports the later addition of hardware and software for the MFM format. The floppy drive is a 96 TPI (tracks-per-inch) mechanism. A block diagram of the Commodore 900's floppy subsystem is in figure 1-1.

1.1 Functional Specification

The Commodore 900 floppy disk is an intelligent disk controller subsystem that resides on the main CPU board. The subsystem uses a modified 6508 CPU with RAM, 4K x 8 bit ROM, 74LS245 bus driver, 8723 disk controller IC, M3470 analog read channel IC, a 7406 for LED drive and motor control, and four 75477 stepper drivers. The ROM contains protocol interpreters, DMA interface drivers, and device driver code. The protocol interpreter reads commands from the Z8001's system memory and jumps to device driver section(s) in the ROM that perform the command for the specific drive. The 8723 provides a 1.5 Mhz system clock for the 6508 from the Z8000 6 Mhz clock, address decode for ROM, selects for the 8716 DMA controller, data separation and write precompensation for the Commodore GCR format, and I/O ports to control LED, motor, head select, and steppers.

The interface to the Commodore 900 system is via an 8716 DMA controller that is shared with the hard disk controller. The commands are passed to the subsystem by writing a SCSI like command to a specified memory location in the Z8001's system memory. The Z8001 then addresses I/O location 0x501. This causes an I/O select and interrupts the subsystem via the 8716. The subystem then performs a DMA read of the command area. Next, the subsystem interprets the command and , if it is a valid command for the floppy, DMA's any sector information to or from the physical device. Then a DMA write to the command area sends SCSI like status information to the Commodore 900's memory and causes a vectored interrupt number 80 to the Z8001 via the 8716.

