The FT232BM is the 2nd generation of FTDI's popular USB UART I.C. This device not only adds extra functionality to its FT8U232AM predecessor and reduces external component count, but also maintains a high degree of pin compatibility with the original, making it easy to upgrade or cost reduce existing designs as well as increasing the potential for using the device in new application areas.

1.0 Features

HARDWARE FEATURES

- Single Chip USB ⇔Asynchronous Serial Data Transfer
- Full Handshaking & Modem Interface Signals
- UART I/F Supports 7 / 8 Bit Data, 1 / 2 Stop Bits and Odd/Even/Mark/Space/No Parity
- Data rate 300 => 3M Baud (TTL)
- Data rate 300 => 1M Baud (RS232)
- Data rate 300 => 3M Baud (RS422/RS485)
- 384 Byte Receive Buffer / 128 Byte Transmit Buffer for high data throughput
- Adjustable RX buffer timeout
- Fully Assisted Hardware or X-On / X-Off Handshaking
- In-built support for event characters and line break condition
- Auto Transmit Buffer control for RS485
- Support for USB Suspend / Resume through SLEEP# and RI# pins
- Support for high power USB Bus powered devices through PWREN# pin
- Integrated level converter on UART and control signals for interfacing to 5V and 3.3V logic
- Integrated 3.3V regulator for USB IO
- Integrated Power-On-Reset circuit
- Integrated 6MHz – 48Mhz clock multiplier PLL
- USB Bulk or Isochronous data transfer modes
- 4.35V to 5.25V single supply operation
- UHCI / OHCI / EHCI host controller compatible
- USB 1.1 and USB 2.0 compatible
- USB VID, PID, Serial Number and Product Description strings in external EEPROM
- EEPROM programmable on-board via USB
- Compact 32-LD LQFP package

VIRTUAL COM PORT (VCP) DRIVERS for

- Windows 98 and Windows 98 SE
- Windows 2000 / ME / Server 2003 / XP
- Windows XP 64 Bit
- Windows XP Embedded
- Windows CE 4.2
- 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 / Server 2003 / XP
- Windows XP 64 Bit
- Windows XP Embedded
- Windows CE 4.2
- Linux 2.40 and greater

APPLICATION AREAS

- USB ⇔RS232 Converters
- USB ⇔ RS422 / RS485 Converters
- Upgrading RS232 Legacy Peripherals to USB
- Cellular and Cordless Phone USB data transfer cables and interfaces
- Interfacing MCU based designs to USB
- USB Audio and Low Bandwidth Video data transfer
- PDA ⇔ USB data transfer
- USB Smart Card Readers
- Set Top Box (S.T.B.) PC - USB interface
- USB Hardware Modems
- USB Wireless Modems
- USB Instrumentation
- USB Bar Code Readers

2.0 Enhancements

This section summarises the enhancements of the 2nd generation device compared to its FT8U232AM predecessor. For further details, consult the device pin-out description and functional descriptions.

- **Integrated Power-On-Reset (POR) Circuit**
  The device now incorporates an internal POR function. The existing RESET# pin is maintained in order to allow external logic to reset the device where required, however for many applications this pin can now simply be hard wired to VCC. In addition, a new reset output pin (RSTOUT#) is provided in order to allow the new POR circuit to provide a stable reset to external MCU and other devices. RSTOUT# was the TEST pin on the previous generation of devices.

- **Integrated RCCLK Circuit**
  In the previous devices, an external RC circuit was required to ensure that the oscillator and clock multiplier PLL frequency was stable prior to enabling the clock internal to the device. This circuit is now embedded on-chip – the pin assigned to this function is now designated as the TEST pin and should be tied to GND for normal operation.

