Future Technology Devices International Ltd

Vinculum Firmware User Manual

Version: 2.3

Issue Date: 2007-12-03

Revision Record Sheet

Authors	Gordon McNab
Filename	**Vinculum Firmware User Manual V2.3 Rev 2.doc**

Revision	Date	Details
1	2007-11-28	Initial Version for Internal Review
2	2006-12-03	Final Revision after Review

Sign Off

Signatory	Signature	Date
Managing Director	**F Dart**	**2007-12-03**
Principal Hardware Engineer	**P Logan**	**2007-12-03**
Principal Software Engineer	**A Miller**	**2007-12-03**
Senior Marketing Manager	**Daniel McCaffrey**	**2007-12-03**
Sales Manager	**Daniel McCaffrey**	**2007-12-03**

Clearance Approval

- This Document is cleared for Future Technology Devices International use and unrestricted circulation.

- An NDA is not required prior to external circulation.

FTDI **#26**
Clearance Number
(Where applicable for external communications)

Table of Contents

List of Tables

List of Figures

1.0 Preface

The objective of this document is to detail the functions performed by the Vinculum VNC1L firmware, the differences between each firmware version and the connections required to be made to the device.

1.1 Conventions

The following conventions are used in this document:

Convention	Description
`Monospaced type`	Indicates input or output from the monitor.
`Boldface monospaced type`	Indicates input supplied by the user.
`Italic monospaced type`	Indicates binary characters (ASCII values of characters).
Important	Signals that the information supplied is important.
Note	Provide additional information about a topic.
Warning	Indicate potential damage to equipment or loss of data.
:	Is used to show a range. For instance, a range of bits 15:9 is bits 15 to 9 (inclusive) of a binary value.
VNC1L	Indicated information specific to the Vinculum VNC1L device.
↵	carriage return (0x0D).
·	space (0x20).
d	Represents a single decimal character (0 to 9).
x	Represents a single hexadecimal character (0 to 9 and A to F).
c	Represents a binary character (0x00 to 0xFF).

Table 1.1 Document Conventions

1.2 References

This document does not describe the hardware interfaces required to connect a microcontroller to a VNC1L device nor does it provide application notes. The following documents are available from FTDI and other sources for this purpose:

Document Reference	Description
Vinculum Website	The main website for the Vinculum family of USB Host Controllers http://www.vinculum.com/
DS_VNC1L-1A	Vinculum Embedded USB Host Controller IC Data Sheet http://www.vinculum.com/
FTDI FT232 / FT245 / FT2232	FTDI FT232 / FT245 / FT2232 Data Sheet http://www.ftdichip.com/
D2XXPG	D2XX Programmer's Guide http://www.ftdichip.com/
VS1003	VS1003 Data Sheet VLSI Solution Oy http://www.vlsi.fi/
USB 2.0	Universal Serial Bus Specification Revision 2.0 USB Implementers Forum http://www.usb.org/

Table 1.2 Other Documents Available

1.3   Acronyms and Abbreviations

The following terms are used within this document:

Terms	Description
8.3	A filename format consisting of 1 to 8 characters, optionally followed by a period ('.') then an option extension of up to 3 characters. For example, "TEST.TXT", "ANEWFILE.1" or "AFILE".
FAT	File Allocation Table. The name for the file system typically used for USB Flash Disks. Variants are FAT12, FAT16 and FAT32, the numbers referring to the number of bits used to specify a cluster on the disk.
BOMS	Bulk Only Mass Storage. A general description given to a USB mass storage device e.g. USB Flash Disk.
MBR	Master Boot Record. Part of the FAT file system.
Monitor	Command line interface which allows instructions to be given to the VNC1L and responses to be returned.
LSB	Least Significant Byte is shown first when representing a multi-byte number as a sequence of bytes. E.g. a number 0x55AAFF00 is stored or specified as 4 bytes 0x00, 0xFF, 0xAA and 0x55.

Terms	Description
MSB	Most Significant Byte is shown first when representing a multi-byte number as a sequence of bytes. E.g. a number 0x55aaff00 is stored or specified as 4 bytes 0x55, 0xaa, 0xff and 0x00.
Sector Size	The size of data which is read from or written to a disk in a single operation.
Cluster Size	The size of data block allocated on a disk to store files or directory structures. Integer multiples of clusters must be used to store large files or directories.
Root Directory	The topmost directory in a FAT file system.

Table 1.3 Acronyms and Abbreviations

2.0   Introduction

The Vinculum VNC1L device has two USB host ports as well as a combined serial UART, SPI or FIFO interface. The combined interface is selectable using a pair of pins connected to either pull-ups or pull-downs.

The firmware for the Vinculum VNC1L allows a command Monitor port to be active on either the combined interface or one of the USB Ports. The main function of the Monitor is to allow an embedded device to communicate via the VNC1L's UART, parallel FIFO or SPI interface port with USB slave peripheral devices.

Typical slave peripheral devices include USB Flash disks; FTDI's FT232, FT245 and FT2232; as well as devices that are Printer class or HID (Human Input Device) class compatible. Other devices may also be supported if they have a suitable interface. For instance, CDC class equipment such as some mobile phones.

The 2 USB Ports are referred to as Port 1 and Port 2 in this document.

2.1   Command Monitor

Communicating with and controlling the Vinculum VNC1L is achieved by sending instructions to the command monitor and interpreting the results. The command monitor provides an interface which can be utilised by a microcontroller to read and send data. A prompt is returned when the Vinculum VNC1L is ready to execute a command. Status and error information is returned after a command has completed and event notifications are relayed to the monitor.

2.2   Command Monitor Modes

The Vinculum VNC1L monitor works in command mode or data mode in a similar way to a modem. Command mode is used to communicate with the VNC1L. Data mode is used to communicate with a slave device on USB Port 1 or 2.

3.0   Firmware Functionality

There are several pre-compiled firmware codes available:

- VDAP (disk and peripherals)
- VDIF (disk and FTDI Interface)
- VMSC (music player)
- VDPS (disk, PC monitor and slave port)
- VCDC (communication class device)

This section discusses the functionality and characteristics of each version.

3.1   Firmware Functionality

Several classes of device can be connected to the USB Ports of the Vinculum VNC1L. Each version of firmware code allows a different combination of devices and monitor ports. By default, the monitor port is active on the combined interface. The exception is the VDIF firmware where the monitor port will automatically move to USB Port 1 if a suitable FTDI USB Slave device is connected.

3.1.1 USB Slave Peripherals

The classes of devices which may be connected to each USB Port are shown for each firmware code shown in Table 3.1.

Device Type	BOMS		FTDI Peripheral		HID		Printer		Hub		PC Host		Monitor Port		CDC	
USB Port	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
VDAP		Y	Y	Y	Y	Y	Y	Y	Y	Y						Y	Y
VDIF		Y		Y		Y		Y					Y			
VMSC		Y		Y		Y		Y		Y						Y
VDPS		Y		Y		Y		Y		Y	Y					
VCDC		Y		Y		Y		Y		Y					Y	Y

Table 3.1 Firmware Functionality

VDAP firmware is the most general purpose and supports most devices on Port 1 and Port 2. The VMSC and VCDC builds are based on the VDAP code and so provide similar functionality. VDPS firmware is required if the Vinculum VNC1L is to be used as both a host and a slave device. VDIF should only be used in exceptional applications where the Monitor must be accessible on USB Port 1.

Note : The VMSC is provided with additional commands specifically for the VMUSIC range of devices to permit playback of MP3 files via a VLSI VS1003 MP3 decoder. The VS1011 MP3 decoder is also supported.

3.1.2 General Limitations

All firmware versions support BOMS devices formatted in FAT12, FAT16 or FAT32 file systems only where the sector size is 512 bytes. No other file systems or sector sizes are allowed. Furthermore, only BOMS devices which support the SCSI or AT command sets are supported.

3.1.3 Firmware Upgrades

There are two methods of upgrading the firmware on the Vinculum VNC1L.

The first is to use the ROM programmer tool, VPROG, available from the Vinculum website (see Section 1.2). This takes the ROM files supplied on the Vinculum website and programs the device *via* a PC serial port or an FT232, FT245 or FT2232 device. This method requires that the PROG pin on the device is pulled low to instantiate a special hardware mode which programs the device's internal FlashROM.

An alternative is to use a Flash Disk to hold the upgrade file. When the Flash Disk is inserted and detected by the Vinculum VNC1L it checks for a file called 'FTRFB.FTD' in the root directory of the disk. If this file is found it will verify the format of the file matches a Vinculum VNC1L firmware upgrade file. If the file format is correct then it will upgrade the firmware with the firmware image from the file. FTD files are supplied on the Vinculum website but will require renaming to 'FTRFB.FTD' to be recognised as firmware upgrade files.

3.2 VDAP Firmware

Figure 3.1 demonstrates using the Vinculum VNC1L to provide USB Host functionality to a microcontroller, which is a typical application for VDAP firmware. This will have the monitor port on the combined interface allowing BOMS devices to be connected to Port 2 and USB Slave Peripherals to Port 1.
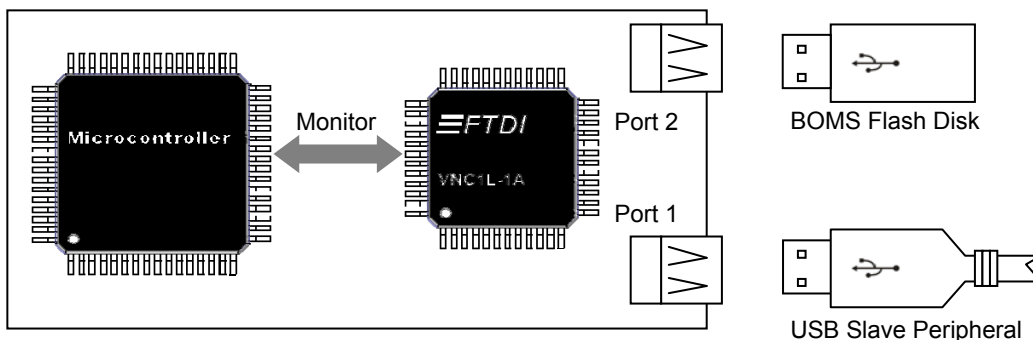
Figure 3.1 Vinculum VNC1L Connected to Microcontroller

3.3 VMSC Firmware

VMSC firmware is similar to VDAP firmware but introduces new commands to allow playback through an MP3 decoder chip. Data is sent to a VLSI VS1003 MP3 decoder over an SPI bus and control of playback is performed by a microcontroller, as illustrated in Figure 3.2.
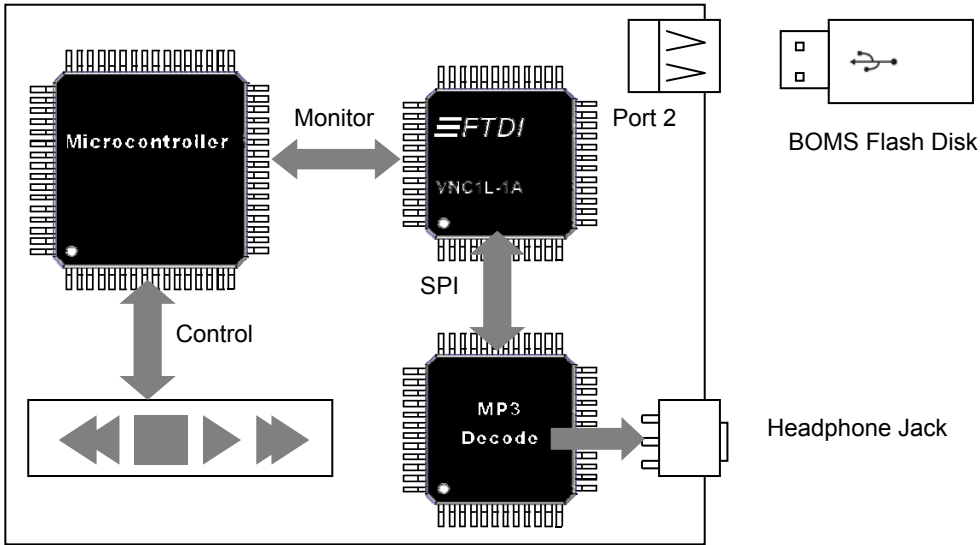
Figure 3.2 Vinculum VNC1L Connected to MP3 Decoder and Microcontroller

The VLSI VS1003 is referenced in Table 1.2.

