

Device Side Data
Model FC5025
USB 5.25" Floppy Controller

Command Set Specification
For Third Party Developers

\$Date: 2010-02-13 05:58:51 \$

Introduction

This document describes the command set used by a host to communicate with the FC5025. It is intended for those who are writing custom drivers. Most people don't need to read this document.

If you have experience with SCSI or with the USB Mass Storage Class, you may find the FC5025's command set similar. However, there are enough differences that code intended for SCSI or for USB MSC will not work with the FC5025.

To understand this document, you should already be familiar with USB, and you should know how to use libusb or your operating system's API to access USB devices. You should also know how a floppy drive works and have information on the track layout of any format you want to support.

Typical sequence of operations

A typical program that makes an image of a disk will do the following:

1. Recalibrate the drive to track 0 (SEEK command).
2. Set the drive to the correct density (FLAGS command).
3. Read a series of sectors (READ FLEXIBLE commands), periodically moving the head to a different track (SEEK commands).

Identifying connected FC5025 controllers

FC5025 controllers can be identified by their Vendor ID (VID) of 0x16c0 and Product ID (PID) of 0x06d6. Drivers should ignore the release number, interface class, etc, and consider any device with this VID/PID combination to be an FC5025.

The FC5025 is not programmed with a serial number. So, if there are multiple FC5025s connected to one host, you must differentiate between them by which USB port each is plugged into.

Endpoints

Endpoint 0 is a control endpoint, used mainly for USB enumeration. All disk operations are initiated by command blocks on Endpoint 1, a bulk endpoint. Sector data is also transferred on Endpoint 1. The maximum packet size on Endpoint 1 is 64 bytes.

Control requests

The FC5025 implements the standard control requests required by the USB specification.

In addition to resetting the toggle bits, a `CLEAR_FEATURE(ENDPOINT_HALT)` or `SET_FEATURE(ENDPOINT_HALT)` on Endpoint 1 (either direction), or a `SET_CONFIGURATION`, will terminate any disk operation in progress without sending a CSW. It will not reset the global flags or clear the current track number. (CSWs and global flags are discussed below.)

Suspend mode behavior

The FC5025 supports a suspend mode, in accordance with the USB specification. A suspend will cause the FC5025 to terminate any disk operation in progress without sending a CSW. It will not reset the global flags. However, it will clear the current track number.

Bulk command protocol

The host initiates an operation by sending a Command Block Wrapper (CBW) on Endpoint 1. If the CBW specifies a transfer length greater than zero, there is then a transfer phase where the FC5025 sends data to the host. After the transfer phase, the FC5025 sends the host a Command Status Wrapper (CSW) indicating the completion status of the command. If the transfer length is zero, the CSW follows the CBW with no transfer phase in between.

CBW format

A CBW is always 63 bytes long. It consists of a 15-byte header followed by a 48-byte Command Descriptor Block (CDB). The CDB's first byte is the opcode and the remaining 47 bytes are the parameters. If a command has less than 47 bytes of parameters, the host must fill the unused bytes with zeros.

Parameters in the CBW header are sent low byte first, but parameters in the CDB are sent high byte first.

The CBW format is as follows:

| <i>Offset</i> | <i>Field</i> | <i>Description</i> |
|---------------|-----------------|--|
| 0-3 | Signature | Sequence 0x43 0x46 0x42 0x43 identifies this as a CBW. |
| 4-7 | Tag | Arbitrary value, echoed by FC5025 in CSW. |
| 8-11 | Transfer length | Number of bytes to transfer, or 0 for no transfer phase. |
| 12 | Reserved | Always set to 0x80. |
| 13-14 | Padding | Always set to 0x00 0x00. |
| 15 | CDB Opcode | Identifies which operation to perform. |
| 16-62 | CDB Parameters | Operation dependent parameters. |

The host should use a different tag value in each CBW, and verify that the FC5025 returns the identical value in the corresponding CSW.

Transfer phase

If the host requested a transfer, the FC5025 will send at most the requested number of bytes during the transfer phase. It may send less than the requested number, but never more. For example, if an error occurs in the middle of an operation, the FC5025 may terminate the transfer phase early. Or, certain operations have fixed-size responses, and the FC5025 will send only enough bytes to contain the response.

If the FC5025 is transferring more than 64 bytes, the transfer phase will begin with a series of 64-byte packets. If the host requested an exact multiple of 64 bytes, and the FC5025 is sending the requested number of bytes, the final packet will be 64 bytes long. Whenever the FC5025 is sending less data than the host requested, the last packet will be less than 64 bytes, or it will be a zero-length packet (ZLP). If the FC5025 has no data to transfer, the transfer phase will consist of a single ZLP. If the host specified a transfer length of zero, the FC5025 will skip the transfer phase entirely without sending a ZLP.

This is the normal way of performing a USB transfer, so if you request an 'n' byte transfer and then try to read 'n' bytes from Endpoint 1, your USB stack should do the right thing.

CSW format

The FC5025 uses 12-byte CSWs. However, future products may use CSWs up to 31 bytes. A driver should accept CSWs up to 31 bytes long and ignore any contents past the 12th byte. Drivers should reject CSWs longer than 31 bytes.

The CSW format is as follows:

| <i>Offset</i> | <i>Field</i> | <i>Description</i> |
|---------------|--------------|--|
| 0-3 | Signature | Sequence 0x42 0x53 0x43 0x46 identifies this as a CBW. |
| 4-7 | Tag | Tag value copied from CBW. |
| 8 | Status | 0 for pass, 1 for fail. |
| 9 | Sense key | |
| 10 | ASC | Additional Sense Code |
| 11 | ASCQ | Additional Sense Code Qualifier |

If an operation succeeds, Status will be 0 and the sense key will be 00 (NO SENSE) or 01 (RECOVERED ERROR). If the sense key is 01, ASC and ASCQ will give more information on the recovered error. If an operation fails, Status will be 1 and sense/ASC/ASCQ will specify the error.

Sense codes

The following sense codes are common to all operations:

| <i>Status</i> | <i>Key</i> | <i>ASC</i> | <i>ASCQ</i> | <i>Description</i> |
|---------------|------------|------------|-------------|---|
| 0 | 00 | 00 | 00 | NO SENSE |
| | | | | Operation completed successfully. |
| 1 | 05 | 20 | 00 | INVALID COMMAND OPERATION CODE |
| | | | | Opcode byte in CDB is not valid. |
| 1 | 05 | 24 | 00 | INVALID FIELD IN CDB |
| | | | | A parameter in the CDB is set to an invalid value for this operation, or a value not compatible with the other parameter settings, or there are non-zero bytes in the padding area at the end of the CDB. |

Some operations may also return other sense codes. Those sense codes are documented in the description of each opcode.