- **Integrated Level Converter on UART interface and control signals**
  The previous devices would drive the UART and control signals at 5V CMOS logic levels. The new device has a separate VCC-IO pin allowing the device to directly interface to 3.3V and other logic families without the need for external level converter I.C.'s

- **Improved Power Management control for USB Bus Powered, high current devices**
  The previous devices had a USBEN pin, which became active when the device was enumerated by USB. To provide power control, this signal had to be externally gated with SLEEP# and RESET#. This gating is now done on-chip - USBEN has now been replaced with the new PWREN# signal which can be used to directly drive a transistor or P-Channel MOSFET in applications where power switching of external circuitry is required. A new EEPROM based option makes the device pull gently down its UART interface lines when the power is shut off (PWREN# is High). In this mode, any residual voltage on external circuitry is bled to GND when power is removed thus ensuring that external circuitry controlled by PWREN# resets reliably when power is restored.

- **Lower Suspend Current**
  Integration of RCCLK within the device and internal design improvements reduce the suspend current of the FT232BM to under 200uA (excluding the 1.5k pull-up on USBDP) in USB suspend mode. This allows greater margin for peripherals to meet the USB Suspend current limit of 500uA.

- **Support for USB Isochronous Transfers**
  Whilst USB Bulk transfer is usually the best choice for data transfer, the scheduling time of the data is not guaranteed. For applications where scheduling latency takes priority over data integrity such as transferring audio and low bandwidth video data, the new device now offers an option of USB Isochronous transfer via an option bit in the EEPROM.

- **Programmable Receive Buffer Timeout**
  In the previous device, the receive buffer timeout used to flush remaining data from the receive buffer was fixed at 16ms timeout. This timeout is now programmable over USB in 1ms increments from 1ms to 255ms, thus allowing the device to be better optimised for protocols requiring faster response times from short data packets.

- **TXDEN Timing fix**
  TXDEN timing has now been fixed to remove the external delay that was previously required for RS485 applications at high baud rates. TXDEN now works correctly during a transmit send-break condition.

- **Relaxed VCC Decoupling**
  The 2nd generation devices now incorporate a level of on-chip VCC decoupling. Though this does not eliminate the need for external decoupling capacitors, it significantly improves the ease of PCB design requirements to meet FCC, CE and other EMI related specifications.

- **Improved PreScaler Granularity**
  The previous version of the Prescaler supported division by (n + 0), (n + 0.125), (n + 0.25) and (n + 0.5) where n is an integer between 2 and 16,384 (2^{14}). To this we have added (n + 0.375), (n + 0.625), (n + 0.75) and (n+ 0.875) which can be used to improve the accuracy of some baud rates and generate new baud rates which were previously impossible (especially with higher baud rates).

- **Bit Bang Mode**
  The 2nd generation device has a new option referred to as “Bit Bang” mode. In Bit Bang mode, the eight UART interface control lines can be switched between UART interface mode and an 8-bit Parallel IO port. Data packets can be sent to the device and they will be sequentially sent to the interface at a rate controlled by the prescaler setting. As well as allowing the device to be used stand-alone as a general purpose IO controller for example controlling lights, relays and switches, some other interesting possibilities exist. For instance, it may be possible to connect the device to an SRAM configurable FPGA as supplied by vendors such as Altera and Xilinx. The FPGA device would normally be un-configured (i.e. have no defined function) at power-up. Application software on the PC could use Bit Bang Mode to download configuration data to the FPGA which would define its hardware function, then after the FPGA device is configured the FT232BM can switch back into UART interface mode to allow the programmed FPGA device to communicate with the PC over USB. This approach allows a customer to create a “generic” USB peripheral who’s hardware function can be defined under control of the application software. The FPGA based hardware can be easily upgraded or totally changed simply by changing the FPGA configuration data file. Application notes, software and development modules for this application area will be available from FTDI and other 3rd parties.