Current track filename is displayed when playback of a track is started. The elapsed time of the current track is displayed every second.

VMSC firmware relies on hardware connections to devices found on a VMUSIC1 or VMUSIC2 module and will therefore not function with other modules. The VMUSIC1 and VMUSIC2 modules do not include USB Port 1.

3.4   VDIF Firmware

VDIF firmware is designed to work in a similar manner as VDAP firmware until a suitable FTDI USB Slave Peripheral is detected on USB Port 1. When this occurs, the monitor ceases to be active on the combined interface and becomes active on the FTDI USB Slave Peripheral, as illustrated in Figure 3.3. The firmware will initially use the same baud rate and flow control settings of the combined interface UART.
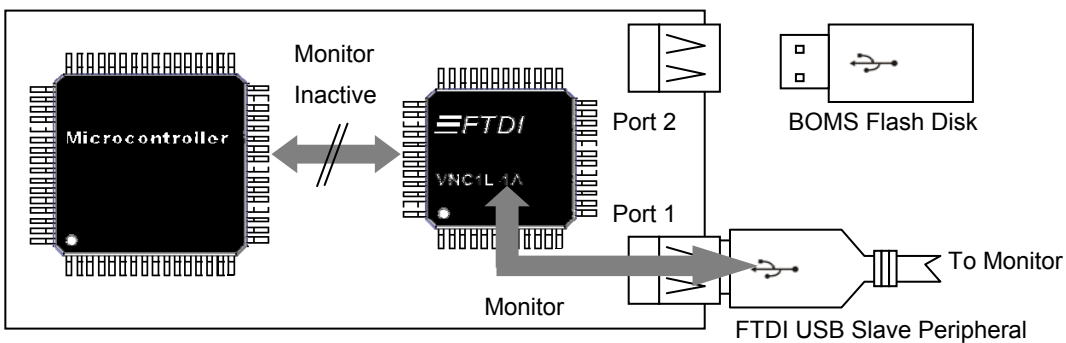
Figure 3.3 Vinculum VNC1L with Monitor on FTDI Slave Peripheral

Suitable FTDI devices to connect to VDIF firmware are the FT232B, FT232R or FT2232. The Serial UART output of these FTDI devices can be connected to Serial UARTs for interaction with the Monitor.

If the Monitor is not required to be available on USB Port 1 then VDAP firmware should be used instead. This firmware is not recommended unless there are specific reasons for using the Monitor on USB Port 1 and as such is only available on request from Technical Support.

3.5 VCDC Firmware

Selected communication class devices can be connected to the Vinculum VNC1L to provide modem-like functionality to microcontrollers. The CDC device must support the sub-class of Abstract Control Model and have compatible bulk endpoints. The CDC firmware requires that call management information can be sent or received over the Data Class interface. A connection to a CDC device is shown in Figure 3.4.
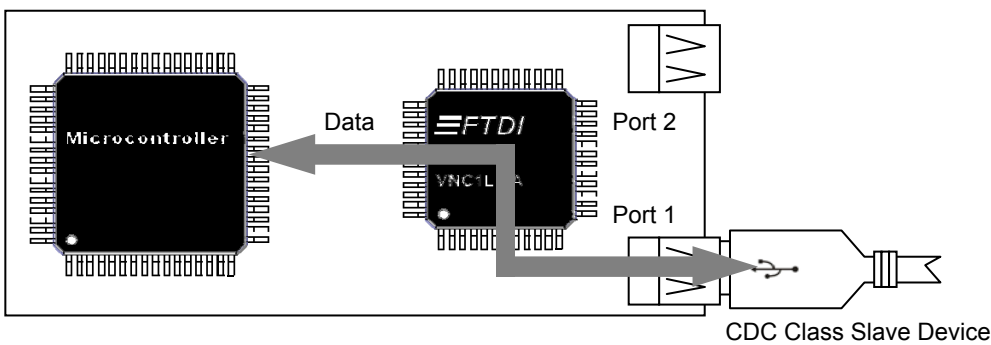
Figure 3.4 Vinculum VNC1L Connected to CDC Class Device

When a CDC device is connected to either USB port on the Vinculum VNC1L it is automatically connected in Data Mode to the Monitor interface. There are separate output pins for modem control (RI#, DCD#) used by the VCDC firmware while the connection is active.

Because of the strict compatibility requirements of this firmware it is only available on request from Technical Support.

3.6 VDPS Firmware

VDPS firmware can be used if a microcontroller is to be connected to a host port on a PC. The Vinculum VNC1L will enumerate as an FTDI FT232B device and load drivers accordingly, as shown in Figure 3.5.
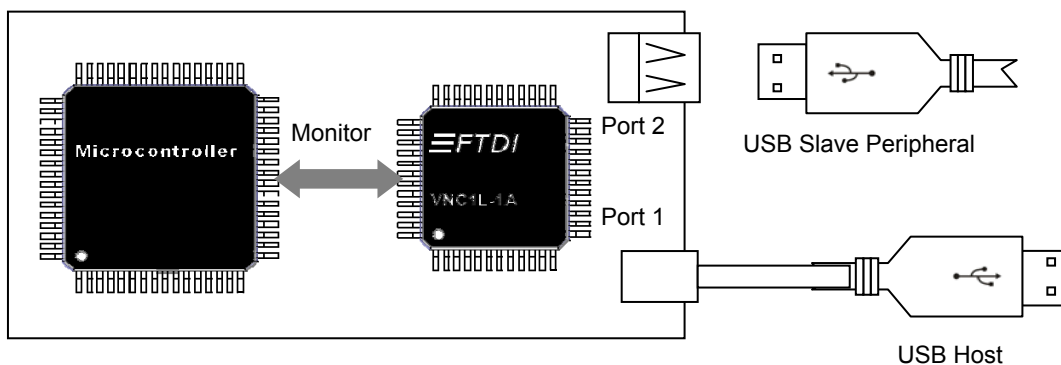
Figure 3.5 Vinculum VNC1L Connected to Host

To operate a Vinculum VNC1L in VDPS mode, the voltage from the USB connection to the host PC must be fed-back through a resistor voltage divider (consisting of 2 100kΩ resisters between USBVCC and USBGND) into pin BD7. Refer to the diagram in Figure 3.6.
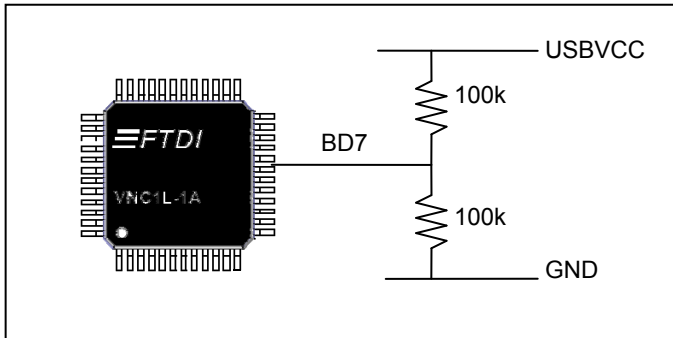
Figure 3.6 Vinculum VNC1L in VDPS mode

4.0 Vinculum VNC1L Configuration

This section discusses the configuration and operating modes of the Vinculum VNC1L Command Monitor.

4.1 Monitor Selection

There are three interface options for the monitor on the combined interface: Serial UART, parallel FIFO and SPI. The pin connections used to select the mode of the interface are shown in Table 4.1.

ACBUS6 (pin 47)	ACBUS5 (pin 46)	Mode
Pull-Up	Pull-Up	Serial UART
Pull-Up	Pull-Down	SPI
Pull-Down	Pull-Up	Parallel FIFO
Pull-Down	Pull-Down	Serial UART

Table 4.1 Combined Interface Selection

Important : Pins ACBUS5 and ACBUS6 should not be tied directly to GND or VCC. Instead, they should be pulled high or low using a resistor of around 47kΩ. These pins are read only at reset, but may then become outputs after the interface choice has been selected (e.g. ACBUS5 becomes DATAACK# if FIFO mode is selected).

The Vinculum VNC1L device pin definitions according to the selected mode of the combined interface are shown in Table 4.2. A full pin-out is available in the Vinculum VNC1L Datasheet which is referenced in Table 1.2. Pins which are marked N/A are not available for use in the selected mode.

Pin No.	Name	Type	Serial UART	Parallel FIFO	SPI
31	ADBUS0	I/O	TXD	D0	SCLK
32	ADBUS1	I/O	RXD	D1	SDI
33	ADBUS2	I/O	RTS#	D2	SDO
34	ADBUS3	I/O	CTS#	D3	CS
35	ADBUS4	I/O	DTR# / DATAACK#	D4	N/A
36	ADBUS5	I/O	DSR# / DATAREQ#	D5	
37	ADBUS6	I/O	DCD#	D6	
38	ADBUS7	I/O	RI#	D7	
41	ACBUS0	I/O	TXDEN#	RXF#	
42	ACBUS1	I/O		TXE#	
43	ACBUS2	I/O		RD#	
44	ACBUS3	I/O		WR	
45	ACBUS4	I/O		DATAREQ#	DATAREQ#
46	ACBUS5	I/O		DATAACK#	DATAACK#

Table 4.2 Combined Interface Pins

When using the Serial UART interface as the command monitor port the default baud rate used is 9600 baud, 8 data bits, 1 start bit, 1 stop bit, and no parity with RTS/CTS hardware handshaking enabled. However, this can be changed while in command mode.

4.2   Monitor Operating Modes

The Vinculum VNC1L can switch between two modes that change the method of communicating with USB Slave devices. In command mode, the Vinculum VNC1L interprets and acts on commands; in Data Mode information from the monitor port is passed straight to another device.

Signals DATAREQ# and DATAACK# allow for switching between command mode and data mode. In Serial UART interface mode they are on the DTR# and DSR# interface pins. In parallel FIFO and SPI modes they are on pins 45 and 46.

4.2.1 Command Mode

When the Vinculum VNC1L firmware starts it is in Command Mode. In this mode the DATAACK# signal is high and the DATAREQ# signal should be held high to stay in command mode.

While in command mode commands are provided to configure the Serial UART interface. These commands can also configure a suitable FTDI USB Slave UART device on USB Port 1. Access to BOMS devices on Port 2 is allowed. This is illustrated in Figure 4.1.
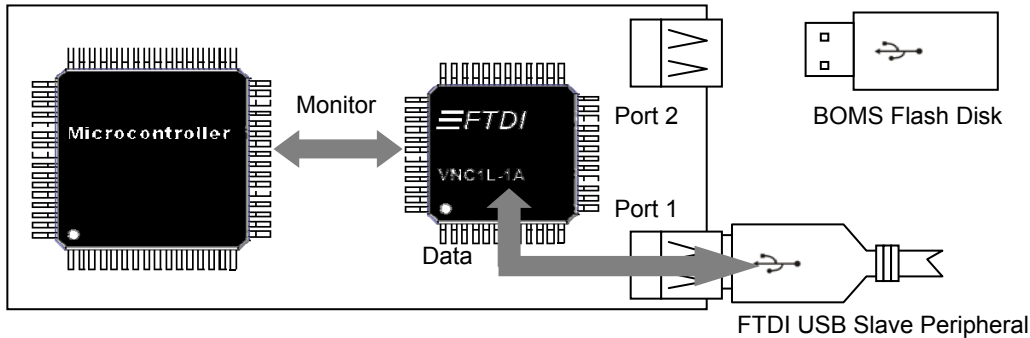
Figure 4.1 Command Mode Connection

4.2.2 Data Mode

In order to switch to data mode the DATAREQ# line should be asserted low. Once the DATAACK# line goes low, any data sent to the Monitor port on the Vinculum VNC1L will be sent to a USB Slave device connected to either of the USB Ports. Any data received from the USB Slave device will be repeated to the Monitor port.

