DLP-USB245M USB to FIFO Parallel Interface Module

The DLP-USB245M is the 2nd generation of DLP Design's USB adapter. This device adds extra functionality to its DLP-USB1 predecessor with a reduced component count and a new low price.

DLP-USB245M FEATURES

- Send/receive data over USB at up to 1 MBytes/sec
- 384-byte FIFO transmit buffer/128-byte FIFO receive buffer for high data throughput
- Simple interface to CPU or MCU bus

- No in-depth knowledge of USB required as all USB protocol is handled automatically within the module

- FTDI's virtual COM port drivers eliminate the need for USB driver development in most cases
- Integrated 3.3v regulator--no external regulator required
- Integrated power-on-reset circuit
- Integrated 6MHz–48Mhz clock multiplier PLL

- USB bulk or isocronous data-transfer modes
- 4.4v-5.25v single supply taken directly from the USB port
- UHCI/OHCI/EHCI host controller compatible
- USB 1.1 and USB 2.0 compatible
- USB VID, PID, serial number and product description
- Strings stored in on-board EEPROM
- EEPROM programmable on board via USB
- Virtual com port (VCP) drivers for:
    - Windows 98 and Windows 98 SE
    - Windows 2000/ME/XP
    - Windows CE**
    - MAC OS-8 and OS-9
    - MAC OS-X**
    - Linux 2.40 and greater
- D2XX (USB direct drivers + DLL S/W interface)
    - Windows 98 and Windows 98 SE
    - Windows 2000/ME/XP

** In planning or under development.

APPLICATION AREAS

- Prototype development
- USB ISDN and ADSL modems
- High-speed USB PDA interface
- USB interface for digital cameras
- USB interface for MP3 players
- High-speed USB instrumentation
- USB smart-card readers
- Set top box (STB) PC-USB interface
- USB hardware modems
- USB wireless modems
- USB bar code readers

GENERAL DESCRIPTION

The DLP-USB245M provides an easy, cost-effective method of transferring data to/from a peripheral and a host at up to 8 million bits (1 megabyte) per second. Its simple, FIFO-like design makes it easy to interface to any microcontroller or microprocessor via IO ports.

To send data from the peripheral to the host computer, simply write the byte-wide data into the module when TXE# is low. If the (384-byte) transmit buffer fills up or is busy storing the previously written byte, the device takes TXE# high in order to stop further data from being written until some of the FIFO data has been transferred over USB to the host.

When the host sends data to the peripheral over USB, the device will take RXF# low to let the peripheral know that at least one byte of data is available. The peripheral then reads the data until RXF# goes high indicating that no more data is available to read.

By using FTDI's virtual COM port drivers, the peripheral looks like a standard COM port to the application software. Commands to set the baud rate are ignored--the device always transfers data at its fastest rate regardless of the application's baud-rate setting. The latest versions of the drivers are available for download from DLP Design's website at http://www.dlpdesign.com.

APPLICATION NOTES

USB devices transfer data in packets. If data is to be sent from the PC, a packet is built up by the application program and is sent via the device driver to the USB scheduler. This scheduler puts a request onto the list of tasks for the USB host controller to perform. This will typically take at least 1 millisecond to execute because it will not pick up the new request until the next USB frame (the frame period is 1 millisecond).

There is, therefore, sizeable overhead (depending on your required throughput) associated with moving the data from the application to the USB device. If data is sent byte-at-a-time by an application, this will severely limit the overall throughput of the system.

It must be stressed that in order to achieve maximum throughput, application programs should send or receive data using buffers and not individual characters.
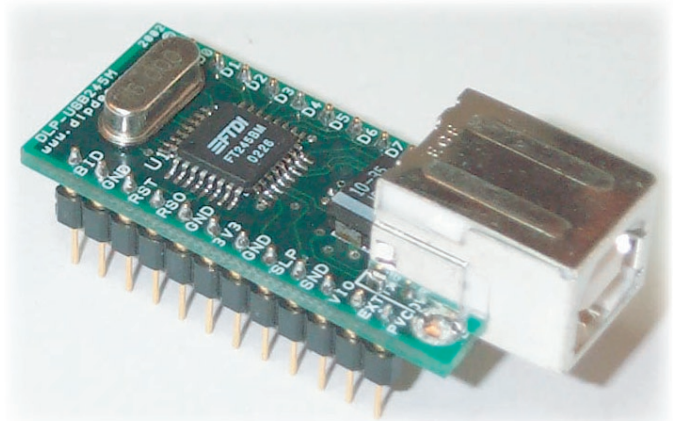
DRIVER SOFTWARE

FTDI's VCP (virtual COM port) driver-executable files are provided royalty free on the condition that they are only used with designs incorporating an FTDI device (i.e. the FT245BM and DLP-USB245M). The latest version of the drivers can be downloaded from http://www,dlpdesign.com or http://www.ftdichip.com

The VCP driver download file is a combined set of drivers for Windows 98, Windows ME, and Windows 2000. Unzip the file to a blank floppy disk or folder on your PC. The drivers can coexist on the same floppy disk or folder since the INF files determine which set of drivers to load for each operating system version. Once loaded, the VCP drivers allow your application software, running on the host PC, to communicate with the DLP-USB245M as though it were connected to a COM (RS-232) port.