Stalls

If the host sends a CBW that is the wrong number of bytes or contains an invalid signature, the FC5025 will stall Endpoint 1. The host must clear the stall with a control request to resume operations.

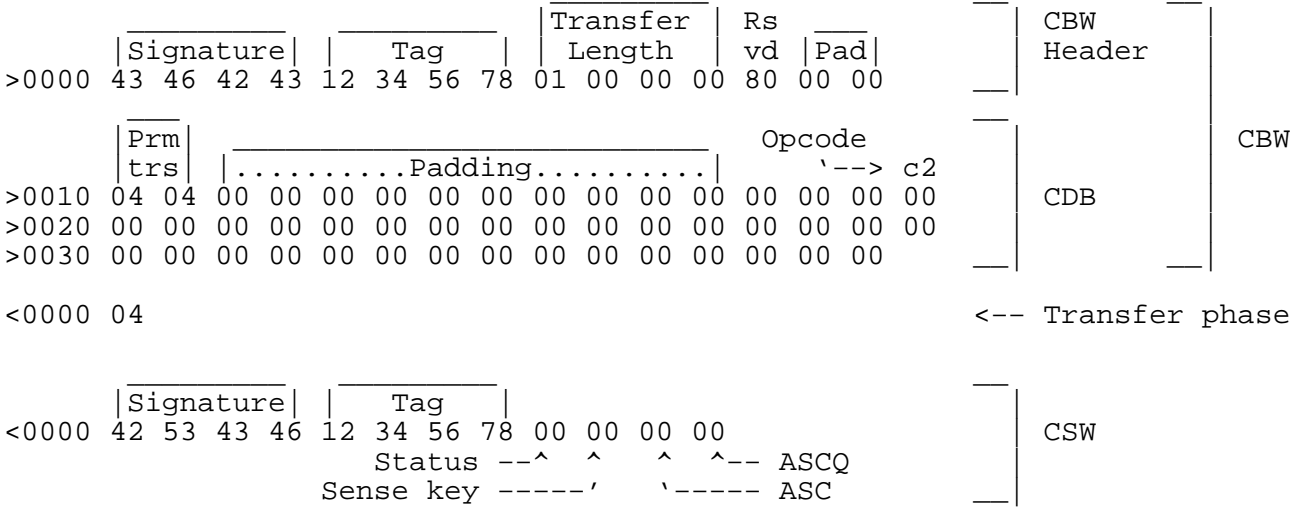
An invalid opcode will not cause a stall, nor will invalid parameters for an operation; the FC5025 will simply send a CSW indicating failure status, sending a zero-length transfer first if necessary.

Recommended timeouts

Normally, the FC5025 will accept a CBW packet immediately. However, in some circumstances, it may take slightly over 1 second for the FC5025 to accept a CBW. This delay can occur on the first CBW after resuming from a suspend or on the first CBW after a CLEAR_FEATURE. Accordingly, a timeout of 1.5 to 2 seconds is recommended when sending a CBW.

Recommended timeouts for the transfer phase and CSW vary depending on the operation. See the description of each opcode for more information.

Example command and response



SEEK operation (0xC0)

| Offset | Field | Description |
|--------|--------------|--|
| 0 | Opcode | Value 0xC0 indicates SEEK operation. |
| 1 | Mode | 0 Absolute 1 Relative (higher-numbered tracks) 2 Relative (lower-numbered tracks) 3 Recalibrate |
| 2 | Step rate | Step rate in 0.2ms increments. Always use 15 (3ms). |
| 3 | Track number | For absolute mode, track number to seek to. For relative modes, number of tracks to step. For recalibrate, maximum number of tracks to step. |

Sense codes

| Key | ASC | ASCQ | Description |
|-----|-----|------|--|
| 05 | 2C | 00 | COMMAND SEQUENCE ERROR An absolute seek was attempted from an unknown starting track. You must perform a successful recalibrate before attempting an absolute seek. |
| 02 | 06 | 00 | NO REFERENCE POSITION FOUND The FC5025 stepped the maximum number of tracks during a recalibrate, but the drive did not assert TRACK 00. |

Notes

The FC5025 keeps track of track numbers from 0 to 254. Relative seeks can be performed even if the FC5025 does not know the current track number. Seeking above track 254 or below track 0 will cause the current track number to become unknown, requiring a recalibrate before you can do an absolute seek. The current track number is also unknown on power-up or after a suspend.

The FC5025 will allow a seek to tracks below track 0 using a relative seek, but the disk drive may silently refuse to actually perform such a seek.

A relative seek of 0 tracks will set the DIRECTION output without generating any step pulses.

Recommended timeout

SEEK will always delay for the step delay before sending the first step pulse and after sending the last pulse. There is an additional delay of 90ms if the drive is not already selected and another 41ms if this seek is in a different direction from the previous one. You may calculate a timeout based on the number of steps, step rate, these additional delays and a reasonable margin. Or you may use a fixed timeout -- a timeout of 600ms to 1 second will allow plenty of time for a maximal-distance seek.

SELF TEST operation (0xC1)

| <i>Offset</i> | <i>Field</i> | <i>Description</i> |
|---------------|--------------|---|
| 0 | Opcode | Value 0xC1 indicates SELF TEST operation. |
| 1-2 | Test number | 0 Board test 1 Interface test |

Transfer phase

The FC5025 will send up to 32 bytes (256 bits) if requested. If the self-tests pass, all bits will be 0. Any errors will be indicated by a 1 bit.

Sense codes

| <i>Key</i> | <i>ASC</i> | <i>ASCQ</i> | <i>Description</i> |
|------------|------------|-------------|---|
| 04 | 40 | 80 | DIAGNOSTIC FAILURE The self-test failed. The exact error(s) are denoted by set bits in the transferred data. |

Notes

The interface test requires a drive to be attached with a disk (any type) inside. The board test will work with or without a drive attached.

The FC5025 may behave incorrectly if a suspend or a control request occurs during a SELF TEST operation.

Error codes

Error codes have the format Exx.yy.z, where xx is the test number, yy is the hex byte offset in the transfer phase, and z is the bit number within the byte (0 being the LSB and 7 the MSB). For example, if the first byte of transferred data for a board test is 0x02, this indicates error E00.00.1.

E00.00.0 DISK CHG stuck high
E00.00.1 READ DATA stuck high
E00.00.2 WRITE PROTECT stuck high
E00.00.3 TRK 00 stuck high
E00.00.4 INDEX stuck high
E00.00.5 DISK CHG stuck low
E00.00.6 READ DATA stuck low
E00.00.7 WRITE PROTECT stuck low
E00.01.0 TRK 00 stuck low
E00.01.1 INDEX stuck low
E01.00.0 Recalibrate failed
E01.00.1 Can't leave track 00
E01.00.2 Can't return to track 00
E01.00.3 Can't deselect drive
E01.00.4 No index pulses
E01.00.5 No data pulses
E01.00.6 Index pulses never stop
E01.00.7 Data pulses never stop

Recommended timeout

A board test should complete within 300ms, an interface test should complete within 4 seconds.