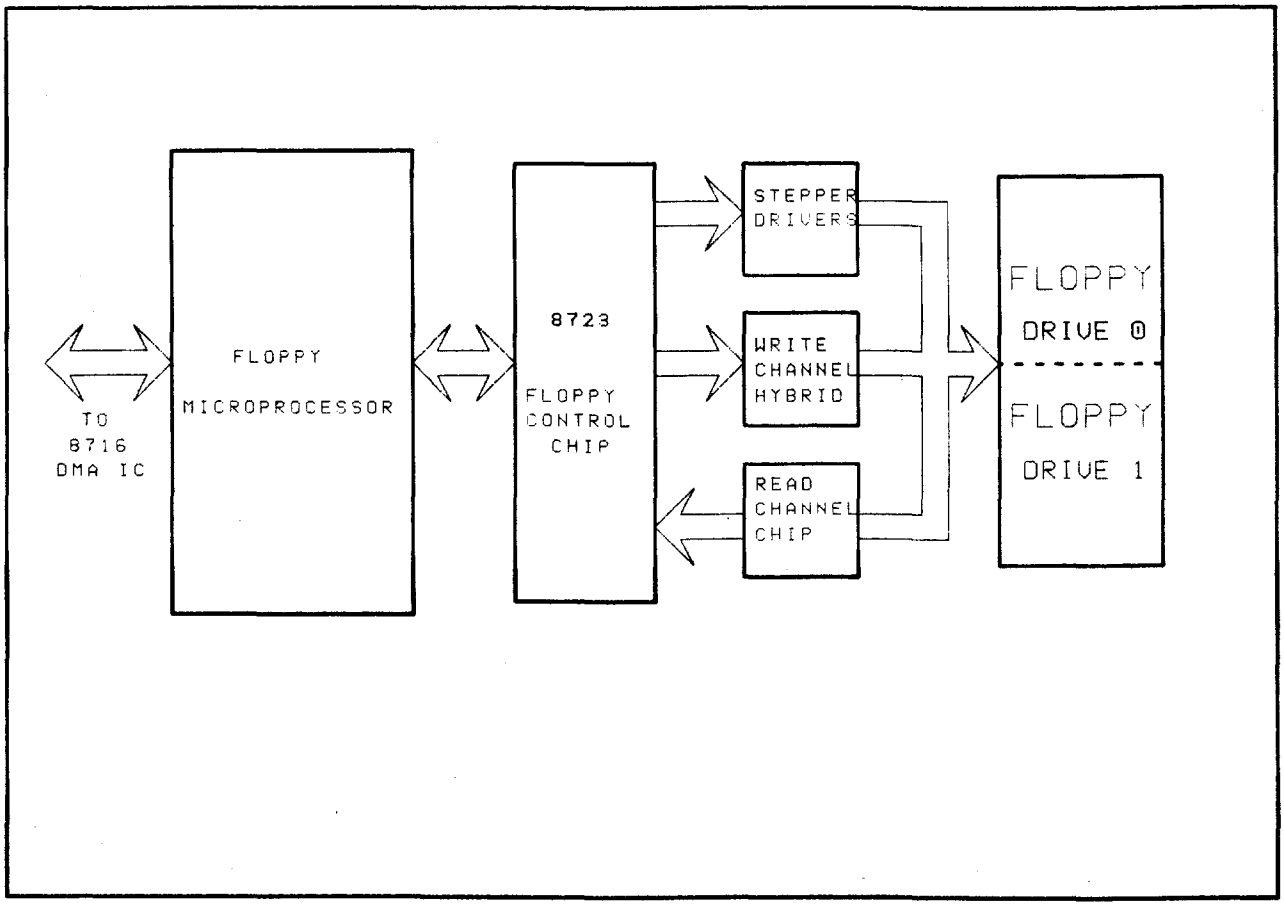

Figure 1-1. Floppy System Block Diagram

2. Hardware

2.1 Custom IC Function

The 8723 custom IC replaces almost all of the digital logic necessary for the floppy disk subsystem. The 8723 provides the following functions:

- o Clock generation for the floppy subsystem's CPU.
- o Data separation of incoming floppy read data.
- o Write precompensation for floppy write data (CBM GCR format).
- o CBM GCR encoding and decoding.
- o Control for steppers, motor, head select, and drive LED's.
- o Address decode for subsystem ROM.
- o Control logic for interface to the Commodore 900's DMA channel.

2.1.1 Clock Generation

The 6Mhz Z8001 clock is divided to 1.5Mhz PH1 and PH2 signals required for the 6508 microprocessor.

2.1.2 Address Decode

The floppy subsystem's memory map appears below. Signals A15, A14, A3, A2, A1, A0, and R/W are decoded to generate the mapping.

Address	Function		
0000	6508 I/O port data direction register		
0001	6508 I/O port		
0002-003F	6508 RAM		
4000	WOHLSSSS	Drive 0 control lines	
	W	Write protect	1=protected
	O	Track 0 detect	1=on track 0
	H	Motor	1=motor on
	L	LED	1=LED on
	SSSS	Stepper control	

(Continued on next page.)

Address	Function		
4001	WOHLSSSS	Drive 1 control lines	
	W	Write protect	1=protected
	O	Track 0 detect	1=on track 0
	H	Motor	1=motor on
	L	LED	1=LED on
	SSSS	Stepper control	
4002	SMWEHDGG	Control lines	
	S	Sync detect / generate	1=detected (read) 1=generate (write)
	M	MFM mode	0=GCR
	W	Write gate	1=on
	E	Erase gate	1=on
	H	Head select	0=lower
	P	PCSD/PCSS enable	0=?
	GG	GCR density region	00=GCR 16 (2.66us)
4003		8716 PCSS register select	
4004		Data to and from disk	
4005		Data with PCSD select	
4006		8716 PCSD register select	
4007	DUWCSCPS	Control lines	
	T	Test Enable	1 = Test mode on
	R	Read Enable (DMA direction)	0 = to Z8001 memory
	W	Write current	
	C	CRC error	
	S	Set CRC	
	C	Clear CRC	
	P	Precomp	
	S	Set overflow	
400C		Test Register (see 8723 spec. for details)	
E000-F000		ROM select	Floppy control ROM

2.1.3 GCR Encode/Decode

When the MFM bit in the control register of the 8723 is low, disk data is modified by the internal GCR encode / decode logic. For a definition of the GCR code, see section 3.1.3.