In addition to VCP drivers, FTDI's D2XX direct drivers for Windows offer an alternative solution to the VCP drivers that allow application software to interface with the FT245BM device using a DLL instead of a virtual com port. The architecture of the D2XX drivers consists of a Windows WDM driver that communicates with the FT245BM device via the Windows USB stack and a DLL that interfaces the application software (written in VC++, C++ Builder, Delphi, VB, etc.) to the WDM driver. An INF installation file, uninstaller program, and D2XX programmer's guide complete the package.

The D2XX direct drivers add support for simultaneous access and control of multiple FT245BM devices. The extended open function (FT_OpenEx) allows the device to be opened either by its product description or serial number, both of which can be programmed to be unique. The list devices function (FT-ListDevices) allows the application software to determine which devices are currently available for use, again by product description or by serial number.

Several new features have been recently added to the D2XX drivers. Functions are provided to program the EEPROM (FT_EE_Program), and read the EEPROM (FT_EE_Read). Unused space in the EEPROM is called the user area (EEUA), and functions are provided to access the EEUA. FT_EE_UASize gets its size, FT_EE_UAWrite writes data into it, and FT_EE_UARead is used to read its contents.

Download FTDI Application Notes AN232-03, AN232-05, AN232-06, and AN232-07 for detailed instructions on how to install and remove the drivers.

EEPROM WRITE UTILITY

The DLP-USB245M has the option to accept manufacturer-specific information that is written into the 93C46 EEPROM. Parameters that can be programmed include the VID and the PID identifiers, the manufacturer's product string, and a serial number.

FTD2XXST is the latest EEPROM serializer and testing utility for the FT245BM device. It replaces the VCP-based 232Prog utility. FTD2XXST is based on the new D2XX drivers and will work on Windows 98, Windows ME, and Windows 2000 platforms. You must install the latest release of the D2XX drivers in order to run this application. (Refer to the SER200.PDF guide that is included in the EEPROM write utility's download zip file for details on how to use the serializer.)

If you have VCP drivers installed on the PC that is to perform the EEPROM write process, you must uninstall these drivers using the uninstaller program (included with the driver files) and install the D2XX drivers prior to running the serializer utility.

QUICK START GUIDE

This guide requires the use of a Windows 98/2000 PC that is equipped with a USB port.

1. Download the DLL version of the device drivers from either dlpdesign.com or ftdichip.com. Unzip the drivers onto a blank floppy disk or into a folder on the hard drive.

2. Download the serializer program from either dlpdesign.com or ftdichip.com. Unzip the package, and place it in a folder on the hard drive.

3. Connect the DLP-USB245M board to the PC via a standard, 6-foot USB cable. This action initiates the loading of the USB drivers. When prompted, select the folder where the DLL version of the device drivers was stored in Step 1. Windows will then complete the installation of the device drivers for the DLP-USB245M board. The next time the DLP-USB245M module is attached, the host PC will immediately load the correct drivers without any prompting. Reboot the PC if prompted to do so.

4. Run the serializer program, and write the VID (0403), PID (6001), a description string of your choosing, and manufacturer's ID as directed by the instruction manual that was downloaded with the serializer software. Terminate the serializer program, and disconnect the DLP-USB245M board from the USB cable. Wait 10 seconds, and reconnect the DLP-USB245M board. Reboot the PC if instructed to do so.

Note: The "test" button on the serializer program is intended for use with the DLP-USB232M module (and FT232BM device) and will fail if used with the DLP-USB245M module. The "read" function will work and will present the contents of the EEPROM as well as the serial number and description strings stored in the EEPROM.

At this point, the DLP-USB245M is ready for use. Note that the DLP-USB245M will appear non-responsive if data sent from the host PC is not read from the FT245BM device by an attached microcontroller, microprocessor, DSP, FPGA, ASIC, etc.