FLAGS operation (0xC2)

| <i>Offset</i> | <i>Field</i> | <i>Description</i> |
|---------------|--------------|---------------------------------------|
| 0 | Opcode | Value 0xC2 indicates FLAGS operation. |
| 1 | Mask | Bitmask of flags to change. |
| 2 | Values | Values for changed flags. |

Transfer phase

If the host requests a transfer, the FC5025 will send a byte containing the new flags value.

Notes

This command modifies and/or queries the global flags. The host may read the current flags without changing them by using a bitmask of 0.

The global flags are reset to 0 on power-on or after a bus reset.

Recommended timeout

A FLAGS command will sometimes delay for about 1 second before completing the transfer phase. This occurs if you change the Density Select flag while the motor is running, or if you turn on the Force Motor On flag when the motor is not yet running. If the host has not requested a transfer, there will be a 1 second delay before completing the CSW phase in these cases. A timeout value of 1.5 to 2 seconds is recommended.

Flags

0 corresponds to the least significant bit, 7 is the most significant bit.

- | | |
|-----|--|
| 0 | Force Motor On Normally, the FC5025 will turn the motor on before a read, and turn it off after there have been no operations for about 3.5 seconds. (Once the motor is on, any command, even one that does not access the disk, will keep it on.) If the Force Motor On flag is set, the FC5025 will instead leave the motor turned on at all times. The FC5025 turns on the motor when you set the flag. |
| 1 | Force Motor Off This flag will prevent the FC5025 from asserting the MOTOR ON signal. It will still assert the drive select signal during a seek or read. If both the Force Motor On and Force Motor Off flags are set, the FC5025 will always leave the drive select signal asserted but will never assert the MOTOR ON signal. |
| 2 | Density Select If this flag is set, the FC5025 will assert the MODESEL signal, informing the drive that the disk is recorded at low density (eg 250/300kbps). If the flag is clear, the FC5025 will deassert the signal, signifying high density (eg 500kbps). |
| 3-7 | Reserved |

DRIVE STATUS operation (0xC3)

| <i>Offset</i> | <i>Field</i> | <i>Description</i> |
|---------------|--------------|--|
| 0 | Opcode | Value 0xC3 indicates DRIVE STATUS operation. |

Transfer phase

| <i>Offset</i> | <i>Field</i> | <i>Description</i> |
|---------------|--------------|---|
| 0 | Track number | Current head position, or 0xFF if unknown. |
| 1-2 | Disk speed | Disk speed in units of 0.01 RPM. |
| 3 | Sector count | 0 No index pulses 1 Soft-sectored 2-255 Hard-sectored |
| 4 | Flags | Bit 0 = pin 34 asserted Bit 1 = disk write-protected |

Notes

A DRIVE STATUS command will not turn on the motor; to measure the disk speed, the disk must already be spinning from a recent read operation or due to the Force Motor On flag.

Also, the write-protect status is not valid unless the drive is already selected, from a recent read or seek or Force Motor On. If the drive is not currently selected, a DRIVE STATUS operation will indicate that the disk is write-allowed regardless of its actual status.

Drives usually use pin 34 to indicate a disk change, but can be jumpered to indicate readiness status through pin 34 instead. The FC5025 does not act on the status of pin 34, it merely reports it in DRIVE STATUS responses.

The sector count field represents the number of physical sector holes, not the number of recorded sectors.

Some drives suppress index pulses during a seek. On these drives, DRIVE STATUS operations after a seek will return an incorrect disk speed value until the disk has completed several revolutions since the seek. On some drives this behavior can be corrected, for example on the TEAC FD55 by closing jumper E2.

Recommended timeout

A DRIVE STATUS command will complete the transfer and CSW phases within a few milliseconds. Drive speed and sector count are continuously measured and the the DRIVE STATUS operation returns the measurements that have already been made. A timeout of 100ms on the host is reasonable.

INDEXES operation (0xC4)

| <i>Offset</i> | <i>Field</i> | <i>Description</i> |
|---------------|--------------|---|
| 0 | Opcode | Value 0xC4 indicates INDEXES operation. |

Transfer phase

The FC5025 sends 3 bytes for each index pulse received. The number of pulses to wait for is automatically determined based on the requested transfer length. Each 3-byte value is the time delta from the previous pulse, in microseconds. A value of 0 signifies that 300ms passed with no index pulse.

Notes

The maximum transfer size for INDEXES is 63 bytes.

An INDEXES command will not turn on the motor; to measure index pulse intervals, the disk must already be spinning from a recent read operation or due to the Force Motor On flag.

Recommended timeout

The recommended timeout is 400ms times the number of values requested, plus 600ms.

READ FLEXIBLE operation (0xC6) - Apple GCR

| Offset | Field | Description |
|--------|-----------------|--|
| 0 | Opcode | Value 0xC6 indicates READ FLEXIBLE operation. |
| 1 | Flags | Bit 0 = Side Select - always use 0 Bit 1 = Send address field prior to data field Bit 2 = Automatic Overrun Recovery Bit 3 = Disable Auto Sync Bit 4 = Report Angular Position Bits 5-7 = Reserved - always use 0 |
| 2 | Format | Value 1 indicates Apple GCR. |
| 3-4 | Bit cell time | Bit cell time in nanoseconds. At 360 RPM, use 3266ns. |
| 5 | Sector hole | Set to 0 for soft sectoring. |
| 6-8 | Read delay | Set to 0 for soft sectoring. |
| 9 | Random read | 0xFF Read a specific sector 0x00 Begin reading at an arbitrary position |
| 10-21 | Address pattern | Address field to look for including prologue, GCR coded. |
| 22-33 | Address mask | Bitmask of significant bits in address pattern. |
| 34-36 | Data prologue | Prologue to look for before data field. Use all 0s to read immediately following address field. |
| 37 | Sub-operation | Set to 0 for read. |
| 38 | Sub-parameter | Set to 0 for read. |

Transfer phase

The FC5025 sends the sector contents as a series of GCR nibbles, as they are on the disk. The host must decode the 5-and-3 or 6-and-2 GCR coding (for 13-sector or 16-sector formats, respectively) and verify the checksum.

If the Send Address Field flag is set, the FC5025 will first send the address field, then the data field, both in the same transfer phase. The address field is sent as 9 GCR-coded nibbles and does not include the prologue. The FC5025 will not read from the gap area between the address and data fields, unless the data prologue parameter is set to all zeros. If the Send Address Field flag is clear, the FC5025 will send only the data field.