In this mode the VNC1L ignores the data and simply passes it between USB and the Monitor port. This is illustrated in Figure 4.2.
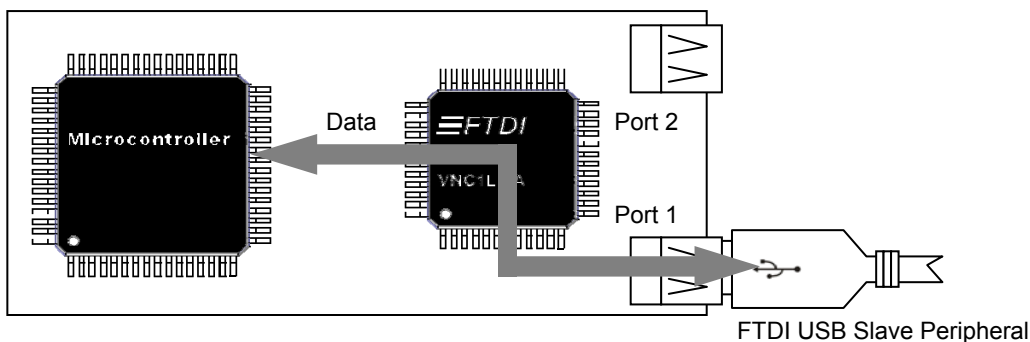
Figure 4.2 Data Mode Connection

There is a special case for compatible CDC Class slave devices in Data Mode on VDAP firmware. Once Data Mode is entered using the DATAREQ# line, a CDC Class driver is invoked to handle the connection and communications between the Monitor's combined interface and the CDC device. This is equivalent to the connection obtained with VCDC firmware but allows control over when to initiate the connection to the CDC device, however, the DTR# and DSR# lines from the modem are not available.

5.0   Monitor Introduction

This section discusses the various methods of interacting with the Monitor.

5.1   Monitor Modes

There are two command entry modes supported in the Monitor - Extended or Short. These determine the way a command is entered and how prompts are displayed. In extended mode, printable characters are used and commands are typically longer. In short command mode, the commands are optimised for programmatical control and have binary values representing the command.

Note : After reset the Monitor defaults to Extended Command Set.

5.2   Numerical Modes

The two numerical modes available for Monitor commands are independent of the selected command set. ASCII Mode (invoked using the IPA command) uses printable characters; Binary Mode (selected by the IPH command) uses binary values.

In binary mode values are always a fixed number of bytes depending on the command. In ASCII mode, an inputted decimal or hexadecimal number is entered and converted from hexadecimal if it starts with '$' or '0x', otherwise it is assumed to be decimal. There is a limitation of 8 characters for a decimal value in ASCII mode, this limits decimal numbers to 99,999,999 (i.e. 0x5F5E0FF in hexadecimal).

Output values are translated to printable characters by formatting each byte as a hexadecimal number prefixed with a '$' symbol. Data returned from devices is not translated into printable characters.

Note : After reset the Monitor defaults to Binary Mode.

Important : In binary mode values are always input MSB first. However, all output in binary or ASCII from the Monitor is LSB first.

5.2.1 Number Input

As an example of ASCII Mode input, the following are all valid and produce the same results. Note the use of Extended Command Set.

```
SBD·$384100↵
SBD·0x384100↵
SBD·3686656↵
SBD·03686656↵
```

The first two examples use hexadecimal input: '0x' and '$' are equivalent prefixes to signify hexadecimal values. The last two examples are the same number 0x384100 converted to decimal. The leading 0 is ignored.

If the command were sent in binary mode using the Short Command Set the command would be sent as the following sequence of bytes:

14 20 38 41 00 0D

Binary numbers are always input MSB first.

5.2.2 Number Output

When a command returns numbers in ASCII mode, which are not raw data from a device, it will translate them into a sequence of hexadecimal numbers. Here, the value 0x000310aa is shown as it would be returned from the Monitor.

`$AA·$10·$03·$00·↵`

Note : After the final value in an ASCII mode number, there is an additional space character before the carriage return. In binary mode the same value would be returned as the following sequence of bytes:

AA 10 03 00 0D

Note : The carriage return character is sent to indicate the end of the response to the command.

5.3   Filenames

Filenames generated by the Vinculum VNC1L Monitor must be uppercase letters and numbers or one of the following characters:

`$ % ' - _ @ ~ ` ! ( ) { } ^ # &`

The firmware uses the Windows 1252 (West European Latin) codepage to translate lowercase to uppercase letters automatically. It will generate an error if a filename contains invalid characters.
ASCII characters in the range of 128 to 255 are allowed. Two-byte character sets may be used where bit 7 in the first byte signifies that the second byte is a continuation of the first.

Important: Long file names in FAT32 file systems are not supported.

5.4   Monitor Startup

Upon starting the Monitor the following string is displayed:

`↵`
`Ver·xx.xxVcccF·On-Line:↵`

This indicates the firmware version number (xx.xx) and the firmware type (Vccc). For instance, version V03.60 with VDAP firmware would report:

`↵`
`Ver·03.60VDAPF·On-Line:↵`

5.5   Command Responses

Typically after a command has completed the monitor will show a prompt or an error response to indicate that it is ready for more commands. Prompts and error responses are always followed by a carriage return character.
.

5.5.1 Successful Command Prompt

The appropriate response from Table 5.1 is displayed upon successful completion of a command.

Extended Command Set	Short Command Set
D:\>↵	>↵ (*3E 0D*)

Table 5.1 Monitor Prompts for Successful Command Completion

5.5.2 Empty Command Prompts

A single carriage return is used to query the presence of a disk. The responses which the Monitor returns are shown in Table 5.2. The command prompt does not reflect the current directory on a disk.

Disk Present	Extended Command Set	Short Command Set
Yes	D:\>↵	>↵ (*3E 0D*)
No	No Disk↵	ND↵ (*4E 44 0D*)

Table 5.2 Monitor Prompts with No Command Issued

5.5.3 Error Messages

If a command is not recognised then the monitor will respond with a Bad Command error. A complete list of error or failure responses is given in Table 5.3. These responses also serve as a prompt for the next Monitor command.

Extended Command Set	Short Command Set	Reason
Bad·Command↵	BC↵ *(42 43 0D)*	Command not recognised.
Command·Failed↵	CF↵ *(43 46 0D)*	Filename or directory name not found.
Disk·Full↵	DF↵ *(44 46 0D)*	No free space on disk.
Invalid↵	FI↵ *(46 49 0D)*	Attempt to open a directory for reading or writing. Attempt to change currently selected directory to a file.
Read·Only↵	RO↵ *(52 4F 0D)*	Attempt to open a read only file for writing.
File·Open↵	FO↵ *(46 4F 0D)*	A file is currently open for writing and must be closed before this command can be executed.
Dir·Not·Empty↵	NE↵ *(4E 45 0D)*	Attempt to delete a directory which is not empty.
Filename·Invalid↵	FN↵ *(46 4E 0D)*	Firmware invalid or contains disallowed characters
No·Upgrade↵	NU↵ *(4E 55 0D)*	Firmware Upgrade file not found on disk

Table 5.3 Monitor Error Responses

5.6   Monitor Events

The Monitor will send notification messages when a device has been inserted or removed from either USB Port.

5.6.1 Slave Devices

When a USB Slave Device is inserted and detected in either USB Port 1 or USB Port 2 an event message as described in Table 5.4 will be shown. The message will be displayed at start-up of the firmware before the initial prompt is displayed if a device is connected. The only exception is VDIF firmware which does not display event messages for USB Port 1 since this is used as a Monitor port.

5.6.2 Disk Drives

If a disk is connected to USB Port 2 then the root directory of the disk is searched for a firmware upgrade file. The file must be named 'FTRFB.FTD'. Further verification is performed on the file to ensure that it is a valid firmware upgrade file before the device is reprogrammed. If there is no upgrade file in the root directory or the file is not a valid upgrade file the No Upgrade message in Table 5.3 will be shown.

The following messages will be displayed when a disk without a firmware upgrade file is inserted into USB Port 2.

```
Device·Detected·P2↵
No·Upgrade↵
D:\>↵
```

5.6.3 Host Device

On VDPS firmware when the Vinculum VNC1L is connected to a PC Host Port the Slave Enabled message is shown from Table 5.4. When disconnected, the Slave Disabled message is shown. Connection and disconnection are defined as the point when DATAREQ# and DATAACK# are used to enter data mode.

Event	Extended Command Set	Short Command Set
Slave Device inserted in USB Port 1	Device·Detected·P1↵	DD1↵ *(44 44 31 0D)*
Slave Device removed from USB Port 1	Device·Removed·P1↵	DR1↵ *(44 52 31 0D)*
Slave Device inserted in USB Port 2	Device·Detected·P2↵	DD2↵ *(44 44 32 0D)*
Slave Device removed from USB Port 2	Device·Removed·P2↵	DR2↵ *(44 52 32 0D)*
Connected to Host Device	Slave·Enabled↵	SDA↵ *(53 44 41 0D)*
Disconnected from Host Device	Slave·Disabled↵	SDD↵ *(53 44 44 0D)*

Table 5.4 Monitor Events

6.0   Monitor Command Set

In the following descriptions, values which are received from or sent to the monitor are represented in *italics* using the codes below:

`file`	valid filename or directory name
	any uppercase letter and number combination plus $ % ` – _ @ ~ ` ! ( ) { } ^ # &
`date`	16 bit value representative of a FAT format file date (See Table 6.4)
`datetime`	32 bit value representative of a FAT format file date and time (See Table 6.4)
`divisor`	3 byte value encoding a baud rate (See Table 6.2)
`qword`	64 bit value (8 bytes)
`dword`	32 bit value (4 bytes)
`word`	16 bit value (2 bytes)
`byte`	8 bit value (1 byte)
`data`	Information returned from an operation which is passed to or from the Monitor unmodified. It is not formatted as hexadecimal numbers when the IPA mode is selected.

In addition, the following symbol is used when describing the Extended Command Set:

+	indicates that two or more values are not separated by a space
	e.g. `byte+word` is a three byte value

6.1   Monitor Configuration Commands

This class of command is used to setup the Monitor and detect the version of firmware running on the Vinculum VNC1L. Commands are listed in Table 6.1.

Extended Command Set	Short Command Set (Hexadecimal Codes)	Function
SCS↵	*10 0D*	Switches to the shortened command set
ECS↵	*11 0D*	Switches to the extended command set
IPA↵	*90 0D*	Monitor commands use ASCII values
IPH↵	*91 0D*	Monitor commands use binary values
SBD·*divisor*↵	*14 20 divisor 0D*	Change monitor baud rate
FWV↵	*13 0D*	Display firmware version
E↵	*45 0D*	Echo 'E' for synchronisation
e↵	*65 0D*	Echo 'e' for synchronisation

Table 6.1 Monitor Configuration Commands

6.1.1 Short Command Set (SCS)

Parameters: **None.**

Switches the Monitor's mode to Short Command codes.

Note : Either Short or Extended Commands Set versions of this command may be sent to the Monitor while it is in Short or Extended Command mode.

Error Codes: **None.**

6.1.2 Extended Command Set (ECS)

Parameters: **None.**

Switches the Monitor's mode to Extended Command codes.

Note : Either Short or Extended Commands Set versions of this command may be sent to the Monitor while it is in Short or Extended Command mode.

Error Codes: **None.**

6.1.3 Monitor Mode ASCII (IPA)

Parameters: **None.**

Switches the Monitor's mode to input and display values in printable ASCII characters.

Error Codes: **None.**

6.1.4 Monitor Mode Binary (IPH)

Parameters: **None.**

Switches the Monitor's mode to input and display values in binary format.

Error Codes: **None.**

6.1.5 Set Baud Rate (SBD)

Parameters:

Baud Rate – 3 Bytes: Encoded baud rate.

Upon changing the baud rate of the Monitor a prompt is sent at the previous baud rate before the change is made. Another prompt is sent after the change has been made to the new baud rate.

The encoded baud rate is represented as 3 bytes from Table 6.2. In binary input mode the three bytes are specified in the order which they should be used in the commands. In ASCII mode byte 1 is the MSB of the numerical value, e.g. 9600 baud would be 384100 when entered in the SBD command. In binary mode the value would be *38 41 00.*

Note : This only affects the Monitor when in Serial UART mode or, with VDIF firmware, connected to a Monitor port through an FTDI USB Slave peripheral.

Baud Rate	1st Byte	2nd Byte	3rd Byte
300	0x10	0x27	0x00
600	0x88	0x13	0x00
1200	0xC4	0x09	0x00
2400	0xE2	0x04	0x00
4800	0x71	0x02	0x00
9600	0x38	0x41	0x00
19200	0x9C	0x80	0x00
38400	0x4E	0xC0	0x00
57600	0x34	0xC0	0x00

Baud Rate	1st Byte	2nd Byte	3rd Byte
115200	0x1A	0x00	0x00
230400	0x0D	0x00	0x00
460800	0x06	0x40	0x00
921600	0x03	0x80	0x00
1000000	0x03	0x00	0x00
1500000	0x02	0x00	0x00
2000000	0x01	0x00	0x00
3000000	0x00	0x00	0x00

Table 6.2 Monitor Baud Rates

There are no error responses generated by this command.