If changing drivers from the VCP to the DLL type (or vice versa), you must first uninstall the currently loaded drivers. This is accomplished by first disconnecting the DLP-USB245M adapter from the host computer and then running the uninstall program for the currently loaded version of the drivers (Ftdiunin.exe for VCP and Ftd2xxun.exe for DLL). This uninstall program is included with the driver files.
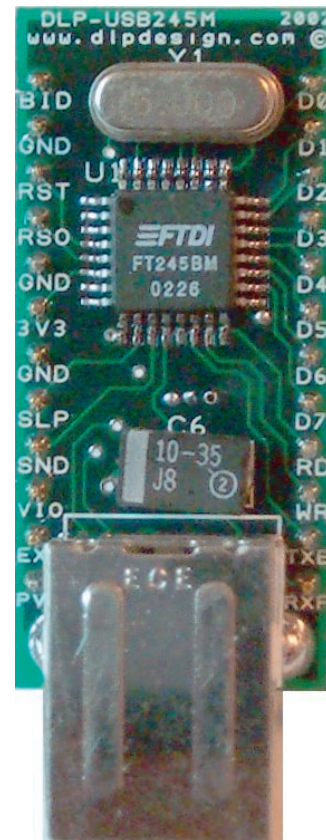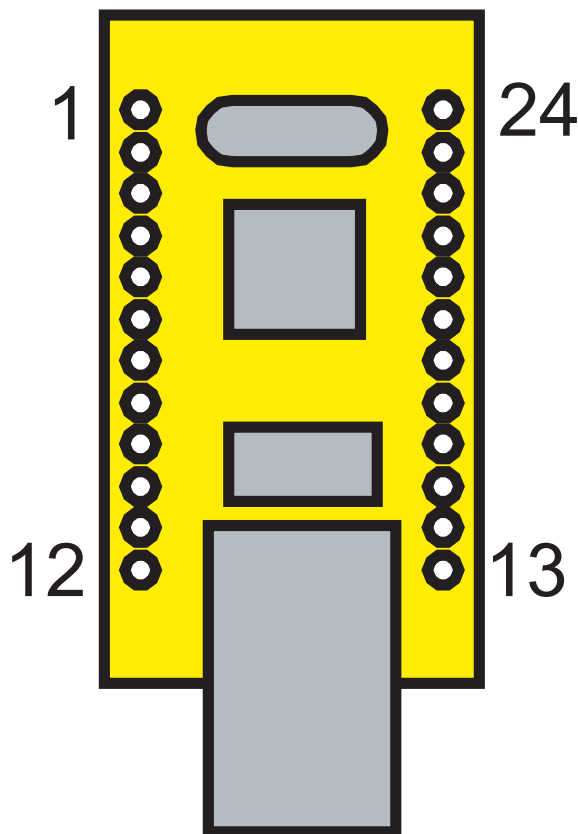
Table 1 - DLP-USB245M PINOUT DESCRIPTION

Pin#	Description
1	**BOARD ID** (Out) Identifies the board as either a DLP-USB245M or DLP-USB232M. High for DLP-USB232M and low for DLP-USB245M.
2	**GROUND**
3	**RESET#** (In) Can be used by an external device to reset the FT245BM. If not required, tie to VCC or leave n/c. This pin has a 47K pull-up to VCC.
4	**RESETO#** (Out) Output of the internal Reset Generator. Stays high impedance for ~2ms after VCC >3.5v and the internal clock starts up, then clamps its output to the 3.3v output of the internal regulator. Taking RESET# low will also force RSTOUT# to go high impedance. RSTOUT# is NOT affected by a USB bus reset.
5	**GROUND**
6	**3V3OUT** (Out) Output from the integrated L.D.O. regulator. Its primary purpose is to provide the internal 3.3v supply to the USB transceiver cell and the RSTOUT# pin. A small amount of current (<=5mA) can be drawn from this pin to power external 3.3v logic if required.
7	**GROUND**
8	**SLEEP** (Out) Goes low after the device is configured via USB, then high during USB suspend. Can be used to control power to external logic using a P-channel logic level MOSFET switch.

|	
9	**SND/WUP** (In) If the DLP-USB245M is in USB suspend, a positive edge on this pin (WAKEUP) initiates a remote wakeup sequence. If the device is active (not in suspend), a positive edge on this pin (SEND) causes the data in the write buffer to be sent to the PC on the next USB data-in request regardless of how many bytes are in the buffer.
10	**VCC-IO** (In) 3.0 volt to +5.25 volt VCC to the UART interface pins 10-12, 14-16, and 18-25. When interfacing with 3.3v external logic, connect VCC-IO to the 3.3v supply of the external logic; otherwise, connect to the VCC to drive out at 5v CMOS level. This pin must be connected to VCC from the target electronics or EXTVCC.
11	**EXTVCC** (In) Use for applying main power (4.4 to 5.25 volts) to the module. Connect to PORTVCC if module is to be powered by the USB port (typical configuration).
12	**PORTVCC** (Out) Power from USB port. Connect to EXTVCC if module is to be powered by the USB port (typical configuration). 500mA is the maximum current available to the USB adapter and target electronics if the USB device is configured for high power.
13	**RXF#** (Out) When low, at least 1 byte is present in the FIFO's 128-byte receive buffer and is ready to be read with RD#. RXF# goes high when the receive buffer is empty.
14	**TXE#** (Out) When low, at least 1 byte is present in the FIFO's 128-byte receive buffer and is ready to be read with RD#. RXF# goes high when the receive buffer is empty.
15	**WR** (In) When taken from a high to a low state, WR reads the eight data lines and writes the byte into the FIFO's transmit buffer. Data written to the transmit buffer is immediately sent to the host PC and placed in the RS-232 buffer opened by the application program.
16	**RD#** (In) When pulled low, RD# takes the eight data lines from a high impedance state to the current byte in the FIFO's receive buffer. Taking RD# high returns the data pins to a high impedance state and prepares the next byte (if available) in the FIFO to be read.
17	**D7** I/O bi-directional data bus bit #7
18	**D6** I/O bi-directional data bus bit #6
19	**D5** I/O bi-directional data bus bit #5
20	**D4** I/O bi-directional data bus bit #4
21	**D3** I/O bi-directional data bus bit #3
22	**D2** I/O bi-directional data bus bit #2
23	**D1** I/O bi-directional data bus bit #1
24	**D0** I/O bi-directional data bus bit #0