If auto sync is enabled (the default), the FC5025 will skip any leading 0 bits between GCR nibbles, so that all transferred GCR nibbles start with a 1 bit. If the Disable Auto Sync flag is set, the FC5025 will include leading 0 bits as part of the GCR nibbles during the data field. Auto sync is always enabled when reading the address field.

Notes

The FC5025 does not perform its own encoding/decoding/checksumming on the address field when looking for a sector; the host must construct the address field for the desired sector, calculate the checksum, 4-and-4 encode the address field and send the pre-encoded address field as the address pattern parameter.

See "READ FLEXIBLE operation (0xC6) - General information" for information on sense codes, the Automatic Overrun Recovery flag and the Report Angular Position flag.

READ FLEXIBLE operation (0xC6) - Apple GCR - Example

| | | | | | | | | | | | | | | | | | | |
|---|----|--------|----|-----------|----|-------------------------|----|--------------------|----|-----|----|-----|----------|----|--------|----|--|--|
| | | | | | | | | | | | | | | | Opcode | | | |
| CBW Header, transfer length = 345 bytes | | | | | | | | | | | | | | | V | | | |
| >0000 | 43 | 46 | 42 | 43 | 12 | 34 | 56 | 78 | 59 | 01 | 00 | 00 | 80 | 00 | 00 | c6 | | |
| AppleGCR | | | | | | | | | | | | | | | | | | |
| OverRecv | | 3266 | | Soft-Sect | | | | Addrss | | Vol | | Trk | | | | | | |
| V V | | ns | | | | | | Prolog | | 254 | | 00 | | | | | | |
| >0010 | 04 | 01 | 0c | c2 | 00 | 00 | 00 | 00 | ff | d5 | aa | 96 | ff | fe | aa | aa | | |
| | | | | | | | | | | | | | | | | | | |
| Sec | | Csum | | Epi | | Address Mask | | | | | | | | | | | | |
| 00 | | ff fe | | de ff | | ff ff ff ff ff ff ff ff | | | | | | | | | | | | |
| >0020 | aa | aa | ff | fe | de | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | | |
| | | | | | | | | | | | | | | | | | | |
| AddrMask | | Data | | Sub | | Padding | | | | | | | | | | | | |
| V | | Prolog | | Op | | | | | | | | | | | | | | |
| >0030 | ff | d5 | aa | ad | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | | |
| | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| <0000 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | | |
| <0010 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | | |
| <0020 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | | |
| <0030 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | | |
| <0040 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | | |
| <0050 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | | |
| <0060 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | | |
| <0070 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | | |
| <0080 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | | |
| <0090 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | | |
| <00a0 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | | |
| <00b0 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | | |
| <00c0 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | | |
| <00d0 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | | |
| <00e0 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | | |
| <00f0 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | | |
| <0100 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | | |
| <0110 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | | |
| <0120 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | | |
| <0130 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | | |
| <0140 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | | |
| <0150 | ff | ff | ff | ff | ff | ff | 96 | de | aa | | | | | | | | | |
| | | | | | | | | Checksum-^ Epilg | | | | | | | | | | |
| | | | | | | | | ~~~~~ | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | |
| <0000 | 42 | 53 | 43 | 46 | 12 | 34 | 56 | 78 | 00 | 00 | 00 | 00 | | | | | | |
| | | | | | | | | | | | | | <--- CSW | | | | | |

CBW
Including
CDB

Transfer
Phase

READ FLEXIBLE operation (0xC6) - Commodore GCR

| Offset | Field | Description |
|--------|-----------------|--|
| 0 | Opcode | Value 0xC6 indicates READ FLEXIBLE operation. |
| 1 | Flags | Bit 0 = Side Select - always use 0 Bit 1 = Send address block prior to data block Bit 2 = Automatic Overrun Recovery Bit 3 = Reserved - always use 0 Bit 4 = Report Angular Position Bits 5-7 = Reserved - always use 0 |
| 2 | Format | Value 2 indicates Commodore GCR. |
| 3-4 | Bit cell time | Bit cell time in nanoseconds. |
| 5 | Sector hole | Set to 0 for soft sectoring. |
| 6-8 | Read delay | Set to 0 for soft sectoring. |
| 9 | Random read | 0xFF Read a specific sector 0x01 Begin reading after any sync mark 0x00 Begin reading at an arbitrary position |
| 10-19 | Address pattern | Address field to look for including block ID, GCR coded. |
| 20-21 | Reserved | Set to 0. |
| 22-31 | Address mask | Bitmask of significant bits in address pattern. |
| 32-33 | Reserved | Set to 0. |
| 34 | Await data blk | 0xFF After finding sector, wait for sync mark. 0x00 Begin reading immediately after address block. |
| 35-36 | Reserved | Set to 0. |
| 37 | Sub-operation | Set to 0 for read. |
| 38 | Sub-parameter | Set to 0 for read. |

Transfer phase

The FC5025 sends the sector contents as a series of GCR nibbles, as they are on the disk. The host must decode the 5/4 GCR code and verify the checksum.

The sector contents are sent starting with the block ID of the data block, which the FC5025 does not verify (it simply waits for the first sync mark after the desired address block). The host should verify that the block ID is correct.

If the Send Address Block flag is set, the FC5025 will first send the address block, then the data block, both in the same transfer phase. The address block is sent as 10 GCR-coded nibbles and includes the block ID. The FC5025 will not read from the gap area between the address and data blocks, unless the Await Data Block parameter is set to zero. If the Send Address Field flag is clear, the FC5025 will send only the data block.

Bit Rates

The bit cell times for Commodore disks are listed below. Normally the FC5025 is used with drives jumpered to run only at 360 RPM, so use the 360 RPM values.

| Tracks | 300RPM | 360RPM |
|--------|--------|--------|
| 1-17 | 3250ns | 2708ns |
| 18-24 | 3500ns | 2917ns |
| 25-30 | 3750ns | 3125ns |
| 31-35 | 4000ns | 3333ns |

Notes

The FC5025 does not perform its own encoding/decoding/checksumming on the address block when looking for a sector; the host must construct the address block for the desired sector, calculate the checksum, 5/4 GCR encode the address block and send the pre-encoded address block as the address pattern parameter.

See "READ FLEXIBLE operation (0xC6) - General information" for information on sense codes, the Automatic Overrun Recovery flag and the Report Angular Position flag.