Error Codes: **None.**

6.1.6 Firmware Version (FWV)

Parameters: **None.**

Retrieves the version of the current monitor firmware and the reflasher code. The data is presented in the following order:

```
FWV↵
↵
MAIN·dd.ddAAAAA↵
RPRG·d.ddR↵
<prompt>
```

Note the blank line preceding the returned values.

Error Codes: **None.**

6.1.7 Echo (E, e)

Parameters: **None.**

Repeats either an upper case 'E' or a lower case 'e' followed by a carriage return character. This is primarily used for synchronisation purposes.

Error Codes: **None.**

6.2   Disk Commands

If a BOMS class device is in USB Port 2 the commands in this section may be used to interact with the file system on the disk. FAT12, FAT16 or FAT32 file systems may be used. Disk Commands are listed in Table 6.3.

There are limitations to the type of disk which can be addressed with the Vinculum VNC1L firmware. All disks must have a sector size of 512 bytes. Most Flash-based and many disk-based drives will have a 512 byte sector size. Various cluster sizes have been tested up to 32kB.

For file I/O, the firmware requires files to be opened using the OPW or OPR commands, written to or read from with the WRF or RDF commands and then closed with the CLF command.

Only one file may be open at any one time. If a file is open for writing, then it may be written to or read from. If open for read then it may only be read from. A file pointer is maintained for the currently open file from where reads or writes will commence. The SEK command can be used to move the file pointer within a file. However, files open for write will be truncated at the file pointer when closed. The end of the file is moved to the position of the file pointer after a write operation.

There is no mechanism for determining the current directory on a disk. It is possible to ascertain if the current directory is the root directory by checking for the absence of the '.' and '..' directories in a directory listing. The prompt in both Extended and Short Command Sets does not change when the current directory is changed with the CD command.

If a disk is incorrectly formatted or the file system is not recognised then the "No Disk" prompt (see Table 5.2) will be returned.

To determine the free space available on the disk the FS or FSE commands must be used. This is not performed at startup in Firmware V03.61 onwards.

As the disk fills, due to file system fragmentation, the firmware may take a short time to find an unused cluster. This can happen mid-way during a WRF write operation or when an OPW or MKD is called. When there are no remaining free clusters on the disk and the firmware searches for a free cluster it will scan the entire disk once then report "Disk Full". This final search for a cluster takes a variable length of time depending on the disk geometry and speed.

If the disk is full and a OPW or MKD command are called the firmware reports "Disk Full" and no file or subdirectory is created. If the disk fills during a WRF operation then the command will continue to accept data until the amount of data specified in the command call has been received then report "Disk Full". The file size will be set to the size of the file when the disk filled and the file will be truncated at this point.

Important: Long file names in FAT32 file systems are not supported.

Warning: If a BOMS device is removed during a write operation, or even while a file is open for writing, then data corruption is likely.

Extended Command Set	Short Command Set (Hexadecimal Codes)	Function
DIR↵	01 0D	List files in current directory
DIR·file↵	01 20 file 0D	List specified file and size
CD·file↵	02 20 file 0D	Change current directory
CD·..↵	02 20 2E 2E 0D	Move up one directory level
RD·file↵	04 20 file 0D	Reads a whole file
DLD·file↵	05 20 file 0D	Delete subdirectory from current directory
MKD·file↵	06 20 file 0D	Make a new subdirectory in the current directory
MKD·file·datetime↵	06 20 file 20 datetime 0D	Make a new subdirectory in the current directory Also specify a file date and time
DLF·file↵	07 20 file 0D	Delete a file
WRF·dword↵ data	08 20 dword 0D data	Write the number of bytes specified in the 1st parameter to the currently open file
OPW·file↵	09 20 file 0D	Open a file for writing or create a new file
OPW·file·datetime↵	09 20 file 20 datetime 0D	Open a file for writing or create a new file Also specify a file date and time
CLF·file↵	0A 20 file 0D	Close the currently open file
RDF·dword↵	0B 20 dword 0D	Read the number of bytes specified in the 1st parameter from the currently open file
REN·file·file↵	0C 20 file 20 file 0D	Rename a file or directory
OPR·file↵	0E 20 file 0D	Open a file for reading
OPR·file·date↵	0E 20 file 20 date 0D	Open a file for reading Also specify a file access date
SEK·dword↵	28 20 dword 0D	Seek to the byte position specified by the 1st parameter in the currently open file
FS↵	12 0D	Returns the free space available on disk if less than 4GB is free
FSE↵	93 0D	Returns the free space available on disk
IDD↵	0F 0D	Display information about the disk if disk is less than 4GB

Extended Command Set	Short Command Set (Hexadecimal Codes)	Function
IDDE↵	94 0D	Display information about the disk
IDD↵	0F 0D	Display information about the disk
DSN↵	2D 0D	Display disk serial number
DVL↵	2E 0D	Display disk volume label
DIRT·file↵	2F 20 file 0D	List specified file and date and time of create, modify and file access

Table 6.3 Disk Commands

32 Bit Values	16 Bit Values	Description	Allowable Values	Meaning
25:31	9:15	Year	0 – 127	0 = 1980 127 = 2107
21:24	5:8	Months	1 – 12	1 = January 12 = December
16:20	0:4	Days	1 – 31	1 = first day of month
11:15	N/A	Hours	0 – 23	24 hour clock
5:10	N/A	Minutes	0 – 59	
0:4	N/A	Seconds/2	0 – 29	0 = 0 seconds 29 = 58 seconds

Table 6.4 File Date and Time Bit Fields

6.2.1 Directory (DIR)

Parameters:

Filename – (Optional) Up to 11 characters ASCII.

This can either list the available files in the current directory (no parameters passed) or show the file size of a specified file. The filename is typically formatted as a 8.3 filename. Note that if the specified file is a directory, a size of zero will be displayed rather than a listing of the files in that directory.

When listing the files in the current directory, only files which can be accessed are shown. Files which include long file names, disk labels or FAT32 extensions are ignored.

The data from the command with no parameters, where the current directory is the root directory, is presented as follows:

```
DIR↵
↵
FILE1.XXX↵
FILE2.↵
```

```
SUBDIR1·DIR↵
NEWSUBDIR·DIR↵
<prompt>
```

If the current directory is a subdirectory then the file listing will have '.' and '..' directories for the current and higher-

level directories respectively.

```
DIR↵
↵
.·DIR↵
..·DIR↵
FILE1.XXX↵
<prompt>
```

Note the blank line preceding the returned values. Directories are signified by the letters "DIR" after the name. There is

no 8.3 formatting applied to directory names greater than 8 characters in length. Filenames are always shown in 8.3

format and the dot for the extension is shown if no file extension is given to the file.

If the filename is actually a subdirectory the command will not list the contents of the subdirectory, as with some

operating systems, but will show the subdirectory as if it were a zero-length file.

In ASCII input mode the output from the command with a valid filename specified is:

```
DIR·file1.XXX↵
↵
FILE1.XXX·$xx·$xx·$xx·$xx·↵
<prompt>
```

Note the space after the last hexadecimal number before the carriage return.

In binary input mode the output from the command will be:

```
DIR·file1.XXX↵
↵
FILE1.XXX·cccc↵
<prompt>
```

Error Codes:

Command Failed – The filename, if specified, does not exist.

File Open – A file is currently opened by a VMUSIC command. Stop playback on the VMUSIC before retrying this

command.

6.2.2 Change Directory (CD)

Parameters:

Filename – Up to 11 characters ASCII.

Changes the current working directory to the subdirectory specified in the parameter. If the filename parameter is '..'

then the current directory is changed to the higher-level directory.

Error Codes:

Command Failed – The filename specified does not exist.

File Invalid – The filename specified is used in the current directory as either a filename, FAT volume label or other FAT

structure.

File Open – A file is currently open for writing or is being accessed by a VMUSIC command. Close the open file first before retrying this command.

6.2.3 Read File (RD)

Parameters:

Filename – Up to 11 characters ASCII.

This command sends the entire contents of a file to the Monitor. If the number of bytes expected to be received from the Monitor is required then a DIR command should be done to find out the size of the file beforehand.

Error Codes:

Command Failed – The filename specified does not exist.

File Open – A file is currently open for writing. Close the open file first before retrying this command.

6.2.4 Delete Directory (DLD)

Parameters:

Filename – Up to 11 characters ASCII.

If the subdirectory in the filename parameter is empty then the subdirectory is removed.

Error Codes:

Command Failed – The filename specified does not exist.

Dir Not Empty – There are files or subdirectories in the subdirectory to be deleted. The subdirectory must be empty before the operation can be completed.

File Open – A file is currently opened by a VMUSIC command. Stop playback on the VMUSIC before retrying this command.

6.2.5 Make Directory (MKD)

Parameters:

Filename – Up to 11 characters ASCII.

File Date and Time – (Optional) 32 bit file date and time.

A subdirectory is created with the name specified in the filename parameter. This may be formatted as 8.3.

An optional parameter is used to specify a directory date and time. See the "32 Bit Values" column in Table 6.4 for format of the parameter value. The resolution for time in this command is 2 seconds. An example is given in the OPW Command description.

When the directory is created, the creation date/time, modified date/time and access date are set to the parameter value. If bits 23:16 of the file time value are zero, indicating an invalid month and day of month, the default directory time is used for the create date/time, modified date/time and access time.

Note : The default directory time is 0x31940000 (2004-12-04 12:00:00).

Error Codes:

Command Failed – The filename specified already exists or the maximum number of files allowed in the root directory has been reached.

Disk Full – There were not enough free allocation units on the disk to complete the write operation.

Filename Invalid – The name of the directory is invalid or contains disallowed characters.

File Open – A file is currently opened by a VMUSIC command. Stop playback on the VMUSIC before retrying this command.

6.2.6 Delete File (DLF)

Parameters:

Filename – Up to 11 characters ASCII.

The file specified in the filename parameter is deleted. The filename is normally specified as 8.3.

Error Codes:

Command Failed – The filename specified does not exist.

File Invalid – The filename specified is used in the current directory as either a subdirectory name, FAT volume label or other FAT structure.

Read Only – The read only attribute of the filename specified is set.

File Open – A file is currently open for writing or is being accessed by a VMUSIC command. Close the open file first before retrying this command.

6.2.7 Write To File (WRF)

Parameters:

Bytes to Write – 32 bits: number of bytes.

This command writes the specified number of bytes to the currently open file. If no file is open then the "File Open" error response is given. Once, a size has been specified then exactly that amount of data must be written to the Monitor port to be written to the file, there is no mechanism to break from the command early.

Data to write to the file is taken directly from the Monitor input and written to the file. There is no conversion from ASCII to binary in IPA mode nor is a carriage return required after writing the data to the Monitor port.

In this example, the command is given in ECS mode, the data size in ASCII mode and the data is represented in binary values.

```
WRF·10↵
01 02 03 04 05 06 07 08 09 0a
<prompt>
```

In SCS mode and IPH mode (values in Binary Mode are sent MSB first):

```
08 20 00 00 00 0A 0d
01 02 03 04 05 06 07 08 09 0a
<prompt>
```

The end of the file is always set to the file pointer after a write operation.

Error Codes:

Disk Full – There were not enough free allocation units on the disk to complete the write operation. This is reported after the command has completed. Data unable to be written to the disk is discarded.

File Invalid – There is no file currently open for writing. Use the OPW command to open a file before retrying the WRF command.

6.2.8 Open File for Write (OPW)

Parameters:

File Name – Up to 11 characters ASCII.

File Date and Time – (Optional) 32 bits: file date and time.

The File specified in filename is opened for writing. The filename is normally specified as 8.3.

Data is normally appended to the end of an existing file or a new empty file is created if it does not exist. The Seek (SEK) command can be used to move to an arbitrary point in the file and commence writing from there.

Note : The file will be truncated to the position of the file pointer when closed.

An optional parameter is used to specify a file date and time. See the "32 Bit Values" column in Table 6.4 for format of the parameter value. The resolution for time in this command is 2 seconds. The rules for when and how the file date and time value is applied are as follows:

Existing files:

Upon opening the file the directory table structure specific for that file is modified. If bits 23:16 of the parameter are zero, indicating an invalid month and day of month, then there is no change made to the modified date/time or access date of the opened file. Otherwise, the modified date/time and the access date of the file are updated using the file date/time passed as a parameter.