DEVICE CONFIGURATION EXAMPLES
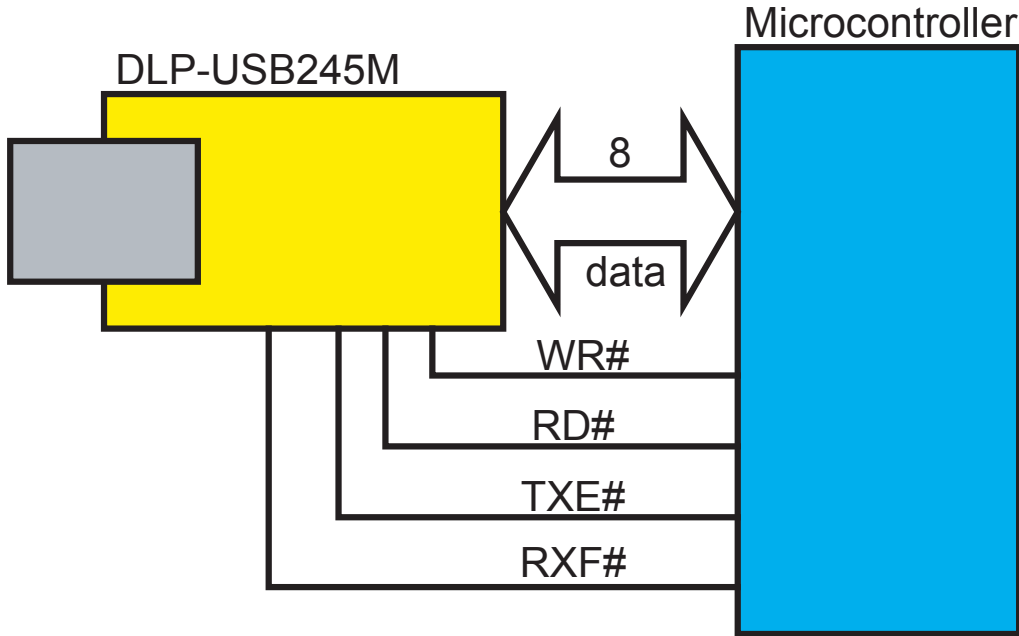
Basic Connection to a Microcontroller

USB Bus-Powered and Self-Powered Configuration

Figure 1 illustrates a typical USB bus-powered configuration. A USB bus-powered device gets its power from the USB bus. Basic rules for USB bus-powered devices are as follows:

a) On plug-in, the device must draw no more than 100mA.

b) On USB suspend, the device must draw no more than 500uA.

c) A bus-powered high-power device (one that draws more than 100mA) should use the SLEEP# pin to keep the current below 100mA on plug-in and 500uA on USB suspend.

d) A device that consumes more than 100mA cannot be plugged into a USB bus-powered hub.

e) No device can draw more that 500mA from the USB bus.

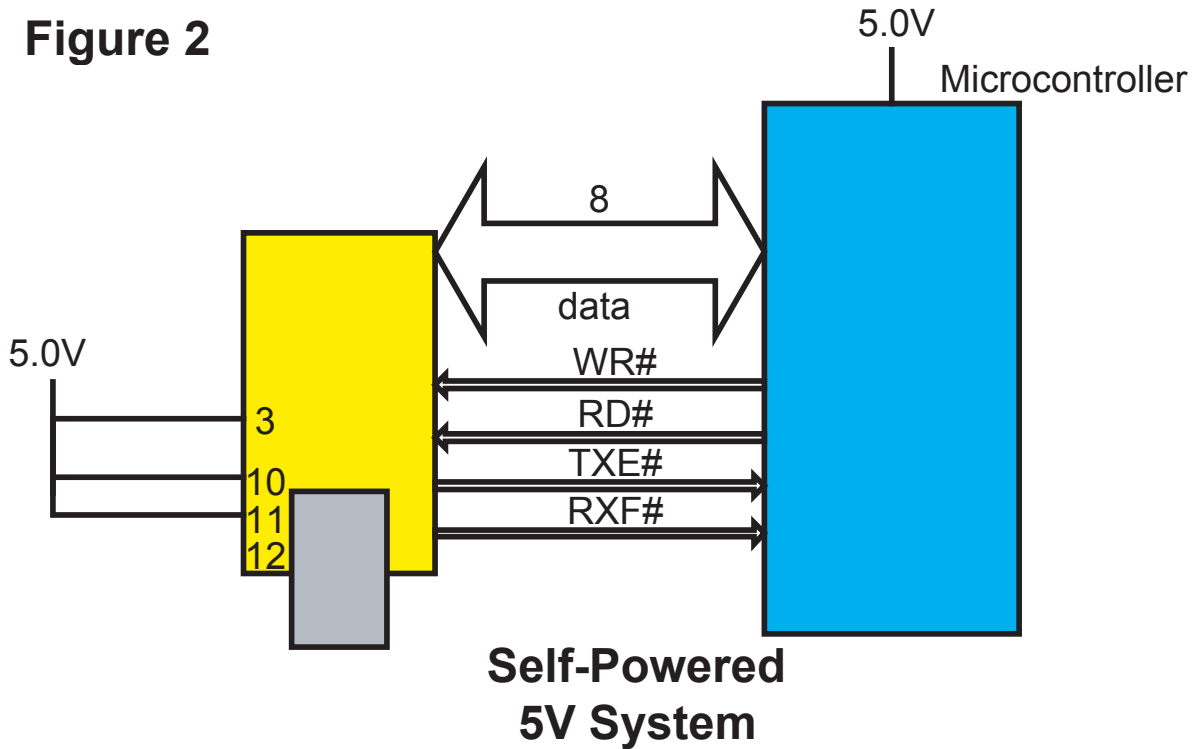f) The power descriptor in the EEPROM should be programmed to a value of zero.