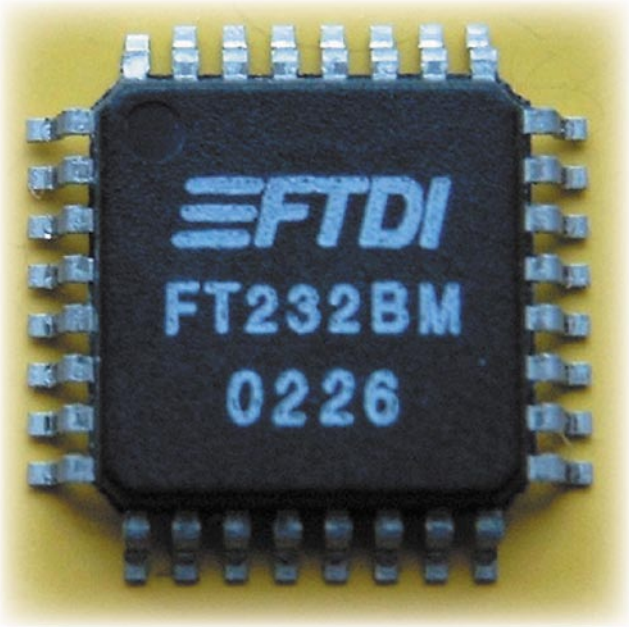
- **PreScaler Divide By 1 Fix**
  The previous device had a problem when the integer part of the divisor was set to 1. In the 2nd generation device setting the prescaler value to 1 gives a baud rate of 2 million baud and setting it to zero gives a baud rate of 3 million baud. Non-integer division is not supported with divisor values of 0 and 1.

- **Less External Support Components**
  As well as eliminating the RCCLK RC network, and for most applications the need for an external reset circuit, we have also eliminated the requirement for a 100K pull-up on EECS to select 6MHz operation. When the FT232BM is being used without the configuration EEPROM, EECS, EESK and EEDATA can now be left n/c. For circuits requiring a long reset time (where the device is reset externally using a reset generator I.C., or reset is controlled by the IO port of a MCU, FPGA or ASIC device) an external transistor circuit is no longer required as the 1.5k pull-up resistor on USBDP can be wired to the RSTOUT# pin instead of to 3.3V. Note : RSTOUT# drives out at 3.3V level, not at 5V VCC level. This is the preferred configuration for new designs.