New files:

When the file is created, the creation date/time, modified date/time and access date are set to the parameter value. If bits 23:16 of the file time value are zero, indicating an invalid month and day of month, the default file time is used for the create date/time, modified date/time and access time.

Note : The Default file time is 0x31940000 (2004-12-04 12:00:00).

Files which have been opened for write must be closed to update the file size record in the directory table. If a file is not closed before the device is reset or removed, then the file size will not be modified and data lost.

For a file time of 7th June 2007 at 14:24:51 the following calculations are made:

Year (2007 – 1980 = 27) → 0x1B; Month (6) → 0x6; Day (7) → 0x7.

Hour (14) → 0xE; Minute (24) → 0x18; Seconds (51 / 2) → 0x19.

File time = (0x1B << 25) | (0x6 << 21) | (7 << 16) | (0xE << 11) | (0x18 << 5) | (0x19)

In binary (spaces between fields) = 0011011 0110 00111 01110 011000 11001b

Finally, convert binary to hexadecimal = 0x36C77319.

```
OPW·NEW.TXT 0x36C77319↵
<prompt>
```

Error Codes:

Command Failed –

File Invalid – The filename specified is used in the current directory is in use as either a subdirectory name, FAT volume label or other FAT structure.

Read Only – The read only attribute of the filename specified is set.

Disk Full – There are no free allocation units on the disk to create a file. The maximum number of files allowed in the root directory has been reached. This number is dependant on the file system.

File Open – Another file is currently open for writing or is being accessed by a VMUSIC command. Close the other file first and retry the OPW command.

Filename Invalid – The name of the directory is invalid or contains disallowed characters.

6.2.9 Close File (CLF)

Parameters:

Filename – Up to 11 characters ASCII.

Close the currently open file. The filename of the file must be specified and must match the filename used to open the file. The filename is normally specified as 8.3.

There is no need to close a file opened for read, however, for future compatibility it is highly recommended that these files be closed.

Error Codes:

Command Failed – The filename specified does not exist.

6.2.10 Read From File (RDF)

Parameters:

Bytes to Read – 32 bit: number of bytes.

Will read the specified number of bytes from the currently open file. Once, a size has been specified then that amount of data will be transferred to the Monitor port, there is no mechanism to break from the command early.

If the number of bytes to read exceeds the number of bytes in the file then the entire remaining contents of the file will be transferred and no error message displayed. For this reason a DIR command should be done to find out the size of the file beforehand.

Data read from the file is taken directly from the file to the Monitor. There is no conversion from binary to ASCII in IPA mode nor is a carriage return sent after the data on the Monitor port.

```
RDF·10↵
helloagain<prompt>
```

Error Codes:

Command Failed – The current offset (read position) in the file is at the end of the file. There is no more data available.

6.2.11 Rename File (REN)

Parameters:

Source Filename – Up to 11 characters ASCII.

Target Filename – Up to 11 characters ASCII.

The file or subdirectory specified by the filename in the first parameter is renamed to the name specified in the second parameter. The filenames are typically formatted as 8.3 and may refer to subdirectories as well as files. There must be a space character between the filenames.

Error Codes:

Command Failed – The filename specified does not exist.

File Open – Another file is currently open for writing or is being accessed by a VMUSIC command. Close the other file first and retry the REN command.

6.2.12 Open File for Read (OPR)

Parameters:

Filename – Up to 11 characters ASCII.

File Date – (Optional) 16 bits: file date.

The file specified in the filename parameter is opened for reading. The filename is typically formatted as 8.3. Data is read from the start of an existing file. No new file is created if the file name specified does not exist. The Seek (SEK) command can be used to move to an arbitrary point in the file and commence reading from there.

An optional parameter is used to specify the date when the file was accessed. Upon opening the file the directory table structure specific for that file is modified. Table 6.4 shows the bitfields used for 16 bit dates used in this parameter. If bits 7:0 of the file date value are zero, indicating an invalid month and day of month, then there is no change made to the directory table of the opened file. Otherwise, the last access time of the file is updated using the file date passed as a parameter.

There is no need to close a file opened for read, however, for future compatibility it is highly recommended that these files be closed.

Error Codes:

Command Failed – The filename specified does not exist.

File Invalid – The filename specified is used in the current directory as either a subdirectory name, FAT volume label or other FAT structure.

File Open – Another file is currently open for writing. Close the other file first and retry the OPR command.

Filename Invalid – The name of the directory is invalid or contains disallowed characters.

6.2.13 Seek (SEK)

Parameters:

Offset in File – 32 bit number of bytes.

This command moves the file pointer to the position specified in the offset parameter. Only absolute positions in the file may be specified, there is no way of performing relative seeks.

Error Codes:

Command Failed – The offset specified is beyond the end of the file.

6.2.14 Free Space (FS, FSE)

Parameters: **None.**

Display the free space remaining on the disk. If there is likely to be more than 4GB of space on the disk then the FSE command must be used. In this case the FS command will report the free space as 0xFFFFFFFF.

The first time either FS or FSE is executed it will scan the entire disk and calculate the free space available. This will take a period of time depending on the total size and the cluster size of the disk, along with the speed at which the disk operates. This variance means that a command can take as short a time as 1 second or as long as a minute to perform the check. The Monitor will not respond to commands during this process.

After the FS or FSE has been called the Monitor will keep track of file operations and report the free space available without requiring a scan of the entire disk.


```
FSE↵
$00·$02·$45·$31·$01·$00·↵
<prompt>
```

```
FS↵
$FF·$FF·$FF·$FF·↵
<prompt>
```

Error Codes: **None.**

6.2.15 Identify Disk Drive (IDD, IDDE)

Parameters: **None.**

The IDD command shows a summary of information about the connected BOMS device.

```
idd↵
↵
USB·VID·=·$05DC↵
USB·PID·=·$A560↵
Vendor·Id·=·LEXAR···↵
Product·Id·=·JD·FIREFLY······↵
Revision·Level·=·3000↵
I/F·=·SCSI↵
FAT32↵
Bytes/Sector·=·$0200↵
Bytes/Cluster·=·$001000↵
Capacity·=·$1EB7C000·Bytes↵
Free·Space·=·$1EB77000·Bytes↵
↵
<prompt>
```

Note : Blank lines are inserted before and after the data returned from the command.

"Product Id" is read from the device and will be a fixed length of 16 characters. Similarly "Vendor Id" is fixed at 8 characters.

The "Free Space" entry is not filled out until an FS or FSE command has been executed and the free space on the disk evaluated.

Error Codes: **None.**

6.2.16 Disk Volume Label (DVL)

Parameters: **None.**

This command obtains the Disk Volume Label from the disk. This is 11 characters and stored in the MBR section of the disk. FAT32 volume labels which are written to the root file directory are not supported.

Error Codes: **None.**

6.2.17 Disk Serial Number (DSN)

Parameters: **None.**

This command displays the 32-bit serial number of the disk. It is displayed in little-endian format, as it is stored on the disk.

Error Codes: **None.**

6.2.18 Directory File Time Command (DIRT)

Parameters:

Filename – Up to 11 characters ASCII.

This command will display the create date/time, modified date/time and access date of a file in the current directory.

Typically the filename will be formatted as 8.3.

Ten bytes of data will be returned, the first 4 are the created date/time, the next 2 are the access date and the last 4 are the modified date/time. All values are sent LSB first.

In ASCII input mode the output from the command with a valid filename specified is:

```
DIRT·file1.XXX↵
↵
FILE1.XXX·$xx·$xx·$xx·$xx·$xx·$xx·$xx·$xx·$xx·$xx·↵
<prompt>
```

Note the space after the last hexadecimal number before the carriage return.

In binary input mode the output from the command is:

```
DIRT·file1.XXX↵
↵
FILE1.XXX·cccccccccc↵
<prompt>
```

The time and date can be decoded using the bit field definitions in Table 6.4.

Error Codes:

Command Failed – The specified filename does not exist.

File Open – A file is currently opened by a VMUSIC command. Stop playback on the VMUSIC before retrying this command.

6.3   Power Management Commands

Extended Command Set	Short Command Set (Hexadecimal Codes)	Function
SUD↵	15 0D	Suspend disk
WKD↵	16 0D	Wake disk
SUM↵	17 0D	Suspend monitor

Table 6.5 Power Management Commands

6.3.1 Suspend Disk (SUD)

Parameters: **None.**

Puts the VNC1L into a mode where the disk is automatically suspended when it is not in use. It is woken up each time a monitor command is called. This only needs to be called once to enable the mode. Disable the mode by calling the Wake Disk command.

There is a large latency in accessing all disks in Suspend Disk Mode. If frequent or rapid access to data on a disk is required then the Wake Disk command should be called before the data is accessed.

Important: Due to the variability of the quality of BOMS devices (especially USB Flash Disks) it is not always possible to reliably suspend and restore disks.

It is highly recommended that the disk should not be suspended while a file is opened for write or read. It is also prudent to test and verify different makes and models disks before deployment.

Error Codes:

File Open – Another file is currently open for writing. Close the other file first and retry the SUD command.

6.3.2 Wake Disk (WKD)

Parameters: **None.**

This command turns off the automatic suspended mode which is enabled by the Suspend Disk command. It should be called before transferring large amounts of data to or from a disk.

Error Codes: **None.**

6.3.3 Suspend Monitor (SUM)

Parameters: **None.**

This command turns off all internal clocks and suspends both USB hosts. The Monitor is disabled, whether it is on UART, SPI or FIFO. The Monitor is started again by toggling the Ring Indicator (RI#) pin. A simple way to resume the Monitor when any input is sent to the device would be to tie Ring Indicator (RI#) and the UART receive data (RXD) pins together.

Error Codes: **None.**

6.4   Unused I/O Pin Commands

Extended Command Set	Short Command Set (Hexadecimal Codes)	Function
`IOR·byte↵`	`29 20 byte 0D`	Read I/O Port (1st parameter is port number)
`IOW·byte+byte+byte↵`	`2A 20 byte byte byte 0D`	Write I/O Port (1st parameter is port number, 2nd is direction, 3rd is value)

Table 6.6 Unused I/O Pin Commands

6.4.1 I/O Read (IOR)

Parameters:

Port Number – Byte: Encoded port number.

Reads the I/O Port specified in Table 6.7. Returns one byte representing the state of the pins on that port.

This example will read Port BD:

`IOR·$02↵`
`$55·↵`

```
<prompt>
```

Port Number	Byte 1
AD	0x00
AC	0x01
BD	0x02
BC	0x03 to 0xFF

Table 6.7 Port Number Encoding

Error Codes: **None.**

6.4.2 I/O Write (IOW)

Parameters:

Port Number – Byte: Encoded port number.

Direction – Byte: Bitmap of output pins.

Value – Byte: Bitmap of output pins to set high.

Writes to the I/O Port specified in Table 6.7. Direction is used to set pins to be output pins and the Value parameter specifies which are to be set high.

Set Port AC (0x01), bit 1 and bit 0 to outputs (0x03), with bit 1 high and bit 0 low (0x02):

`IOW·$010302`↵
```
<prompt>
```

Error Codes: **None.**

6.5   Printer Commands

Extended Command Set	Short Command Set (Hexadecimal Codes)	Function
PGS↵	*81 0D*	Get printer status
PSR↵	*82 0D*	Printer soft reset

Table 6.8 Printer Commands

6.5.1 Printer Get Status (PGS)

Parameters: **None.**

Retrieves the status of a printer. This is one byte and decoded using the information in Table 6.8.

Bit	Meaning
7:6	Always 0
5	Paper Empty
4	Selected
3	Not Error
0:2	Always 0

Table 6.9 Printer Status Encoding

Error Codes: **None.**

6.5.2 Printer Soft Reset (PSR)

Parameters: **None.**

Sends a soft reset command to a printer.

Error Codes: **None.**

6.6   USB Device Commands

The Vinculum VNC1L firmware implements a convention to allocate device numbers to interfaces on USB devices. There can be up to 16 devices accessible to the Monitor to allow for multiple interfaces presented by devices (such as the FT2232 chip).

The mapping of physical devices to device numbers used in the Monitor is dependent on other devices connected and the order in which these devices have been connected. It is therefore not safe to assume that a given device will always be allocated the same device number. The QD (Query Device) command is provided to verify device numbers.