Figure 1
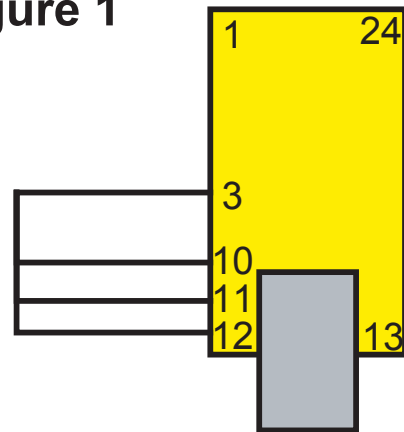
Basic Bus-Powered 5V System

Figure 2 illustrates a typical USB self-powered configuration.
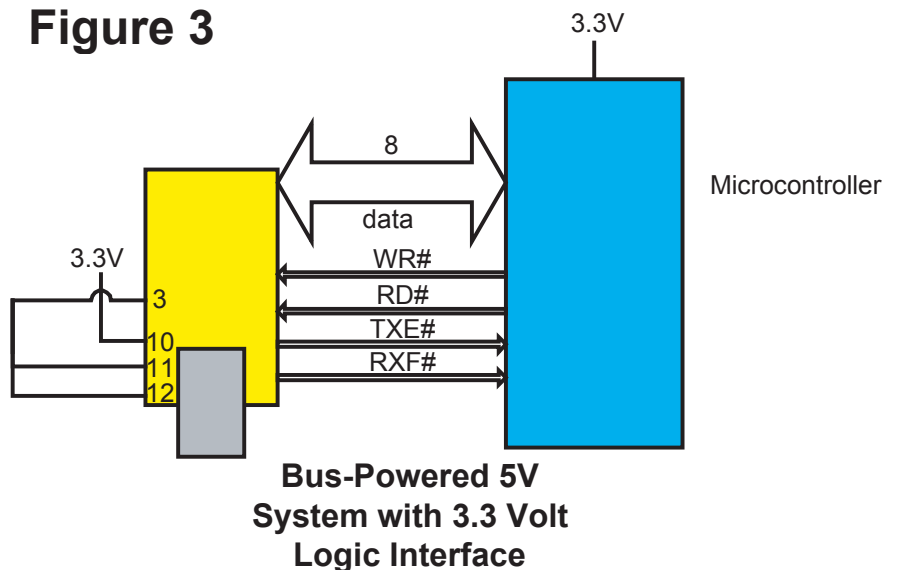
Figure 2

A USB self-powered device gets its power from its own power supply and does not draw current from the USB bus.

Basic rules for USB self-powered devices are as follows:

a) A self-powered device should not force current down the USB bus when the USB host or hub controller is powered down.

b) A self-powered device can take as much current as it likes during normal operation and USB suspend as it has its own power supply.

c) A self-powered device can be used with any USB host and both bus and self-powered USB hubs.

Figure 3 shows how to configure the DLP-USB245M to interface with a 3.3v logic device. In this example, the target electronics provides the 3.3 volts via the VCCIO line (Pin 10) which, in turn, will cause the FIFO interface IO pins to drive out at 3.3v level.

Figure 3

BUS-POWERED CIRCUIT WITH POWER CONTROL

USB bus-powered circuits need to be able to power down in USB suspend mode in order to meet the <= 500uA total suspend current requirement (including external logic).