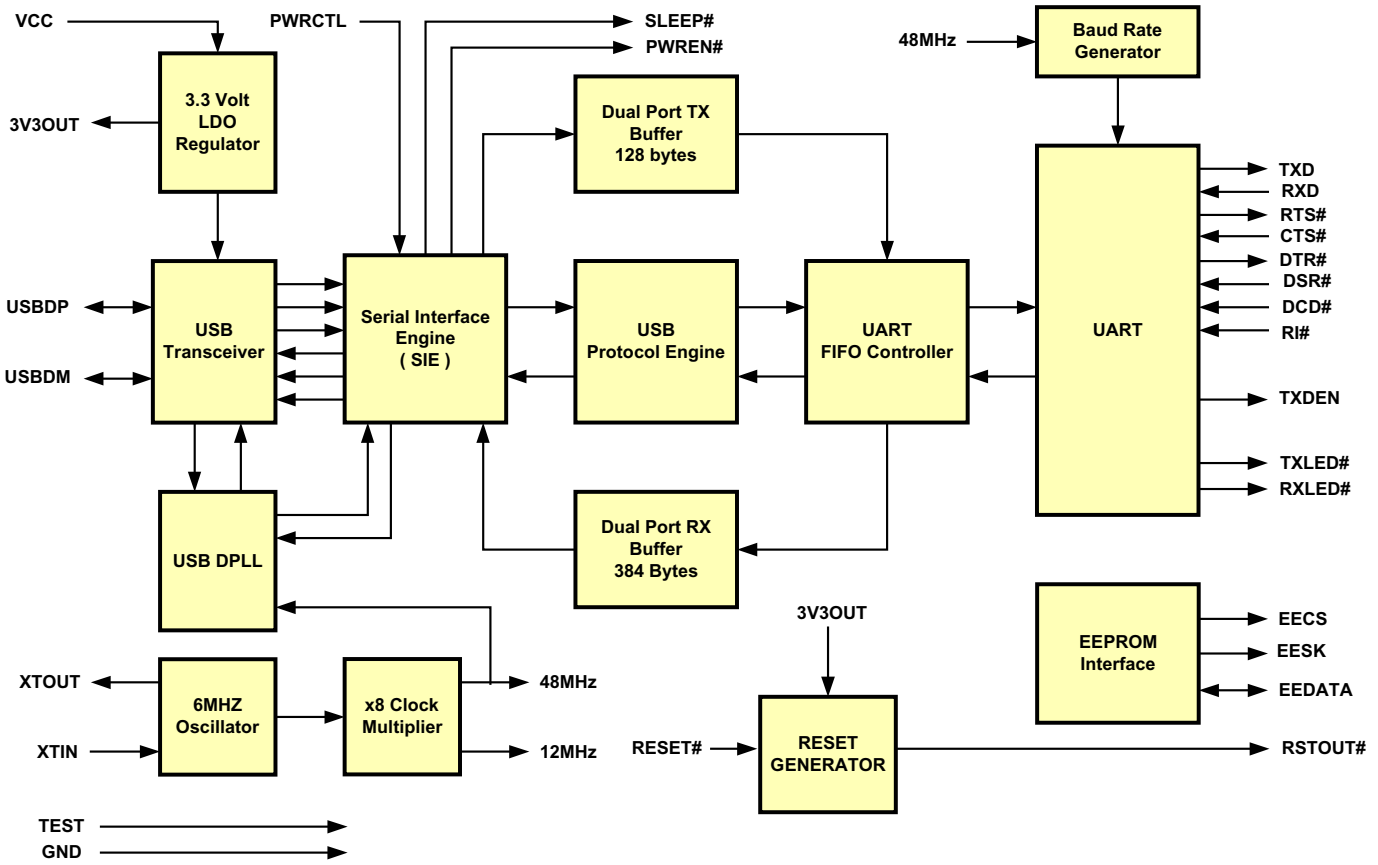
- **Extended EEPROM Support**
  The previous generation of devices only supported EEPROM of type 93C46 (64 x 16 bit). The new devices will also work with EEPROM type 93C56 (128 x 16 bit) and 93C66 (256 x 16 bit). The extra space is not used by the device, however it is available for use by other external MCU / logic whilst the FT232BM is being held in reset.

- **USB 2.0 (full speed option)**
  A new EEPROM based option allows the FT232BM to return a USB 2.0 device descriptor as opposed to USB 1.1. Note : The device would be a USB 2.0 Full Speed device (12Mb/s) as opposed to a USB 2.0 High Speed device (480Mb/s).

- **Multiple Device Support without EEPROM**
  When no EEPROM (or a blank or invalid EEPROM) is attached to the device, the FT232BM no longer gives a serial number as part of its USB descriptor. This allows multiple devices to be simultaneously connected to the same PC. However, we still highly recommend that EEPROM is used, as without serial numbers a device can only be identified by which hub port in the USB tree it is connected to which can change if the end user re-plugs the device into a different port.



3.0 Block Diagram (Simplified)



3.1 Functional Block Descriptions

- **3.3V LDO Regulator**

  The 3.3V LDO Regulator generates the 3.3 volt reference voltage for driving the USB transceiver cell output buffers. It requires an external decoupling capacitor to be attached to the 3V3OUT regulator output pin. It also provides 3.3V power to the RSTOUT# pin. The main function of this block is to power the USB Transceiver and the Reset Generator Cells rather than to power external logic. However, external circuitry requiring 3.3V nominal at a current of not greater than 5mA could also draw its power from the 3V3OUT pin if required.

- **USB Transceiver**

  The USB Transceiver Cell provides the USB 1.1 / USB 2.0 full-speed physical interface to the USB cable. The output drivers provide 3.3 volt level slew rate control signalling, whilst a differential receiver and two single ended receivers provide USB data in, SEO and USB Reset condition detection.