When the write gate bit of the control register is set high the GCR encode logic is enabled. The GCR 8 to 10 bit conversion is done with GCR PLA #1 and GCR PLA #2 as two seperate 4 to 5 bit conversions. The PLA's are inside the 8723 IC and perform a 4 bit to 4 bit conversion. The fifth bit is routed to the output. An additional input for sync generate forces all the data ouputs of the PLA's high (note: since bit 2 and 6 are routed without conversion, to generate sync, bit 2 and bit 6 should also be high). The 10 bit converted output is then input to the shifters and latched at the next GCR-BRDY (byte ready) output from the divide by 10 counter. GCR-BRDY also signals the 6508 that another byte can be written by creating a negitive edge on the SO (set overflow) pin of the 6508. The shifter is then controlled by the GCR state machine and the three most significant bits are inputs to the state machine. These three bits are used to determine the write precompensation for the middle bit. GCR0 and GCR1 are also inputs to the state machine to determine the bit rate. Data is output on the GCR-D and GCR-ND outputs of the state machine. For a detailed description of the GCR state machine, refer to the 8723 IC specification.

When the write gate bit of the control register is low GCR decode logic is enabled. Data from the disk is syncronized to the 6 Mhz state machine clock and is input to the state machine. This read pulse is sampled at the 6 Mhz state machine clock rate. The state machine performs phase locked data separation for the GCR density region specified by the GCR0 and GCR1 signals. The state machine then clocks data out to the shift registers with the outputs of the shifter connected to the GCR PLA #1 and GCR PLA #2 for GCR decode. The PLA's also generate a sync detect signal if all the data inputs are high. This resets the divide by 10 counter to establish byte synchronization. Shifts are counted by the divide by 10 counter to generate the GCR-BRDY signal. This signal then latches the decoded output into the GCR read latch and generates a negitive edge on the SO (set overflow) pin of the 6508 to signal the processor a byte is available to be read.

2.1.4 MFM Encode/Decode

When the MFM bit in the control register is hi the disk data select uses the MFM encode / decode logic. The logic for MFM read/write may be added at a later date.

3. Command Protocol

Commands are passed to the subsystem by writing a SCSI like command to a page in the Commodore 900's memory (see system specification for exact location). These commands are eight words in length for every physical device. The highest priority device is at an offset of 0x0000 from the start of the physical segment, next at offset 0x0010, and last at 0x0020. The first eight bytes of the command area are used to pass command information to the controller and the last four bytes are used to receive the status information back from the controller. The twelfth byte should be set to 0xFF to flag that a valid command for this device is present in the command block. After the command is written to the command page an I/O select is generated by writing a 0x00 to a specific I/O location. This select line is connected to the 8716. When the 8716 sees a valid select it latches the IRQ line to the controller low and causes an interrupt to be sent to the controller. The controller then sets up the address of the command page with the 8716 LD2 and LD1 states and reads the command from the Commodore 900's system memory. The floppy command is checked and if the command pending flag is found the DMA register is set up and the device driver for the floppy is called. If the command checked does not have the command pending flag set then the floppy controller disconnects from the DMA bus. Upon completion of a command, status information is placed at the end of the eight word command field and a vectored interrupt to the Z8001 is generated by the 8716 DMA IC. For the floppy driver a subset of class code 0 is the only section of the SCSI that is implemented. A DMA address must be specified for all commands that require data transfer.

Floppy Command Block

CLASS 0 COMMANDS

BYTE	BIT 7	6	5	4	3	2	1	0
00	CLASS CODE			OPCODE				
01	LOGICAL UNIT NUMBER			MSB LOGICAL BLOCK ADDRESS				
02	LOGICAL BLOCK ADDRESS							
03	LSB LOGICAL BLOCK ADDRESS							
04	NUMBER OF BLOCKS							
05	RSVD	RSVD	INT REQ	SPARE	SPARE	SPARE	RSVD	RSVD
06	DMA PAGE MSB							
07	DMA PAGE LSB							
08 - 11	RESERVED FOR DUAL ADDRESS AND EXTENDED BLOCK ADDRESS							
12	AD VALID	ERROR CLASS			ERROR CODE			
13	RESERVED			MSB LOGICAL BLOCK ADDRESS				
14	LOGICAL BLOCK ADDRESS							
15	LSB LOGICAL BLOCK ADDRESS							

DEFINITION OF COMMANDS and PARAMETERS

CLASS CODE — Always = 0

OPCODE

- Format = 04 — Action: entire disk is formatted. Logical Unit # is only parameter.
- Read = 08 — Action: Read from specifed unit # and block address into the specified DMA page.
- Write = 0A — Action: Write from a specified DMA page to a specified block address on a specified unit.
- Change command block = 0F — Action: Change the physical address of the floppy command block to the location specified by DMA page.