Figure 4

Power Controlled by SLEEP

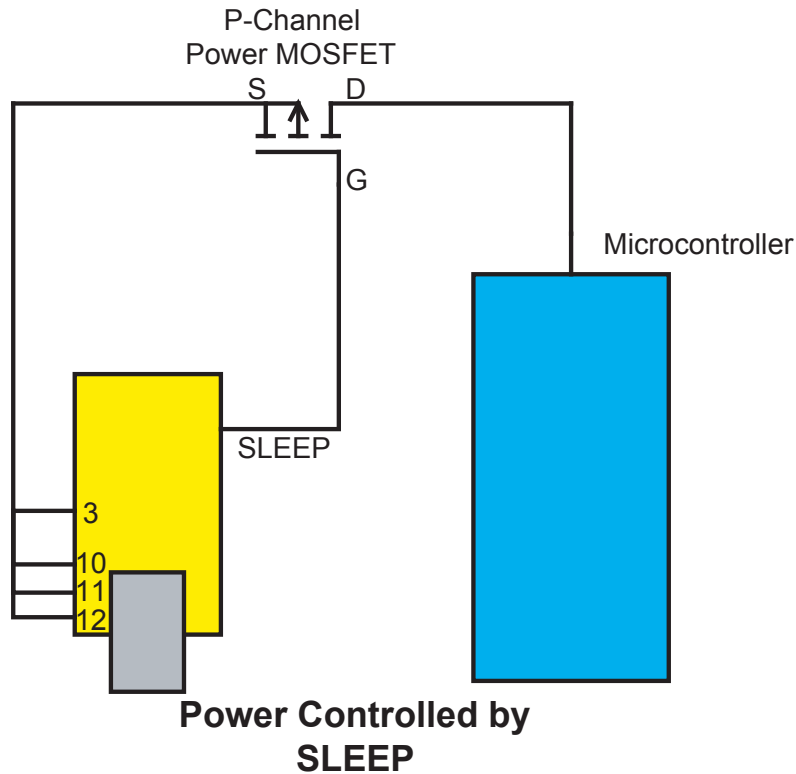**Figure 4** shows how to use a discrete P-channel logic level MOSFET to control the power to external logic circuits. A suitable device could be a Fairchild NDT456P or equivalent. This configuration is suitable for powering external logic where the normal supply current is <=100mA and the logic to be controlled does not generate an appreciable current surge at power-up. For power switching external logic that takes over 100mA or generates a current surge on power-up, we recommend that a dedicated power switch i.c. with an inbuilt "soft-start" be used instead of a MOSFET. A suitable power switch i.c. for such an application would be a Micrel (www.micrel.com) MIC2025-2BM or equivalent. Please note the following points in connection with power-controlled designs:

a) The logic to be controlled must have its own reset circuitry so that it will automatically reset itself when power is reapplied upon coming out of suspend.

b) Set the soft pull-down option bit in the FT232BM EEPROM.

c) For 3.3v power-controlled circuits, VCCIO must not be powered down with the external circuitry (PWREN# gets its VCC supply from VCCIO).

DLP-USB245M TIMING DIAGRAM – FIFO READ CYCLE
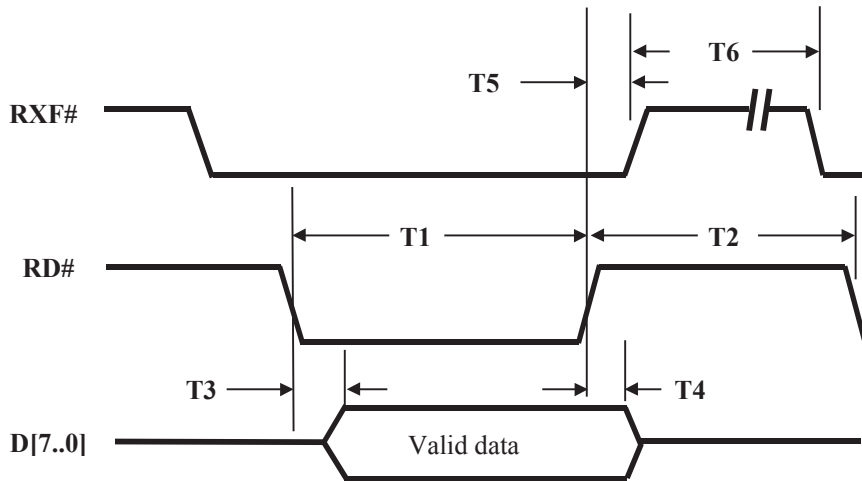
Time	Description	Min	Max	Unit
T1	RD Active Pulse Width	50		ns
T2	RD to RD Pre-Charge Time	50		ns
T3	RD Active to Valid Data		30	ns
T4	Valid Data Hold Time from RD Inactive	10		ns
T5	RD Inactive to RXF#	5	25	ns
T6	RXF inactive after RD cycle	80		ns

DLP-USB1 TIMING DIAGRAM – FIFO WRITE CYCLE
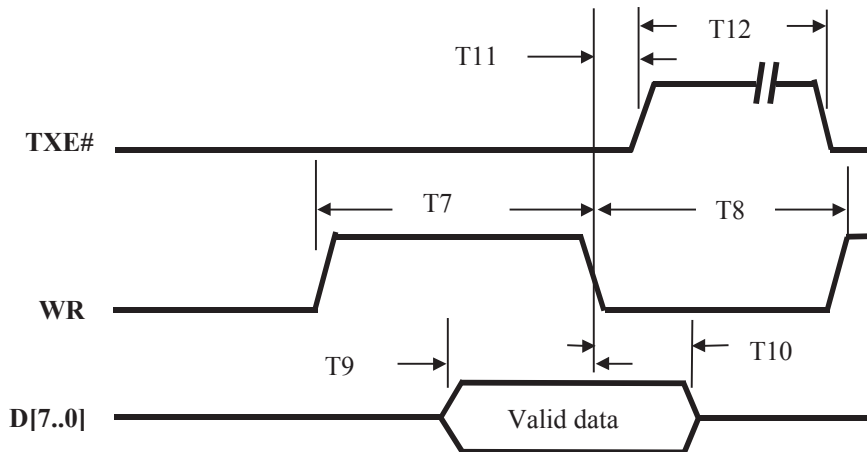
Time	Description	Min	Max	Unit
T7	WR Active Pulse Width	50		ns
T8	WR to WR Pre-Charge Time	50		ns
T9	Data Setup Time before WR inactive		20	ns
T10	Data Hold Time from WR inactive	10		ns
T11	WR Inactive to TXE#	5	25	ns
T12	TXE inactive after RD cycle	80		Ns