- **USB DPLL**

  The USB DPLL cell locks on to the incoming NRZI USB data and provides separate recovered clock and data signals to the SIE block.

- **6MHz Oscillator**

  The 6MHz Oscillator cell generates a 6MHz reference clock input to the x8 Clock multiplier from an external 6MHz crystal or ceramic resonator.

- **x8 Clock Multiplier**
  The x8 Clock Multiplier takes the 6MHz input from the Oscillator cell and generates a 12MHz reference clock for the SIE, USB Protocol Engine and UART FIFO controller blocks. It also generates a 48MHz reference clock for the USB DPPL and the Baud Rate Generator blocks.

- **Serial Interface Engine (SIE)**
  The Serial Interface Engine (SIE) block performs the Parallel to Serial and Serial to Parallel conversion of the USB data. In accordance to the USB 2.0 specification, it performs bit stuffing / un-stuffing and CRC5 / CRC16 generation / checking on the USB data stream.

- **USB Protocol Engine**
  The USB Protocol Engine manages the data stream from the device USB control endpoint. It handles the low level USB protocol (Chapter 9) requests generated by the USB host controller and the commands for controlling the functional parameters of the UART.

- **Dual Port TX Buffer (128 bytes)**
  Data from the USB data out endpoint is stored in the Dual Port TX buffer and removed from the buffer to the UART transmit register under control of the UART FIFO controller.

- **Dual Port RX Buffer (384 bytes)**
  Data from the UART receive register is stored in the Dual Port RX buffer prior to being removed by the SIE on a USB request for data from the device data in endpoint.

- **UART FIFO Controller**
  The UART FIFO controller handles the transfer of data between the Dual Port RX and TX buffers and the UART transmit and receive registers.

- **UART**
  The UART performs asynchronous 7 / 8 bit Parallel to Serial and Serial to Parallel conversion of the data on the RS232 (RS422 and RS485) interface. Control signals supported by the UART include RTS, CTS, DSR , DTR, DCD and RI. The UART provides a transmitter enable control signal (TXDEN) to assist with interfacing to RS485 transceivers. The UART supports RTS/CTS, DSR/DTR and X-On/X-Off handshaking options. Handshaking, where required, is handled in hardware to ensure fast response times. The UART also supports the RS232 BREAK setting and detection conditions.

- **Baud Rate Generator**
  The Baud Rate Generator provides a x16 clock input to the UART from the 48MHz reference clock and consists of a 14 bit prescaler and 3 register bits which provide fine tuning of the baud rate (used to divide by a number plus a fraction). This determines the Baud Rate of the UART which is programmable from 183 baud to 3 million baud.

- **RESET Generator**
  The Reset Generator Cell provides a reliable power-on reset to the device internal circuitry on power up. An additional RESET# input and RSTOUT# output are provided to allow other devices to reset the FT232BM or the FT232BM to reset other devices respectively. During reset, RSTOUT# is driven low, otherwise it drives out at the 3.3V provided by the onboard regulator. RSTOUT# can be used to control the 1.5k pull-up on USBDP directly where delayed USB enumeration is required. It can also be used to reset other devices. RSTOUT# will stay high-impedance for approximately 5ms after VCC has risen above 3.5V AND the device oscillator is running AND RESET# is high. RESET# should be tied to VCC unless it is a requirement to reset the device from external logic or an external reset generator i.c.