READ FLEXIBLE operation (0xC6) - Commodore GCR - Example

| | | | | | | | | | | | | | | | | |
|---|----|----|----|---------|----|----|----|-----------|----|----|----|--------------|----|---------|--------|----|
| | | | | | | | | | | | | | | | Opcode | |
| CBW Header, transfer length = 325 bytes | | | | | | | | | | | | | | | V | |
| >0000 | 43 | 46 | 42 | 43 | 12 | 34 | 56 | 78 | 45 | 01 | 00 | 00 | 80 | 00 | 00 | c6 |
| CommodoreGCR | | | | | | | | | | | | | | | | |
| OverRecv | | | | 2708 | | | | Soft-Sect | | | | ID Csum | | Sec Trk | | ID |
| V | | V | | ns | | | | | | | | 08 10 | | 00 01 | | 1 |
| >0010 | 04 | 02 | 0a | 94 | 00 | 00 | 00 | 00 | ff | 52 | 56 | a5 | 29 | 4b | 92 | a6 |
| | | | | | | | | | | | | | | | | |
| ID | | | | Address | | | | | | | | | | | | |
| V | | | | Padding | | | | | | | | Address Mask | | | | |
| >0020 | b5 | 00 | 00 | 00 | 00 | 00 | ff | ff | ff | ff | ff | ff | ff | ff | 00 | 00 |
| | | | | | | | | | | | | | | | | |
| AddrMask | | | | Sub | | | | | | | | | | | | |
| V | | | | Rsv | | | | Op | | | | Padding | | | | |
| >0030 | 00 | ff | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | |
| <0000 | 55 | d4 | a5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 |
| <0010 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 |
| <0020 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 |
| <0030 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d |
| <0040 | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b |
| <0050 | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 |
| <0060 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 |
| <0070 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 |
| <0080 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d |
| <0090 | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b |
| <00a0 | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 |
| <00b0 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 |
| <00c0 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 |
| <00d0 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d |
| <00e0 | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b |
| <00f0 | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 |
| <0100 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 |
| <0110 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 |
| <0120 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d |
| <0130 | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b | 52 | d4 | b5 | 2d | 4b |
| <0140 | 52 | d4 | b5 | 29 | 4a | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | |
| <0000 | 42 | 53 | 43 | 46 | 12 | 34 | 56 | 78 | 00 | 00 | 00 | 00 | | | | |

CBW Including CDB

Transfer Phase

<--- CSW

READ FLEXIBLE operation (0xC6) - FM

| Offset | Field | Description |
|--------|---------------|--|
| 0 | Opcode | Value 0xC6 indicates READ FLEXIBLE operation. |
| 1 | Flags | Bit 0 = Side Select Bit 1 = Send ID field prior to data field Bit 2 = Automatic Overrun Recovery Bit 3 = Reserved - always use 0 Bit 4 = Report Angular Position Bits 5-7 = Reserved - always use 0 |
| 2 | Format | Value 3 indicates FM ("single density"). |
| 3-4 | Bit cell time | Bit cell time in nanoseconds. |
| 5 | Sector hole | Set to 0 for soft sectoring. |
| 6-8 | Read delay | Set to 0 for soft sectoring. |
| 9 | IDAM | 0x02-0xFF Low nibble of IDAM, including clock bits. 0x01 Begin reading after any preamble 0x00 Begin reading at an arbitrary position |
| 10-21 | ID pattern | ID field to look for, including clock bits. |
| 22-33 | ID mask | Bitmask of significant bits in ID pattern. |
| 34 | DAM | Low nibble of DAM, including clock bits. To read immediately following address field, set to 0. |
| 35-36 | Reserved | Set to 0. |
| 37 | Sub-operation | Set to 0 for read. |
| 38 | Sub-parameter | Set to 0 for read. |

Transfer phase

The FC5025 sends both the clock and data bits. For example, if the sector starts with the byte 0x42 (**0100 0010** binary), the transfer phase will begin with the two bytes 0xBA 0xAE (**10111010 10101110**), each byte in the transfer representing 4 bits of encoded sector data. The host must remove the clock bits and verify the checksum.

If the Send ID Field flag is set, the FC5025 will first send the ID field, then the data field, both in the same transfer phase. The ID field is sent as 12 bytes, including clock bits, without the IDAM. The FC5025 will not read from the gap area between the ID and data fields, unless the DAM parameter is set to zero. If the Send ID Field flag is clear, the FC5025 will send only the data field.

Notes

The FC5025 does not perform its own checksumming on the ID field when looking for a sector; the host must calculate the ID field checksum and send the entire ID field, including clock bits, as the ID pattern parameter.

See "READ FLEXIBLE operation (0xC6) - General information" for information on sense codes, the Automatic Overrun Recovery flag and the Report Angular Position flag.

READ FLEXIBLE operation (0xC6) - FM - Example

| CBW Header, transfer length = 516 bytes | | | | | | | | | | | | | | | Opcode |
|---|------|----------|-----|----|-----------|---------|----|----|-----|----|-----|----|-----|----|--------|
| | | | | | | | | | | | | | | | V |
| >0000 | 43 | 46 | 42 | 43 | 12 | 34 | 56 | 78 | 04 | 02 | 00 | 00 | 80 | 00 | c6 |
| FM IDAM FE/C7 | | | | | | | | | | | | | | | |
| OverRecv | 6667 | | ns | | Soft-Sect | | V | | Trk | | Sid | | Sec | | Len |
| | V | V | | | | | | | | | | | | | |
| >0010 | 04 | 03 | 1a | 0b | 00 | 00 | 00 | 00 | 7e | aa | aa | aa | aa | aa | aa |
| Len | | | | | | | | | | | | | | | |
| | V | CRC F1D3 | | | | ID Mask | | | | | | | | | |
| >0020 | ab | ff | ab | fb | af | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff |
| DAM FB/C7 | | | | | | | | | | | | | | | |
| ID Mask | | | Sub | | Padding | | | | | | | | | | |
| | V | V | Rsv | | Op | | | | | | | | | | |
| >0030 | ff | 6f | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| | | | | | | | | | | | | | | | |
| <0000 | ba | ae | bb | ae | ba | eb | ba | ba | ba | bf | ba | bb | ba | eb | ba |
| <0010 | ba | eb | ae | aa | aa | ab | be | ea | aa | eb | ba | ba | bb | af | ba |
| <0020 | ae | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa |
| <0030 | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa |
| <0040 | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa |
| <0050 | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa |
| <0060 | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa |
| <0070 | bf | ff | aa | aa | aa | aa | aa | aa | ff | fa | ff | ff | ff | ff | ff |
| <0080 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff |
| <0090 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ab | ff | aa |
| <00a0 | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa |
| <00b0 | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa | aa |
| <00c0 | aa | aa | aa | aa | aa | aa | aa | aa | aa | ff | ff | ff | ff | ff | ff |
| <00d0 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff |
| <00e0 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff |
| <00f0 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff |
| <0100 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff |
| <0110 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff |
| <0120 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff |
| <0130 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff |
| <0140 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff |
| <0150 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff |
| <0160 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff |
| <0170 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff |
| <0180 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff |
| <0190 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff |
| <01a0 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff |
| <01b0 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff |
| <01c0 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff |
| <01d0 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff |
| <01e0 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff |
| <01f0 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff |
| <0200 | ff | ea | be | be | | | | | | | | | | | |
| CRC | | | | | | | | | | | | | | | |
| --- | | | | | | | | | | | | | | | |
| <0000 | 42 | 53 | 43 | 46 | 12 | 34 | 56 | 78 | 00 | 00 | 00 | 00 | | | |
| | | | | | | | | | | | | | | | |