MECHANICAL DRAWINGS (PRELIMINARY)
INCHES (MILLIMETERS) UNLESS OTHERWISE NOTED

1.5 typ.
(38.1 typ.)

0.1 typ.
(2.54 typ.)

0.65 typ
(16.5 typ.)

0.235 typ
(6.0 typ.)

0.27 typ
(6.86 typ.)

0.50 typ
(12.7 typ.)

0.16 typ
(4.2 typ.)

.019 typ. dia
(0.48 typ.)

0.6 typ
(15.24 typ.)

0.7 typ.
(17.8 typ.)

0.35 typ
(9.0 typ.)
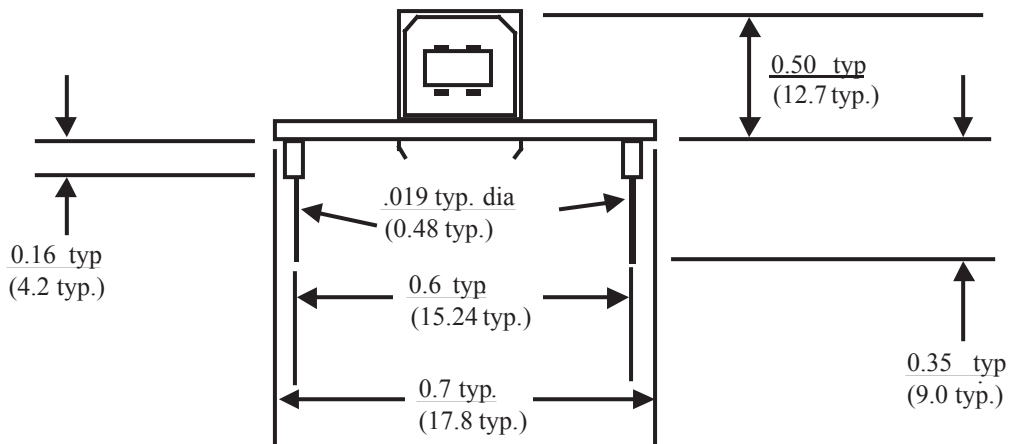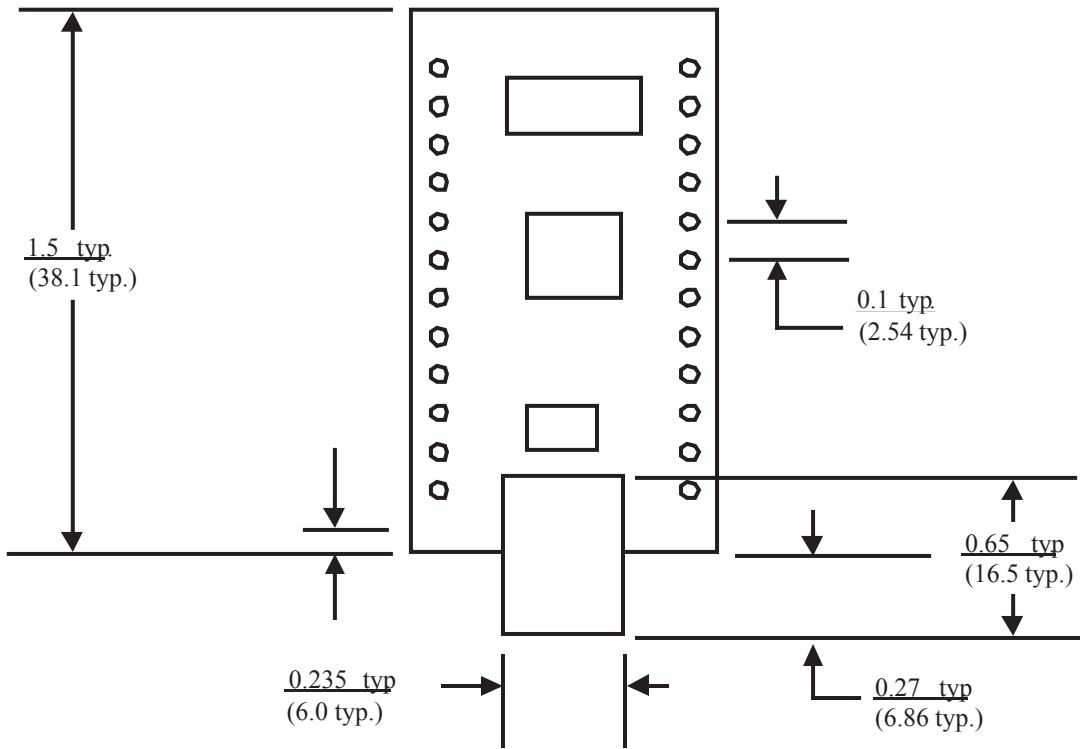
Absolute Maximum Ratings

These are the absolute maximum ratings for the FT245BM device in accordance with the Absolute Maximum Rating System (IEC 60134). Exceeding these may cause permanent damage to the device.