- **EEPROM Interface**

  Though the FT232BM will work without the optional EEPROM, an external 93C46 (93C56 or 93C66) EEPROM can be used to customise the USB VID, PID, Serial Number, Product Description Strings and Power Descriptor value of the FT232BM for OEM applications. Other parameters controlled by the EEPROM include Remote Wake Up, Isochronous Transfer Mode, Soft Pull Down on Power-Off and USB 2.0 descriptor modes.
  The EEPROM should be a 16 bit wide configuration such as a MicroChip 93LC46B or equivalent capable of a 1Mb/s clock rate at VCC = 4.35V to 5.25V. The EEPROM is programmable-on board over USB using a utility available from FTDI's web site (http://www.ftdichip.com). This allows a blank part to be soldered onto the PCB and programmed as part of the manufacturing and test process.
  If no EEPROM is connected (or the EEPROM is blank), the FT232BM will use its built-in default VID, PID Product Description and Power Descriptor Value. In this case, the device will not have a serial number as part of the USB descriptor.

4.0 Device Pin-Out

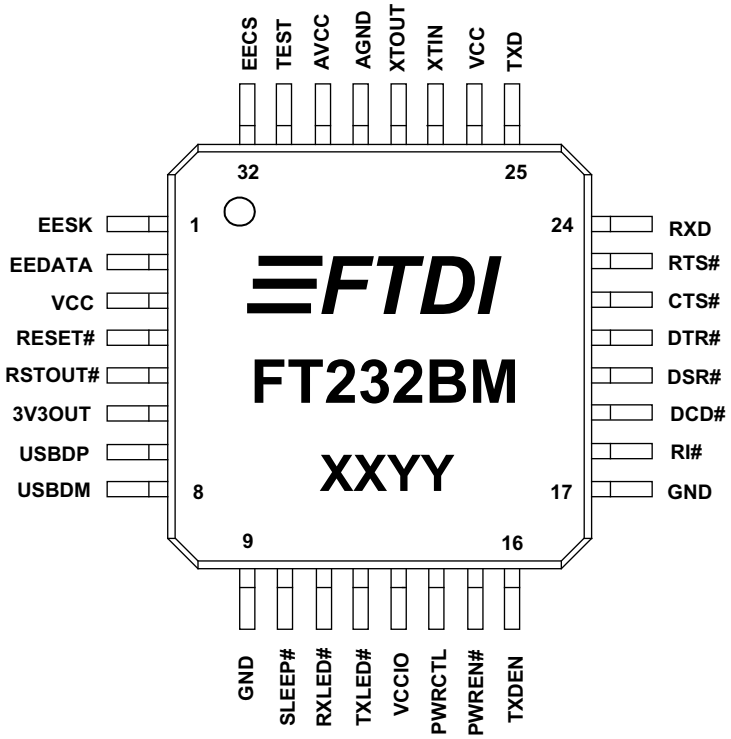


Figure 1
Pin-Out
(LQFP-32 Package)

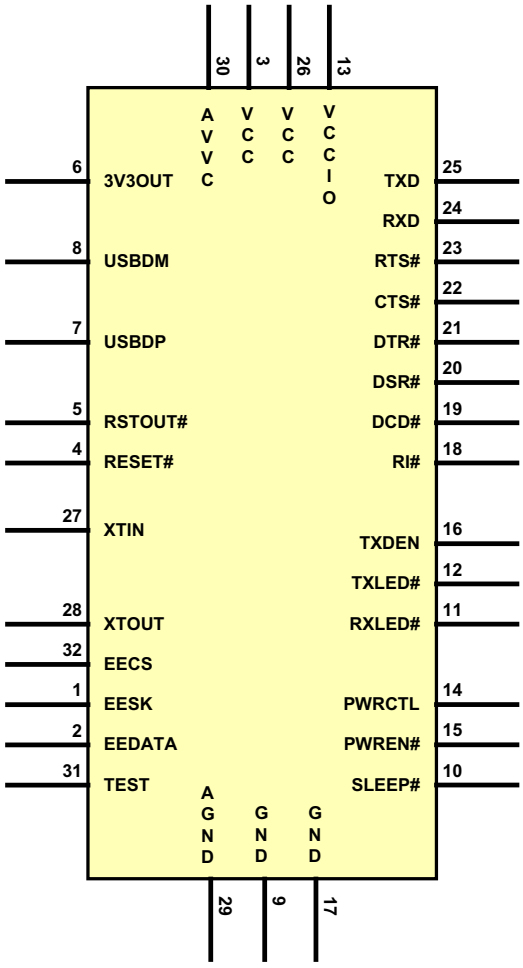


Figure 2
Pin-Out
(Schematic Symbol)