CBW
Including
CDB

Transfer
Phase

<--- CSW

READ FLEXIBLE operation (0xC6) - MFM

| Offset | Field | Description |
|--------|---------------|--|
| 0 | Opcode | Value 0xC6 indicates READ FLEXIBLE operation. |
| 1 | Flags | Bit 0 = Side Select Bit 1 = Send ID field prior to data field Bit 2 = Automatic Overrun Recovery Bit 3 = Reserved - always use 0 Bit 4 = Report Angular Position Bit 5 = Adaptive PLL - 0=enabled, 1=disabled Bits 6-7 = Reserved - always use 0 |
| 2 | Format | Value 4 indicates MFM ("double density"). |
| 3-4 | Bit cell time | Bit cell time in nanoseconds. |
| 5 | Sector hole | Set to 0 for soft sectoring. |
| 6-8 | Read delay | Set to 0 for soft sectoring. |
| 9 | IDAM | 0x02-0xFF IDAM, MFM encoded and compressed. 0x01 Begin reading after any preamble 0x00 Begin reading at an arbitrary position |
| 10-21 | ID pattern | ID field to look for, MFM encoded and compressed. |
| 22-33 | ID mask | Bitmask of significant bits in ID pattern. |
| 34 | DAM | DAM, MFM encoded and compressed. To read immediately following address field, set to 0. |
| 35-36 | Reserved | Set to 0. |
| 37 | Sub-operation | Set to 0 for read. |
| 38 | Sub-parameter | Set to 0 for read. |

Transfer phase

The FC5025 sends both clock and data bits. However, to reduce the transfer rate, it compresses the data, as follows: Due to the MFM encoding rules, a "1" data bit will never be immediately adjacent to a "1" clock bit. Therefore, when including both clock and data bits, any pair of "1" bits will have at least one "0" bit between them. The FC5025 takes advantage of this by omitting the first "0" bit following any "1" bit. The host must take this into account to decode the original sector data. This scheme results in variable compressed data length, ranging from a minimum of 8 output bits per byte of sector data to a maximum of 12 output bits. The host must decode this data stream and verify the checksum.

As an example, suppose a sector contains the byte 0x4E (0100 1110 binary), and the previous byte is 0x00. According to the MFM encoding rules, the clock byte will be 0x90 (10010000), and after interleaving the clock and data bits, the raw data stream on disk will be 1001001001010100. After removing the first 0 bit after each 1, the FC5025 will transmit 1010101110 to the host -- the first byte after compression is 0xAB (10101011), and the high 2 bits of the next byte will be 10.

If the Send ID Field flag is set, the FC5025 will first send the ID field, then the data field, both in the same transfer phase. The ID field is sent as 13 bytes, including clock bits, compressed, and including the IDAM. The FC5025 will not read from the gap area between the ID and data fields, unless the DAM parameter is set to zero. If the Send ID Field flag is clear, the FC5025 will send only the data field.

The usual DAM for MFM formats encodes to 9 bits according to the above compression scheme, but the FC5025 only compares the first 8 bits with the DAM parameter. The first bit sent to the host will be the ninth bit of the DAM. The host should verify that this bit corresponds to a correct DAM.

The FC5025 reads until it has enough output bytes to satisfy the requested transfer length. As described above, the number of output bits may vary from 8 to 12 per byte of sector data. The host assume that each byte of data will encode to 12 output bits, and set the transfer length accordingly. If the sector ends up encoding more efficiently, the FC5025 will continue to read past the end of the sector; the host should simply discard the additional data.