Before sending or receiving data from a device the SC (Set Current) command must be used to specify the device interface to use.

Extended Command Set	Short Command Set (Hexadecimal Codes)	Function
QP1↵	2B 0D	Query port 1
QP2↵	2C 0D	Query port 2
QD·byte↵	85 20 byte 0D	Query device specified in the 1st parameter
SC·byte↵	86 20 byte 0D	Set device specified in the 1st parameter as the current device
DSD·byte↵ data	83 20 byte 0D data	Send data to USB device where the size of the data is specified in the 1st parameter
DRD↵	84 0D	Read back data from USB device
SSU·qword↵ data	9A 20 qword 0D	Send setup data to device control endpoint
SF·byte↵	87 20 byte 0D	Set device specified in the 1st parameter as an FTDI device
QSS↵	98 0D	Query Slave Status (only available on VDPS)

Table 6.10 USB Device Commands

Note : The QSS command can be used only on VDPS Vinculum VNC1L firmware.

6.6.1 Query Port (QP1, QP2)

Parameters: **None.**

The Query Port commands display two bytes of information about devices connected to USB Port 1 for QP1 or USB Port 2 for QP2. The first byte shows the class or classes of connected devices. The second byte is always zero.

The connected devices bitmask is described in Table 6.11. Multiple device classes may be connected to the port, for example, 0x8C 0x00 will be returned if a Printer Class and a HID Class device are connected to a Hub Device.

If the first byte is 0x40 then the device class has not been recognised by the Vinculum VNC1L firmware. A value of 0x00 0x00 indicates that there are no devices connected to the port.

Bit	Device Type
7	Hub Device on Port
6	Unknown Device
5	BOMS Class Device
4	CDC Class Device
3	HID Class Device
2	Printer Class Device
1	Reserved (always 0)
0	FTDI FT232 / FT245 / FT2232 Device

Table 6.11 Device Type Bit Definitions

Error Codes: **None.**

6.6.2 Query Device (QD)

Parameters:

Device Number – 8 bits

This command is used to find information on device interface. The device number can be a valid device in the range of 0 to 15. This will return a block of 32 bytes as shown in Table 6.12.

This information is used to identify the device type to allow an application to find a specific device connected to the Vinculum VNC1L.

Offset	Size (bytes)	Data
0	1	USB Address
1	1	Control End Point 0 Size
2	1	Pipe In End Point Number
3	1	Pipe In End Point Size
4	1	Pipe Out End Point Number
5	1	Pipe Out End Point Size
6	1	Data Toggles
7	1	Device Type: See Table 6.11
8	1	Reserved
9	1	Location: 1 – USB Port 1, 2 – USB Port 2
10	1	MI Index
11	1	Device Class
12	1	Device Sub Class
13	1	Device Protocol
14	2	VID
16	2	PID
18	2	BCD
20	1	Device Speed: 1 – Full Speed, 2 – Low Speed, 4 – Low Speed Device on Hub, 8 – Device on Low Speed Hub
21 - 31	10	Reserved

Table 6.12 USB Device Information Record

Definitions of USB class codes, USB VID, PID and BCD numbers can be found in the USB 2.0 Specification referenced in Table 1.2.

Error Codes:

Command Failed – The specified USB device is out of range.

6.6.3 Set Current (SC)

Parameters:

Device Number – 8 bits

This command is used to select an interface on a USB device for communication. All output from DSD (Device Send Data) and DRD (Device Read Data) commands are routed to this device. Valid values for Device Number are 0 to 15. In VDPS firmware the Device Number can be replaced by the ASCII letter 'S' to indicate the Slave Device.

Error Codes:

Command Failed – The specified USB device is out of range.

6.6.4 Device Send Data (DSD)

Parameters:

Bytes to Write – 8 bits: number of bytes.

The DSD command sends a packet of data to a USB device. The target device must be set as the current device (refer to the SC command).

The data to be sent to the device is taken directly from the Monitor input. There is no conversion from ASCII to binary in IPA mode nor is a carriage return required after writing the data to the Monitor port. USB byte ordering applies to the data received.

Up to 128 bytes may be transferred in a single packet. If, however, the OUT end point of the device accepts a maximum size less than this then that will be the maximum number of bytes which can be transferred. The size of the OUT end point can be found using the QD command output in Table 6.12.

In this example, the command is given in ECS mode, the data size in ASCII mode and the data is represented in binary values.

```
DSD·10↵
01 02 03 04 05 06 07 08 09 0a
<prompt>
```

In SCS mode and IPH mode:

```
83 20 0a 0d
01 02 03 04 05 06 07 08 09 0a
<prompt>
```

Error Codes:

Command Failed – The currently selected USB device cannot be written to or does not exist. The number of bytes specified to write is zero, greater than 128 or greater than the end point maximum size. The device does not have an OUT Endpoint.

6.6.5 Device Read Data (DRD)

Parameters: **None.**

The DRD command reads pending data from a USB device. The first byte returned is the size of the data, followed by a carriage return character. In IPA mode, this is shown as an ASCII representation of the number, in IPH (binary) mode the first byte received is the size of the data. If no bytes are available the size will be zero.

The target device must be set as the current device (refer to the SC command).

If more than zero bytes are to be received from the device the Monitor will display the data. There is no conversion of the data from binary to ASCII in IPA mode nor is a carriage return sent after receiving data on the Monitor. USB byte ordering applies to the data received.

In ASCII mode:

drd↵
$04.↵
00 FF 00 00 <prompt>

In binary mode:

84 0D
04 0D
00 FF 00 00 <prompt>

Error Codes: **None.**

6.6.6 Device Send Setup Data (SSU)

Parameters:

Packet to Write – 8 bytes: Setup Packet.

The SSU command sends a packet of data to a USB device control endpoint with a setup command. The target device must be set as the current device (refer to the SC command).

Section 9.3 of the USB Specification describes the format of a Setup Packet which is sent to a device control endpoint. Every Setup Packet is 8 bytes, however, a follow-on data stage can be specified which can send either Data Out to the device or receive Data In from the device.

Offset	Size (bytes)	Field	Description
0	1	bmRequestType	Bitmap describing the characteristics of the Setup Packet
1	1	bRequest	Specific request Refer to USB Specification Section 9.4
2	2	wValue	Dependant on request
4	2	wIndex	Dependant on request
6	2	wLength	Number of bytes to transfer if there is a data stage to follow

Table 6.13 SSU Command Setup Packet Format

Important: The byte order for multi-byte data (wValue, wIndex and wLength values) in Table 6.13 is LSB-first. This is the USB protocol byte order rather than the Monitor command input byte order.

Input or output for the data stage is taken directly from or sent directly to the Monitor input. There is no conversion from ASCII to binary in IPA mode. No carriage return is required after writing the data to the Monitor nor is one sent after receiving data on the Monitor. USB byte ordering applies to the data stage.

A maximum of 128 bytes may be sent or received during the data stage.

This example sends a Setup Packet requesting a data stage of 8 more output bytes (wLength = 8). The command is given in ECS mode, the 8 setup command bytes are given in ASCII mode and the data stage output is represented in binary values.

```
SSU·$2109000201000800↵
01 02 03 04 05 06 07 08
<prompt>
```

In SCS mode and IPH mode:

```
9A 20 21 09 00 02 01 00 08 00 0D
01 02 03 04 05 06 07 08
<prompt>
```

This is a further example where the data stage is requesting a number of input bytes. The command is given in ECS mode, the 8 setup command bytes are given in ASCII mode and the data stage input is represented in binary values.

```
SSU·$A109000201000800↵
$08 $00↵
01 02 03 04 05 06 07 08 <prompt>
```

In SCS mode and IPH mode:

```
9A 20 A1 09 00 02 01 00 08 00 0D
08 00 0D
01 02 03 04 05 06 07 08 <prompt>
```

Error Codes:

Command Failed – The currently selected USB device cannot be written to or does not exist. The number of bytes specified to write is greater than 128.

6.6.7 Set FTDI (SF)

Parameters:

Device Number – 8 bits

This is used to force the Monitor to treat a device which has not got an FTDI VID and PID as if it were an FTDI device. It is particularly useful if the VID and PID on an FT232, FT245 or FT2232 have been programmed to custom values. Valid values for Device Number are 0 to 15.

Error Codes:

Command Failed – The specified USB device is out of range.

6.6.8 Query Slave Status (QSS)

Parameters: **None.**

This is used only in VDPS firmware to query the status of the slave device of the Vinculum VNC1L. It reports three bytes: the first is the current status of the slave as per Table 6.14; the second is the number of bytes of data awaiting transmission to the host; the third is the number of bytes received from the host which are waiting to be read by the Monitor.

1st Byte	Meaning
7:4	Reserved
3	Data available from slave (RX)
2	Data available to slave (TX)
1	Suspended
0	Connected

Table 6.14 Query Slave Status Values

Error Codes: **None.**

6.7   Commands to FT232/FT245/FT2232 on Port 1 or Port 2

If an FTDI FT232, FT245 or FT2232 is connected to USB Port 1 or Port 2 then the commands in Table 6.15 may be used to modify the settings of the device. The commands are analagous to the FTDI API Functions in the FTDI D2XX Programmer's Guide (reference in Table 1.2).

The interface device must first be selected using the USB Device Command SC (Set Current) before utilising these commands.

Extended Command Set	Short Command Set (Hexadecimal Codes)	Function
FBD·*divisor*↵	*18 20 divisor 0D*	Set baud rate
FMC·*word*↵	*19 20 word 0D*	Set modem control
FSD·*word*↵	*1A 20 word 0D*	Set data characteristics
FFC·*byte*↵	*1B 20 byte 0D*	Set flow control
FGM↵	*1C 0D*	Get modem status
FSL·*byte*↵	*22 20 byte 0D*	Set latency timer
FSB·*word*↵	*23 20 word 0D*	Set bit mode
FGB↵	*24 0D*	Get bit mode

Table 6.15 FT232/FT245/FT2232 Commands

6.7.1 Set Baud Rate (FBD)

Parameters:

Baud Rate – 3 Bytes: Encoded baud rate.

Sets the baud rate on an FT232, FT245 or FT2232 connected to Port 1 or Port 2.

The encoded baud rate is represented as 3 bytes from Table 6.2. In binary input mode the three bytes are specified in the order which they should be used in the commands. In ASCII mode byte 1 is the MSB of the numerical value, e.g. 9600 baud would be $00384100 when entered in the SBD command.

Error Codes: **None.**

6.7.2 Set Modem Control (FMC)

Parameters:

Modem Control – 2 Bytes: Encoded modem control.

Sets the modem control bytes on an FT232, FT245 or FT2232 connected to Port 1 or Port 2.

The encoded modem control bytes are shown in Table 6.16. DTR# and RTS# may be individually changed or left by utilising the mask in the second byte.

1st Byte	Operation
0	DTR# State 0 = off, 1 = on
1	RTS# State 0 = off, 1 = on
7:2	Reserved '0'

2nd Byte	Operation
0	1 = change DTR, 0 = leave DTR alone
1	1 = change RTS, 0 = leave RTS alone
7:2	Reserved '0'

Table 6.16 FT232/FT245/FT2232 Modem Control Bytes

Error Codes: **None.**

6.7.3 Set Data Characteristics (FSD)

Parameters:

Data Char – 2 Bytes: Encoded data characteristics.

Sets the data characteristics on an FT232, FT245 or FT2232 connected to Port 1 or Port 2.

In Table 6.17 the two encoded bytes are shown to allow changing the number of data bits, parity, number of stop bits and break.

1st Byte	Operation
7:0	Number of Data bits: 7 – 7 data bits 8 – 8 data bits

2nd Byte	Operation
2:0	Parity : 0 – none 1 – odd 2 – even 3 – mark 4 - space
5:3	Number of Stop bits : 0 - 1 stop bit 1 - 1 stop bit 2 - 2 stop bits
6	1 = Send break 0 = Stop break
7	Reserved '0'

Table 6.17 FT232/FT245/FT2232 Data Characteristics Bytes

Error Codes: **None.**

6.7.4 Set Flow Control (FFC)

Parameters:

Flow Control – Byte: Encoded flow control.

Sets the flow control mode on an FT232, FT245 or FT2232 connected to Port 1 or Port 2.

Table 6.18 lists the flow control options available.