LOGICAL UNIT NUMBER

Drive 0 = 0
Drive 1 = 1

LOGICAL BLOCK ADDRESS — 21 bit starting block address (not used for format)

NUMBER OF BLOCKS — Number of blocks to transfer starting from the logical block address (not used for format)

RSVD — Reserved

SPARE — Spare

INT REQ — Interrupt host on every sector transfered

OFF = 0
ON = 1

DMA PAGE — Physical starting address of data in host (used in format to point to drive parameters)

STATUS INFORMATION RETURNED AFTER EVERY COMMAND

AD VALID	**Logical address valid**
ERROR CLASS:	
ERROR CODE	**CLASS 07**
00	No sense
03	Media error (Non recoverable)
05	Illegal request
06	Media change
07	Write protect
09	Sector interrupt
0A	Power up failed (diagnostics)

3.1 GCR Format

The Commodore 900 system runs the Coherent operating system and requires a 512 byte sector size. This format uses the existing Commodore GCR data encode scheme with variable density (see table 3.2.2). This format allows for 1,224,704 bytes of storage per drive. The data rate for one track varies from 41K to 33K bytes/second for the different regions. The multi track transfers including step settle and latency is 38K to 31K bytes/second.

3.1.1 GCR 512 Format

This section describes the CBM GCR 512 disk format used by the Commodore 900.

3.1.1.1 ID Field

The ID field for each sector is described below.

	! SYNC	! SECTOR	! TRACK	! GAP1 !
BITS	40	10	10	120

SYNC	-	20 bits of 1's for synchronization
SECTOR	-	Sector (0-15) with bit 4 as parity (even)
TRACK	-	Track number 0-79 with bit 7 as parity (even)
GAP 1	-	Time required to turn on write gate

3.1.1.2 Data Field

The data field for a sector is below.

	! SYNC	! AM	! DATA	! CHECKSUM	! GAP2 !
BITS	40	10	5120	10	>120

SYNC	-	20 bits of 1's for synchronization
AM	-	Address mark = hex 88
DATA	-	512 bytes of data
CHECKSUM	-	Exclusive OR of data bytes and checksum = 0
GAP2	-	Variable determined by drive speed

3.1.1.3 GAP2 Determination

During format GAP2 allows for 2% speed variation at the end of each sector, so that fast running drives will not write over the begining of the next sector. This 2% is for drives that are running at the correct speed, so this figure is adjusted for the drive speed at the time of format. Compensating for motor speed error in format allows the GAP2 to be 2% vs. the 4% required if the format speed is not known. This generates another 2% of disk space for storage and checks the drive for an out of specification speed. The format performs a speed check to by writing 38*256 bytes of GCR 0 (210,773ms or 105%) followed by 4 bytes of sync. The GCR 0 bytes are then counted until sync is found. This number is the total number of bytes that can be written on track 1 at the current speed. A minimum GAP2 of 12 bytes is selected for a drive that is running 2% fast and then this is incremented by 1 for every 16 bytes/track extra. The same proceedure is followed for all four of the density regions.

3.1.2 GCR Data Rates/Formatted Capacity

This table lists the data rates and capacity for each GCR region on the disk. The 8723 IC is set up for the appropriate region by the GCR0 and GCR1 control bits.

GCR0	GCR1	TRACK REGION	BIT TIME	SECTORS/ TRACK	SECTORS/ REGION
0	0	1-39	2.16	16	624
1	0	40-53	2.33	15	210
0	1	54-64	2.50	14	154
1	1	65-80	2.66	13	208

		SECTORS/SIDE	=		1,196
		SECTORS/DISK	=		2,392
		BYTES/DISK	=		1,224,704
		BYTES/CONTROLER	=		2,449,408

3.1.3 GCR Code Conversion

The 4 to 5 bit GCR conversion used in the floppy subsystem is shown below.

#	BINARY	GCR	#	BINARY	GCR
0	0000	01010	8	1000	01001
1	0001	01011	9	1001	11001
2	0010	10010	A	1010	11010
3	0011	10011	B	1011	11011
4	0100	01110	C	1100	01101
5	0101	01111	D	1101	11101
6	0110	10110	E	1110	11110
7	0111	10111	F	1111	10101
	-	-		-	-