| CBW Header, transfer length = 772 bytes | | | | | | | | | | | | | | | Opcode | | | |
|---|----|----|------|---------|-----------|----|----|---------|----|-----|------|-----|-----|------|---------|----|-------------------------|--|
| >0000 | 43 | 46 | 42 | 43 | 12 | 34 | 56 | 78 | 04 | 03 | 00 | 00 | 80 | 00 | 00 | c6 | CBW Including CDB | |
| MFM | | | | IDAM | | | | FE | | | | | | | | | | |
| OverRecv | V | V | 2000 | | Soft-Sect | | | | V | Trk | Side | Sec | Len | CRC | | | | |
| | V | V | ns | | | | | | V | 00 | 00 | 01 | 02 | CA6F | | | | |
| >0010 | 04 | 04 | 07 | d0 | 00 | 00 | 00 | 00 | fe | ff | ff | fe | be | 9a | 95 | 9e | Transfer Phase | |
| ID Pattern | | | | Padding | | | | ID Mask | | | | | | | | | | |
| DAM | | FB | | | | | | | | | | | | | | | | |
| ID Mask | V | V | Rsv | | Sub | | Op | | | | | | | | Padding | | | |
| >0030 | 00 | f9 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | | |
| <0000 | f2 | 6b | d5 | 5e | aa | 97 | 55 | 65 | 5f | af | d9 | 2a | 76 | df | ff | f5 | Transfer Phase | |
| <0010 | fd | 7e | bf | fe | 9d | ff | f6 | ff | 57 | ea | f7 | 7f | f7 | bf | fe | bf | | |
| <0020 | ff | ff | ff | ff | ff | ff | ff | ff | ff | ff | f4 | a9 | 6b | db | d3 | 75 | | |
| <0030 | aa | ba | af | 57 | d5 | d5 | e9 | 59 | 2d | 2a | fd | 7e | bf | 5f | 5b | 57 | | |
| <0040 | a4 | 97 | 6d | 5a | b5 | fa | fd | 7b | e5 | b5 | ef | ad | cc | ba | 7b | ff | | |
| <0050 | 7d | e9 | be | f3 | 99 | ef | ff | 66 | 6d | 2b | 3b | 7a | 93 | 74 | d4 | ae | | |
| <0060 | 7e | be | be | a7 | 57 | 4d | fd | fa | f5 | ca | bf | 6e | fe | da | b4 | ff | | |
| <0070 | 7b | ed | 37 | 77 | 6e | fa | b5 | d5 | 9f | ad | aa | 5b | be | 9b | 77 | ba | | |
| <0080 | fa | fa | f9 | eb | 2d | 59 | d5 | 7a | ad | 7b | fb | ab | db | ab | 3e | dc | | |
| <0090 | be | b5 | a6 | ee | ea | cf | ab | 55 | dd | af | be | a5 | fd | 7b | eb | ce | | |
| <00a0 | ab | 74 | d7 | df | db | db | bb | 7d | ea | da | 6e | f5 | f7 | f6 | f6 | f7 | | |
| <00b0 | 5f | 57 | bc | ab | fd | 2d | 92 | f7 | d5 | aa | d3 | 52 | ab | ea | 5b | 2a | | |
| <00c0 | a7 | d5 | aa | d3 | 55 | 33 | ea | 77 | af | ff | 79 | d5 | 6e | b4 | fa | b4 | | |
| <00d0 | db | de | 99 | f7 | f7 | bb | 2a | de | 7f | 5b | f5 | ed | 55 | 4f | ab | db | | |
| <00e0 | 4d | 54 | cf | bf | e9 | cd | ff | e9 | b4 | da | 6a | 55 | 7d | 4b | a4 | bd | | |
| <00f0 | f5 | cb | 55 | 5f | f7 | 57 | 73 | 37 | fe | f2 | d2 | 6f | fd | d5 | d3 | 2d | | |
| <0100 | 3f | 9c | ea | e9 | bf | a7 | ce | b5 | f3 | eb | 95 | ae | 4b | a4 | b6 | 4f | | |
| <0110 | ea | f4 | ea | e9 | bf | bd | 72 | b5 | cb | db | 4f | 95 | eb | e7 | 97 | 4f | | |
| <0120 | bf | ed | 7b | ed | 65 | a6 | a7 | bb | f6 | f7 | aa | de | 5a | fe | 9a | cb | | |
| <0130 | 6a | 4d | d3 | 74 | db | 93 | e5 | ad | 32 | dd | b2 | aa | ea | b5 | a5 | de | | |
| <0140 | f6 | 4f | bb | 53 | ff | cf | db | 7d | bd | aa | a9 | f7 | ab | 69 | aa | 99 | | |
| <0150 | f5 | 39 | bf | fd | e7 | 57 | db | fd | 2f | 4a | a9 | 6f | 2f | 72 | ff | ba | | |
| <0160 | 99 | b4 | df | fb | cb | a4 | bf | f4 | d4 | 99 | 53 | 77 | 52 | 73 | 7a | 5f | | |
| <0170 | af | eb | de | 55 | ff | fb | 6f | 7a | 67 | d7 | 69 | da | 5a | 5a | 77 | 49 | | |
| ... | | | | | | | | | | | | | | | | | | |
| <02c0 | 55 | d5 | 75 | 5d | 57 | 55 | d5 | 75 | 5d | 57 | 55 | d5 | 75 | 5d | 57 | 55 | | |
| <02d0 | d5 | 75 | 5d | 57 | 55 | d5 | 75 | 5d | 57 | 7f | ff | ff | ff | ff | ff | ff | | |
| <02e0 | ff | ff | ff | ff | ff | a5 | 2c | a5 | 94 | bf | df | ff | ff | af | d2 | be | | |
| <02f0 | bd | ab | aa | ea | ba | ae | ab | aa | ea | ba | ae | ab | aa | ea | ba | ae | | |
| <0300 | ab | aa | ea | ba | | | | | | | | | | | | | | |
| <0000 | 42 | 53 | 43 | 46 | 12 | 34 | 56 | 78 | 00 | 00 | 00 | 00 | | | | | <--- CSW | |

READ FLEXIBLE operation (0xC6) - Hard sectored / index hole relative

| Offset | Field | Description |
|--------|---------------|---|
| 0 | Opcode | Value 0xC6 indicates READ FLEXIBLE operation. |
| 1 | Flags | Bit 0 = Side Select Bit 1 = Send ID field prior to data field Bit 2 = Reserved - always use 0 Bit 3 = Disable Auto Sync (Apple GCR only) Bit 4 = Report Angular Position Bit 5 = Adaptive PLL - 0=enabled, 1=disabled (MFM only) Bits 6-7 = Reserved - always use 0 |
| 2 | Format | 1 Apple GCR 2 Commodore GCR 3 FM 4 MFM |
| 3-4 | Bit cell time | Bit cell time in nanoseconds. |
| 5 | Sector hole | 1 = First sector hole after index hole 2-99 = N'th sector hole after index hole 254 = Index hole (hard sectored disks) 255 = Index hole (soft sectored disks) |
| 6-8 | Read delay | Microseconds to delay after sector hole leading edge before reading. |
| 9-38 | ... | Format-dependent parameters |

Notes

For MFM, after taking into account the cumulative index position error between the writing and reading drives, the read delay must place the beginning read position no earlier than 4 byte times before the beginning of the preamble and no later than 8 byte times before the end of the preamble.

To calculate the correct read delay value, use the formula $WD + PRE/2 - 6*BT$, where WD is the writing drive's write delay from the sector hole leading edge, PRE is the preamble duration, and BT is the byte time (8 times the bit cell time), all in microseconds. This information should be available in the documentation of the system that wrote the disk. If the disk was written at 300 RPM and will be read at 360 RPM, compensate for the speed difference by multiplying all values by 5/6ths before plugging them into the formula.

For FM, the beginning read position must be no earlier than 2 byte times before the beginning of the preamble, and no later than 4 byte times before the end of the preamble. Use the formula $WD + PRE/2 - 3*BT$ to calculate the correct read delay value, compensating for the speed difference if necessary.

If the host initiates a hard-sectored READ FLEXIBLE operation immediately after a SEEK completes, the FC5025 will wait for several successive sector pulses followed by an index pulse before attempting to locate the requested sector hole. This ensures that the FC5025 will always read the correct sector even if the drive has suppressed index pulses during the seek operation.

Because the FC5025 ignores index pulses during soft-sectored reads, a drive with a misaligned index sensor may work perfectly for soft-sectored reads but fail completely for hard-sectored reads.

Automatic Overrun Recovery cannot be enabled for hard-sectored reads.

READ FLEXIBLE operation (0xC6) - General information

If the Report Angular Position flag is set, the FC5025 will note the angular position of the disk after reading the final disk byte and transmit the position as the last 3 bytes of the transfer. The angular position is measured in microseconds, relative to the most recent index pulse. If there has been no index pulse for comparison, the position is transmitted as 0xFFFFFFFF.

During heavy activity, the host may momentarily stop accepting data from the FC5025 during a read, causing the FC5025's internal buffer to fill. If the Automatic Overrun Recovery flag is set, the FC5025 will wait for the host to resume accepting data, then locate the sector again and resume reading from where it left off. This is transparent to the host and allows the FC5025 to successfully complete the read. The FC5025 will then send a sense key of RECOVERED ERROR during the CSW phase. If the Automatic Overrun Recovery flag is not set, the FC5025 will instead terminate the read and return an error code.