Byte	Operation
0	Hardware handshake RTS / CTS
1	Hardware handshake DTR / DSR
2	Software handshake XOFF / XOFF
7:3	Reserved '0'

Table 6.18 FT232/FT245/FT2232 Flow Control Bytes

Error Codes: **None.**

6.7.5 Get Modem Status (FGM)

Parameters: **None.**

Gets the modem status from an FT232, FT245 or FT2232 connected to Port 1 or Port 2.

Two bytes are returned. Table 6.16 is used to decode the bits to determine the status of DTR# and RTS#.

Error Codes: **None.**

6.7.6 Set Latency Timer (FSL)

Parameters:

Latency – Byte: Latency value.

Sets the latency timer on an FT232, FT245 or FT2232 connected to Port 1 or Port 2.

One byte is supplied which specifies the latency timer value in milliseconds. The default is 16ms.

Error Codes: **None.**

6.7.7 Get Bit Mode (FGB)

Parameters: **None.**

Retrieves the state of the bit mode pins from an FT232, FT245 or FT2232 connected to Port 1 or Port 2.

This example reads the bit mode:

```
FGB·$02↵
$50·↵
<prompt>
```

One byte is returned with the state.

Error Codes: **None.**

6.7.8 Set Bit Mode (FSB)

Parameters:

Bit Mask – Byte: Mask of bits to change.

Bit Enable – Byte: Value of bits to change.

Sends the SetBitMode command to an FT232, FT245 or FT2232 connected to Port 1 or Port 2.

This example sets the bit mode bit 4 to disabled:

```
FSB·$1000↵
<prompt>
```

Error Codes: **None.**

6.8   VMUSIC Commands

On VMSC firmware the following extension commands are available to interface with an MP3 decoder.

To pause playback, send either the 'E' or 'e' character to the monitor without sending the carriage return character. To resume playback, send the carriage return character.

The Vinculum VNC1L firmware supports MP3 files and unlocked WMA files up to 320 kbit/sec. This limit is dependent on the speed and characteristics of the disk used to store the data files.

Extended Command Set	Short Command Set (Hexadecimal Codes)	Function
VPF·*file*↵	*1D 20 file 0D*	Plays a single file
VRF·*file*↵	*89 20 file 0D*	Repeatedly plays a single file
VST↵	*20 0D*	Stops playback
V3A↵	*21 0D*	Plays all MP3 files
VRA↵	*8A 0D*	Repeatedly plays all MP3 files
VRR↵	*8F 0D*	Repeatedly plays random MP3 files
VSF↵	*25 0D*	Skip forward one track
VSB↵	*26 0D*	Skip back one track
VSD↵	*8E 0D*	Skip forward one whole directory
VP↵	*8B 0D*	Pause playback
VF↵	*8C 0D*	Fast forward 5 seconds
VB↵	*8D 0D*	Rewind 5 seconds
VRD·*byte*↵	*1F 20 byte 0D*	Reads command register
VWR·*byte+word*↵	*1E 20 byte word 0D*	Writes command register
VSV·*byte*↵	*88 20 byte 0D*	Sets playback volume

Table 6.19 VMusic Commands

The status information in Table 6.20 is displayed on the Monitor during playback. The "Playing" message is shown at the start of each track and "Stopped" when the last track has completed or the VST command issued. If ID3 information is available from the track then this is displayed in 5 lines immediately following the "Playing" message. If an ID3 tag cannot be decoded then a blank line is displayed. Each line of ID3 data may be up to 256 bytes long, including the terminating carriage return.

Extended Command Set	Short Command Set	Reason
Playing·*file*↵	P·*file*↵ *(50 20 file 0D)*	Track started
track number↵ *title*↵ *artist*↵ *album*↵ *composer*↵	*track number*↵ *title*↵ *artist*↵ *album*↵ *composer*↵	ID3 information
T *time*↵	T↵ *(53 time 0D)*	Elapsed time
Stopped↵	S↵ *(52 0D)*	Playback stopped

Table 6.20 VMusic Track Information

While tracks are being played the elapsed time, in seconds, is displayed. The time is a 16-bit value and the display format depends on the Numerical Mode selected.

6.8.1 Play File (VPF and VRF)

Parameters:

Filename – Up to 11 characters ASCII.

Send the specified file to the VLSI VS1003 MP3 decoder. This example shows playback of a file "ST1.MP3" in Extended Command Set and IPA Mode:

```
VPF·ST1.MP3↵
Playing·ST1.MP3↵
1/10↵
Song One↵
Blurb↵
Best of Blurb↵
Damian Allbran↵
T $01·$00·↵
T $02·$00·↵
T $03·$00·↵
Stopped↵
```

If the VRF command is used then the file will play repeatedly until a VST command is issued.

Error Codes:

Command Failed – The specified filename does not exist.

6.8.2 Stop (VST)

Parameters: **None.**

Stop playback of current file. Stop all tracks if playing multiple files using the Play All Tracks (V3A or VRA) command.

Error Codes: **None.**

6.8.3 Play All Tracks (V3A ,VRA and VRR)

Parameters: **None.**

Play all files in the current directory and all sub-directories. Files which do not have the extension ".MP3" or ".WMA" are ignored. Tracks are played sequentially in the order in which they are physically stored on the disk.

This example shows playback of all files in a directory using the Extended Command Set and IPH Mode. The example files are ST2.MP3 which is 4 seconds long and ST1.MP3 which is 3 seconds long.

```
V3A↵
Playing·ST2.MP3↵
2/10↵
Song Two↵
Blurb↵
The Very Best of Blurb↵
Graeme Boxon↵
53 20 01 00 0D
53 20 02 00 0D
53 20 03 00 0D
53 20 04 00 0D
Playing·ST1.MP3↵
1/10↵
Song One↵
Blurb↵
Best of Blurb↵
Damian Allbran↵
53 20 01 00 0D
53 20 02 00 0D
53 20 03 00 0D
Stopped↵
```

If VRA is called then all files are repeated when the files in current directory and sub-directories are completed.

The VRR command selects a new file at random after the current file has completed playing. It does not shuffle the order in which files are played.

Error Codes: **None.**

6.8.4 Skip Back and Forward (VSF and VSB)

Parameters: **None.**

Skip to the next or previous track if playing files using the Play All Tracks (V3A or VRA) commands. Skips to another random track if using the VRR command.

Error Codes: **None.**

6.8.5 Skip Forward One Directory (VSD)

Parameters: **None.**

Skip to the next directory track if playing files using the Play All Tracks (V3A or VRA) commands.

Error Codes: **None.**

6.8.6 Pause (VP)

Parameters: **None.**

Pauses playback of current file. If other VMusic playback commands are called during pause then the selected track starts playing regardless of whether the device was previously in the paused state.

Error Codes: **None.**

6.8.7 Fast Forward or Rewind (VF and VB)

Parameters: **None.**

Skip forwards or back 5 seconds through the currently playing file.

Error Codes: **None.**

6.8.8 Write Command Register (VWR)

Parameters:

Register Number – 8 bits

Register Value – 16 bits

Writes a value to the command registers on the VLSI VS1003 MP3 decoder. This can be used for setting the playback volume. The VLSI VS1003 is referenced in Table 1.2.

Error Codes: **None.**

6.8.9 Read Command Register (VRD)

Parameters:

Register Number – 8 bits

Reads from the command register on the VLSI VS1003. The VLSI VS1003 is referenced in Table 1.2.

Error Codes: **None.**

6.8.10 Set Volume (VSV)

Parameters:

Volume – 8 bits

Sets the volume for both left and right channels in the VLSI VS1003. A volume of 0x00 is maximum volume, 0xFE is minimum volume. The VLSI VS1003 is referenced in Table 1.2.

Error Codes: **None.**

6.8.11 Pause Playback (VP)

Parameters: **None.**

Pauses the current file if a file is currently playing or resumes playback if the playback is currently paused. If a VSF or VSB command is issued during a pause then the next or previous file will start playing. The prompt is always displayted

after a VP command although track information (playing time and file names playing) may appear between the time the VP command is sent to the VNC1L and the prompt appearing.

Error Codes: **None.**

6.9   Debug Commands

Extended Command Set	Short Command Set (Hexadecimal Codes)	Function
`SD·dword`↵	`03 20 dword 0D`	Sector dump
`SW·dword`↵ `data`	`92 20 dword 0D data`	Sector write
`FWU·file`↵	`95 20 file 0D`	Upgrade firmware from named file on disk

Table 6.21 Debug Commands

6.9.1 Sector Dump (SD)

Parameters:

Sector Number – 32bits

Dumps raw binary of sector number specified. Data length is always 512 bytes and is not converted to ASCII when in IPA Mode.

Error Codes: **None.**

6.9.2 Sector Write (SW)

Parameters:

Sector Number – 32bits

Writes raw binary from the Monitor to sector number specified. Data length is always 512 bytes and is not converted from ASCII when in IPA Mode.

Important: Use with care as disk and file system corruption can occur.

Error Codes: **None.**

6.9.3 Firmware Upgrade (FWU)

Parameters:

Filename – Up to 11 characters ASCII.

Forces a firmware upgrade from named file on disk. Only FTD reflash files are supported – do not use with the ROM files used by the VPROG application.

A device reboot will occur once the upgrade has been performed.

Important: Use with care as firmware corruption can occur if an invalid upgrade file is used. Only use FTD files supplied by FTDI.

Error Codes:

No Upgrade – The specified filename does not exist or is identical to the firmware currently in Flash.

7.0   LED Function

The following section explains LED functions for the Vinculum VNC1L VDAP and VMSC firmware for various modules.

7.1   VDIP1 and VDIP2 Modules

The VDIP1 module has 3 LEDs: LED1 – Green; LED2 – Green; LED3 – Yellow. On the VDIP2 module there are just 2: LED1 – Green; LED2 – Green. The behaviour of the LEDs is described in Table 7.1.

Operation	LED Behaviour
Power On	LED1 and LED2 flash alternately for 2 seconds. Repeated until monitor connects.
USB Disk Initialisation	LED1 on, LED2 off
USB Disk Ready	LED1 off, LED2 on
USB Disk Removed	LED1 off, LED2 off
Commands from monitor Port to USB Disk	LED1 off, LED2 flashes
Commands from monitor Port with USB Disk removed	LED1 off, LED2 off

Table 7.1 VDIP1 and VDIP2 LED Functions

7.2   VDRIVE1 and VDRIVE2 Modules

The VDRIVE1 and VDRIVE2 modules have 1 LED bi-colour Green/Red LED. The behaviour of the LED is described in Table 7.2.

Operation	LED Behaviour
Power On	LED flashes Green/Red alternately for 2 seconds. Repeated until monitor connects.
USB Disk Initialisation	LED Red
USB Disk Ready	LED Green
USB Disk Removed	LED off
Commands from monitor Port to USB Disk	LED flashes Green
Commands from monitor Port with USB Disk removed	LED off

Table 7.2 VDRIVE1 and VDRIVE2 LED Functions

7.3   VMUSIC1 and VMUSIC2 Modules

The VMUSIC1 and VMUSIC2 modules have 1 LED bi-colour Green/Red LED. The behaviour of the LED is described in Table 7.3.

Operation	LED Behaviour
Power On	LED flashes Green/Red alternately for 2 seconds. Repeated until monitor connects.
USB Disk Initialisation	LED Red
USB Disk Ready	LED Green
USB Disk Removed	LED off
Commands from monitor Port to USB Disk	LED flashes Green
Commands from monitor Port with USB Disk removed	LED off
Music Playing	LED flashes Green
Music Stopped/Paused	LED Green

Table 7.3 VMUSIC1 and VMUSIC2 LED Functions

8.0 Examples

This section gives example configuration and commands for commonly used Vinculum VNC1L applications.

8.1 Sending Data to USB Device (Data Mode)

In this example an FT2232 dual chip device is connected to USB Port 1. VDAP firmware is used.

The output from the QD (Query Device) command is as follows.

```
qd·0↵
$01·$08·$81·$40·$02·$40·$00·$01·$01·$01·$00·$FF·$FF·$FF·$03·
$04·$10·$60·$00·$05·$01·$00·$00·$00·$00·$00·$00·$00·$00·$00·
$00·$00·↵
D:\>↵
qd·1↵
$01·$08·$83·$40·$04·$40·$00·$01·$01·$01·$01·$FF·$FF·$FF·$03·
$04·$10·$60·$00·$05·$01·$00·$00·$00·$00·$00·$00·$00·$00·$00·
$00·$00·↵
D:\>↵
```

The two interfaces share the same address but have a different interface number at byte 11 and different endpoint numbers. The first device is port A on a FT2232 chip and the second one is port B. It is important that before attempting to talk to a device that it is set to be the current device with the SC (Set Current) command.