- Storage Temperature ……………………………………………… –65°C to + 150°C
- Ambient Temperature (Power Applied)………………………….. 0°C to + 70°C
- VCC Supply Voltage …………………………………………….. -0.5v to +6.00v
- DC Input Voltage - Inputs …………………………………….. -0.5v to VCC + 0.5v
- DC Input Voltage - High Impedance Bidirectionals ……………….. -0.5v to VCC + 0.5v
- DC Output Current – Outputs ……………………………………. 24mA
- DC Output Current – Low Impedance Bidirectionals ………………. 24mA
- Power Dissipation (VCC = 5.25v) ……………………………………… 500mW
- Electrostatic Discharge Voltage (I < 1uA) ……………………………. +/- 2000v
- Latch Up Current (Vi < 0 or Vi > Vcc) ……………………………….. 100mA

DC Characteristics (Ambient Temperature = 0 .. 70°C)

Operating Voltage and Current

Parameter	Description	Min	Typ	Max	Units	Conditions
Vcc1	VCC Operating Supply Voltage	4.4	5.0	5.25	V	
Vcc2	VCCIO Operating Supply Voltage	3.0	-	5.25	V	
Icc1	Operating Supply Current	-	25	-	mA	Normal Operation
Icc2	Operating Supply Current	-	350	400	uA	USB Suspend

FIFO IO Pin Characteristics (VCCIO = 5.0v)

Parameter	Description	Min	Typ	Max	Units	Conditions
Voh	Output Voltage High	4.4	-	4.9	V	I source = 2mA
Vol	Output Voltage Low	0.1	-	0.7	V	I sink = 4 mA
Vin	Input Switching Threshold	1.1	1.5	1.9	V	Note 1
VHys	Input Switching Hysteresis		200		mV	

FIFO IO Pin Characteristics (VCCIO = 3.3v)

Parameter	Description	Min	Typ	Max	Units	Conditions
Voh	Output Voltage High	2.7	-	3.2	V	I source = 2mA
Vol	Output Voltage Low	0.1	-	0.7	V	I sink = 4 mA
Vin	Input Switching Threshold	1.0	1.4	1.8	V	Note 1
VHys	Input Switching Hysteresis		200		mV	

Note 1 – Inputs have an internal 200k pull-up resistor to VCCIO.

RESET# Pin Characteristics

Parameter	Description	Min	Typ	Max	Units	Conditions
Vin	Input Switching Threshold	1.1	1.5	1.9	V	
VHys	Input Switching Hysteresis		200		mV	

RSTOUT Pin Characteristics

Parameter	Description	Min	Typ	Max	Units	Conditions
Voh	Output Voltage High	3.0	-	3.6	V	I source = 2mA
Iol	Leakage Current Tri-State	-	-	5	uA	

USB IO Pin Characteristics

Parameter	Description	Min	Typ	Max	Units	Conditions
UVoh	IO Pins Static Output (High)	2.8		3.6v	V	RI = 1k5 to 3V3Out (D+) RI = 15k to GND (D-)
UVol	IO Pins Static Output (Low)	0		0.3	V	RI = 1k5 to 3V3Out (D+) RI = 15k to GND (D-)
UVse	Single Ended Rx Threshold	0.8		2.0	V	
UCom	Differential Common Mode	0.8		2.5	V	
UVDif	Differential Input Sensitivity	0.2			V	
UDrvZ	Driver Output Impedance	29		44	ohm	Note 2

Note 2 – Driver Output Impedance includes the external 27R series resistors on USBDP and USBDM pins.

Disclaimer

Neither the whole nor any part of the information contained in, or the product described in this datasheet, may be adapted or reproduced in any material or electronic form without the prior written consent of the copyright holder.

This product and its documentation are supplied on an as is basis and no warranty as to their suitability for any particular purpose is either made or implied. DLP Design will not accept any claim for damages howsoever arising as a result of use or failure of this product. Your statutory rights are not affected.

This product or any variant of it is not intended for use in any medical appliance, device or system in which the failure of the product might reasonably be expected to result in personal injury.

This document provides preliminary information that may be subject to change without notice.

Contact Information

DLP Design
PO Box 503762
San Diego, CA 92150-3762
Phone: 858-513-2777
Fax: 858-513-2777
Email: support@dlpdesign.com
Internet: http://www.dlpdesign.com

Appendix A – DLP USB245M Schematic