You should enable Automatic Overrun Recovery for normal sector reads. However, there are some cases where Automatic Overrun Recovery will not operate correctly. For example, if you are using an ID pattern that does not uniquely identify a single sector, the FC5025 may read from the wrong sector after recovering from an overrun. Or, if you are reading the inter-sector gaps, the write splices may read as different data after recovering from an overrun, causing the FC5025 to resume from the wrong point. In these cases, you should disable Automatic Overrun Recovery.

The FC5025 enforces a maximum gap size between ID and data fields to prevent misreads if the first DAM is not detected.

Sense codes

| Key | ASC | ASCQ | Description |
|-----|-----|------|--|
| 04 | 01 | 00 | NO INDEX/SECTOR SIGNAL A hard-sectored read was requested but no index pulses at all were detected within 1.3 seconds. |
| 03 | 11 | 00 | UNRECOVERED READ ERROR Usually caused if the FC5025 stops receiving data partway through a sector, due to a cable problem or similar. May also occur on a flaky disk if it takes multiple revolutions to locate the sector causing the total read time to exceed 1.3 seconds. |
| 0B | 11 | 11 | READ ERROR - LOSS OF STREAMING The host did not accept data from the FC5025 fast enough, causing the FC5025's internal buffer to overflow, and Automatic Overrun Recovery was disabled. |
| 03 | 12 | 00 | ADDRESS MARK NOT FOUND FOR ID FIELD The FC5025 was not able to find any IDAMs within 1.3 seconds. |
| 03 | 13 | 00 | ADDRESS MARK NOT FOUND FOR DATA FIELD The FC5025 found the ID field for the requested sector, but it was not followed by the specified DAM. |
| 03 | 14 | 01 | RECORD NOT FOUND At least one IDAM occurred, but none of the ID fields matched the specified pattern. |
| 04 | 44 | 00 | INTERNAL TARGET FAILURE May be caused by incorrect format setting or bit cell time. |
| 01 | 80 | 01 | RECOVERED LOSS OF STREAMING The sector was successfully read with Automatic Overrun Recovery. |

Recommended timeout

The FC5025 delays for about 1 second after turning on the motor, then internally times out if a read has not completed after about 1.3 seconds. Automatic Overrun Recovery causes the 1.3 second timer to restart. A timeout of 3 or 4 seconds on the host is reasonable.

READ FLEXIBLE operation (0xC6) - Histogram

| <i>Offset</i> | <i>Field</i> | <i>Description</i> |
|---------------|---------------|--|
| 0 | Opcode | Value 0xC6 indicates READ FLEXIBLE operation. |
| 1 | Flags | Bit 0 = Side Select Bits 1-4 = Reserved - always use 0 Bit 5 = Adaptive PLL - 0=enabled, 1=disabled (MFM only) Bits 6-7 = Reserved - always use 0 |
| 2 | Format | 1 Apple GCR 2 Commodore GCR 3 FM 4 MFM |
| 3-36 | ... | Format-dependent parameters |
| 37 | Sub-operation | Set to 1 for histogram. |
| 38 | Sub-parameter | Histogram duration in milliseconds (14ms to 250ms). |

Transfer phase

A histogram operation yields 192 2-byte values, for a total of 384 bytes. Each value is a tally of how many flux transition pairs fell into an 83 1/3 ns wide bin. The first value is always 0, the second is the count of flux transitions that occurred ≤ 83.3 ns after the preceding flux transition, etc. Flux transitions exceeding 15.9us from the preceding one are all counted in the 192nd bin.

Notes

You can perform a histogram on the entire track, or on a specific region. If you specify an ID pattern, etc, like for a read, the FC5025 will locate that sector before beginning the histogram.

To take a histogram of the whole track, use an IDAM value of 0. In this case you may set the format parameter to anything, regardless of the actual format of the disk, but the other parameters must be within the valid range for the format specified in the CDB.

The Adaptive PLL flag controls the PLL behavior when locating a specific sector, if you have specified an ID pattern. The PLL is not used during the histogram itself.

READ ID operation (0xC7) - Apple GCR

| <i>Offset</i> | <i>Field</i> | <i>Description</i> |
|---------------|---------------|---|
| 0 | Opcode | Value 0xC7 indicates READ ID operation. |
| 1 | Side select | Use 0. |
| 2 | Format | Value 1 indicates Apple GCR. |
| 3-4 | Bit cell time | Bit cell time in nanoseconds. |
| 5-7 | Addr prologue | Address prologue. |

READ ID operation (0xC7) - Commodore GCR

| <i>Offset</i> | <i>Field</i> | <i>Description</i> |
|---------------|---------------|--|
| 0 | Opcode | Value 0xC7 indicates READ ID operation. |
| 1 | Side select | Use 0. |
| 2 | Format | Value 2 indicates Commodore GCR. |
| 3-4 | Bit cell time | Bit cell time in nanoseconds. |
| 5-6 | Block ID | Block ID for header blocks, GCR encoded, 10 bits. Low 6 bits of second byte must be zeros. |

READ ID operation (0xC7) - FM

| <i>Offset</i> | <i>Field</i> | <i>Description</i> |
|---------------|---------------|---|
| 0 | Opcode | Value 0xC7 indicates READ ID operation. |
| 1 | Side select | Side 0 or 1. |
| 2 | Format | Value 3 indicates FM ("single density"). |
| 3-4 | Bit cell time | Bit cell time in nanoseconds. |
| 5 | IDAM | Low nibble of IDAM, including clock bits. |

READ ID operation (0xC7) - MFM

| <i>Offset</i> | <i>Field</i> | <i>Description</i> |
|---------------|---------------|---|
| 0 | Opcode | Value 0xC7 indicates READ ID operation. |
| 1 | Side select | Side 0 or 1. |
| 2 | Format | Value 4 indicates MFM ("double density"). |
| 3-4 | Bit cell time | Bit cell time in nanoseconds. |
| 5 | IDAM | IDAM value, MFM encoded and compressed. |

Transfer phase

For Apple GCR, Commodore GCR, FM and MFM, the FC5025 sends 9, 10, 12 or 13 bytes, respectively, for each ID field. The number of ID fields to read is automatically calculated based on the transfer length. The format of the transferred ID fields is the same as in a READ FLEXIBLE operation with the Send Address Field flag set.

Sense codes

| <i>Key</i> | <i>ASC</i> | <i>ASCQ</i> | <i>Description</i> |
|------------|------------|-------------|---|
| 04 | 44 | 00 | INTERNAL TARGET FAILURE May be caused by incorrect format setting or bit cell time. |
| 0B | 11 | 11 | READ ERROR - LOSS OF STREAMING The host did not accept data from the FC5025 fast enough, causing the FC5025's internal buffer to overflow. |
| 03 | 12 | 00 | ADDRESS MARK NOT FOUND FOR ID FIELD No ID fields occurred within 1.3 seconds. |