To select device 0 or port A of the FT2232 chip:

```
sc·0↵
D:\>↵
```

At this point communication is open between the Monitor and the endpoint of the selected device. FTDI setup commands (or data) can be sent to the device.

```
fbd·$384100↵
D:\>↵
fmc·$0303↵
D:\>↵
ffc·$01↵
D:\>↵
```

The commands set baud rate to 9600, RTS/DTR active and RTS/CTS flow control respectively. Data mode can be entered by setting DATAREQ# active (low) and waiting for DATAACK# (low).
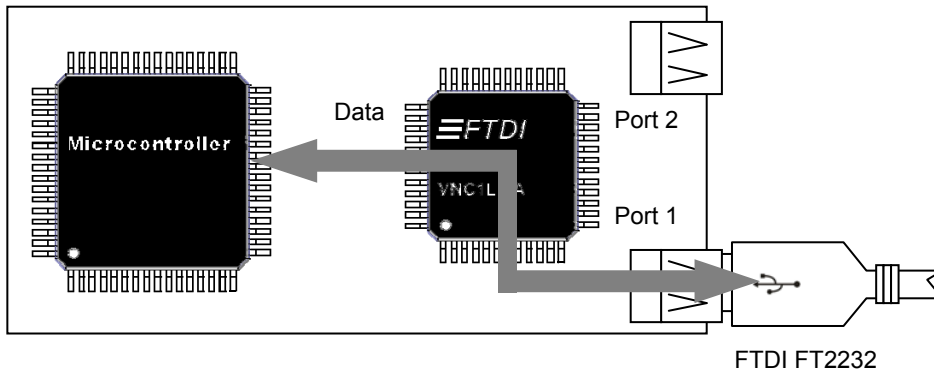
Figure 8.1 Vinculum VNC1L Connected to USB Device in Data Mode

Once the acknowledge (DATAACK# low) has been received all data sent to the Monitor will go directly to port A of the FT2232 chip and all data received by the port A of the FT2232 chip will be echoed to the Monitor. This is illustrated in Figure 8.1.

8.2 Sending Data to USB Device (Command Mode)

In this example we connect a Vinculum VNC1L running VDAP firmware to a USB device using command mode only, as illustrated in Figure 8.2.
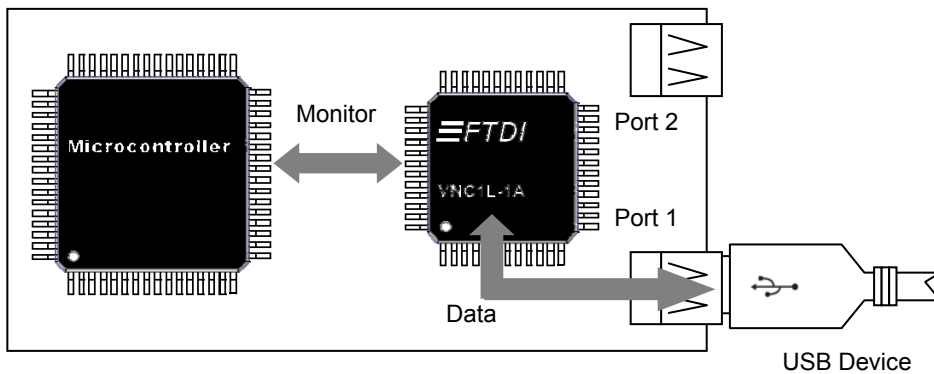

Figure 8.2 Vinculum VNC1L Connected to USB Device in Command Mode

To send data to the device, firstly, we need to setup communications with the printer, using the SC (Set Current) command, then utilise the DSD (Device Send Data) command to send a string of text. The DSD command can only send a maximum of 128 characters at a time.

The example below is to send a short string to USB Printer which is able to receive standard printable characters sent from the USB Port. A suitable printer to use is an HP DeskJet 460.

```
Device·Detected·P1↵
ipa↵
D:\>↵
qp1↵
$24·$00·↵
D:\>↵
```

```
qd·0↵
$01·$08·$81·$40·$01·$40·$03·$04·$01·$01·$00·$07·$01·$02·$F0·
$03·$12·$13·$00·$01·$01·$00·$00·$00·$00·$00·$00·$00·$00·$00·
$00·$00·↵
D:\>↵
sc·0↵
D:\>↵
dsd·9↵
hello·2u↵
↵
D:\>↵
```

Reading data from a device is performed in much the same way. After the device has been sent as the current device the DRD (Device Read Data) command can read pending data from the device and display that on the monitor. In this example a USB mouse is read:

```
D:\>↵
drd↵
$04·↵
00 FF 00 00
D:\>↵
```

Note : The Monitor returned the size of the data read as the first byte. Since this example is in IPA mode, the size is given in ASCII text and the data which follows is in binary format (indicated by *italics* in the example).

8.3   VDPS to PC Host

In VDPS firmware, USB Port 1 is a USB Slave port. USB Port 2 is still configured as a USB Host port and can be used as normal from the Monitor.

When USB Port 1 is connected to a host PC it will enumerate as if it were an FT232B device and the operating system can load the FTDI drivers accordingly. You can select the slave port to be the current device and go into data mode using DATAREQ# and DATAACK#.
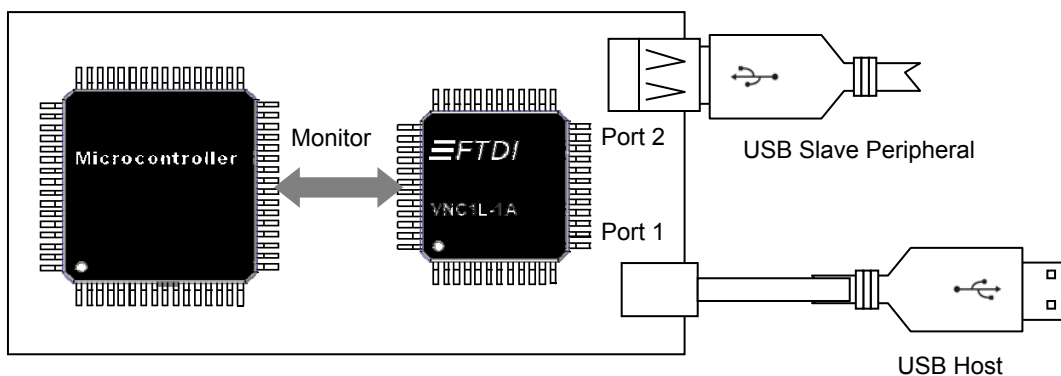

Figure 8.3 Vinculum VNC1L Connected to Host and a Slave Device

When the device is connected to a host PC and the drivers have been loaded you will receive a message:

```
Slave·Enabled↵
```

When the device is disconnected you will receive a message:

```
Slave·Disabled↵
```

You can check the status of the slave connection at any time with the command QSS (Query Slave Status). This will return 3 bytes of data, for example:

```
Slave·Enabled↵
IPA↵
D:\>↵
QSS↵
$01·$00·$00·↵
D:\>↵
```

To access the slave port you have to use the data mode via DATAREQ# and DATAACK# lines. First you should select the slave device with the SC command and the parameter 'S' to indicate the slave. For example:

```
SC·S↵
D:\>↵
```

Then activate DATAREQ# and wait for DATAACK#.

At this stage anything sent to the Vinculum VNC1L Monitor will be passed on to the host PC. Similarly, any data sent by the host PC will be passed to the Vinculum VNC1L Monitor without the Vinculum VNC1L interpreting the data.

An example of demonstrating this would be to open a terminal emulation program to talk to the Vinculum host PC port and send data to the Vinculum VNC1L Monitor port.

To come out of data mode, drop the DATAREQ# line. The peripheral on USB Port 2 can now be used using commands from the Monitor.

The DSD and DRD commands cannot be used to send and receive data from the slave device.

9.0 Troubleshooting

The following section should be consulted if you are having problems with the Vinculum VNC1L firmware. Please consult the FAQ section on the Vinculum Website.

9.1 Monitor Port

9.1.1 Cannot Connect to Monitor Port

After the Power On sequence if the LEDs continue to flash then there is no monitor connected to the Vinculum VNC1L. The Monitor is required for using the Vinculum VNC1L firmware.

If the Monitor is on a Serial UART then check that RTS / CTS is connected and the terminal application or microcontroller is correctly using RTS / CTS for flow control.

If the LEDs stop flashing but no prompt is received from the Monitor port: on Serial UART check the baud rate, number of data bytes, stop bits and parity correspond to the expected values; on SPI and FIFO modes this can be caused by incorrect connections to the VNC1L or if there is no clock signal supplied to the SPI port on the VNC1L.

9.1.2 No Prompt after USB Disk Detected

If the Vinculum VNC1L reports "Device Detected P2" then fails to send a prompt when a USB disk is connected then the disk may not be supported by Vinculum VNC1L. Certain types and models of USB disks fail to initialise properly, for instance, devices which are or emulate USB CDROM drives or USB floppy disks, devices which do not support the SCSI or AT command sets.

If the Vinculum VNC1L identifies that a USB disk does not have a sector size of 512 bytes then it will not allow access to the FAT file system on the disk.

9.1.3 Device Unresponsive after FS or FSE Commands

When performing an FS or FSE command on a USB disk from the Monitor for the first time after a disk has been detected, it may take several seconds for the command to complete. In this time the firmware is calculating the free space available on the disk. The time taken depends on the amount of size of the disk, the speed at which the disk accesses data and the geometry of the disk. In some cases this can take up to 60 seconds for large disks with small cluster sizes but most disks will take around 5 to 10 seconds.

9.1.4 Device Unresponsive after Disk Operation

When reading or writing to a USB disk, the size of the data to be transferred must be known and sent as part of the command which initiates the read or write. There is no mechanism for abandoning the operation once it is started.

If insufficient bytes have been sent or received the device will appear unresponsive. On the VDRIVE, VDIP and VMUSIC modules, one LED will flash continually until the required number of bytes has been transferred.

9.2   USB Devices

9.2.1 Low Speed Devices

USB 1.1 Low Speed Devices connected directly to the Vinculum VNC1L do not receive a Start of Frame (SOF) packet. A Keep Alive state is asserted once every millisecond. This should not be an issue for successful operation of most Low Speed Devices. However, problems may be encountered if Low Speed Devices are present on USB Hubs.

9.2.2 USB Disk Support

Only USB disks which support the SCSI or AT command sets will operate on the Vinculum. These devices can be identified by their USB Class, Subclass and Protocol values of 0x08, 0x06 and 0x50 respectively.

Disks which do not have a sector size of 512 bytes are not supported.

10.0 Contact Information

Head Office - Glasgow, UK

Future Technology Devices International Limited
373 Scotland Street
Glasgow G5 8QB
United Kingdom
Tel: +44 (0) 141 429 2777
Fax: +44 (0) 141 429 2758
E-Mail (Sales): vinculum.sales@ftdichip.com
E-Mail (Support): vinculum.support@ftdichip.com
E-Mail (General Enquiries): admin1@ftdichip.com
Web Site URL: http://www.vinculum.com
Web Shop URL: http://apple.clickandbuild.com/cnb/shop/ftdichip

Branch Office - Taiwan

Future Technology Devices International Limited (Taiwan)
4F, No 16-1, Sec. 6 Mincyuan East Road
Neihu District
Taipei 114
Taiwan, R.O.C.
Tel: +886 2 8791 3570
Fax: +886 2 8791 3576
E-Mail (Sales): tw.sales1@ftdichip.com
E-Mail (Support): tw.support1@ftdichip.com
E-Mail (General Enquiries): tw.admin1@ftdichip.com
Web Site URL: http://www.vinculum.com

Branch Office - Hillsboro, Oregon, USA

Future Technology Devices International Limited (USA)
7235 NW Evergreen Parkway, Suite 600
Hillsboro, OR 97124-5803
USA
Tel: +1 (503) 547-0988
Fax: +1 (503) 547-0987
E-Mail (Sales): us.sales@ftdichip.com
E-Mail (Support): us.support@ftdichip.com
E-Mail (General Enquiries): us.admin@ftdichip.com
Web Site URL: http://www.vinculum.com

Distributors and Sales Representatives

Please visit the Sales Network page of the FTDI Web site for the contact details of our distributor(s) in your country.