4.1 Signal Descriptions

Table 1 - FT232BM - PINOUT DESCRIPTION

UART INTERFACE GROUP

Pin#	Signal	Type	Description
25	TXD	OUT	Transmit Asynchronous Data Output
24	RXD	IN	Receive Asynchronous Data Input
23	RTS#	OUT	Request To Send Control Output / Handshake signal
22	CTS#	IN	Clear To Send Control Input / Handshake signal
21	DTR#	OUT	Data Terminal Ready Control Output / Handshake signal
20	DSR#	IN	Data Set Ready Control Input / Handshake signal
19	DCD#	IN	Data Carrier Detect Control Input
18	RI#	IN	Ring Indicator Control Input. When the Remote Wakeup option is enabled in the EEPROM, taking RI# low can be used to resume the PC USB Host controller from suspend.
16	TXDEN	OUT	Enable Transmit Data for RS485

USB INTERFACE GROUP

Pin#	Signal	Type	Description
7	USBDP	I/O	USB Data Signal Plus (Requires 1.5k pull-up to 3V3OUT or RSTOUT#)
8	USBDM	I/O	USB Data Signal Minus

EEPROM INTERFACE GROUP

Pin#	Signal	Type	Description
32	EECS	I/O	EEPROM – Chip Select. For 48MHz operation pull EECS to GND using a 10K resistor. For 6MHz operation no resistor is required. Tri-State during device reset. **Note 1
1	EESK	OUT	Clock signal to EEPROM. Tri-State during device reset, else drives out. Adding a 10K pull down resistor onto EESK will cause the FT232BM to use USB Product ID 6004 (hex) instead of 6001 (hex). All of the other USB device descriptors are unchanged.**Note 1
2	EEDATA	I/O	EEPROM – Data I/O Connect directly to Data-In of the EEPROM and to Data-Out of the EEPROM via a 2.2K resistor. Also, pull Data-Out of the EEPROM to VCC via a 10K resistor for correct operation. Tri-State during device reset. **Note 1

POWER CONTROL GROUP

Pin#	Signal	Type	Description
10	SLEEP#	OUT	Goes Low during USB Suspend Mode. Typically used to power-down an external TTL to RS232 level converter i.c. in USB <=> RS232 converter designs.
15	PWREN#	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. Enable the Interface Pull-Down Option in EEPROM when using the PWREN# pin in this way.
14	PWRCTL	IN	Bus Powered – Tie Low / Self Powered – Tie High (to VCCIO)

MISCELLANEOUS SIGNAL GROUP

Pin#	Signal	Type	Description
4	RESET#	IN	Can be used by an external device to reset the FT232BM. If not required, tie to VCC.
5	RSTOUT#	OUT	Output of the internal Reset Generator. Stays high impedance for ~ 5ms 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 drive low. RSTOUT# is NOT affected by a USB Bus Reset.
12	TXLED#	O.C.	LED Drive - Pulses Low when Transmitting Data via USB
11	RXLED#	O.C.	LED Drive - Pulses Low when Receiving Data via USB
27	XTIN	IN	Input to 6MHz Crystal Oscillator Cell. This pin can also be driven by an external 6MHz clock if required. Note : Switching threshold of this pin is VCC/2, so if driving from an external source, the source must be driving at 5V CMOS level or a.c. coupled to centre around VCC/2.
28	XTOUT	OUT	Output from 6MHz Crystal Oscillator Cell. XTOUT stops oscillating during USB suspend, so take care if using this signal to clock external logic.
31	TEST	IN	Puts device in I.C. test mode – must be tied to GND for normal operation.

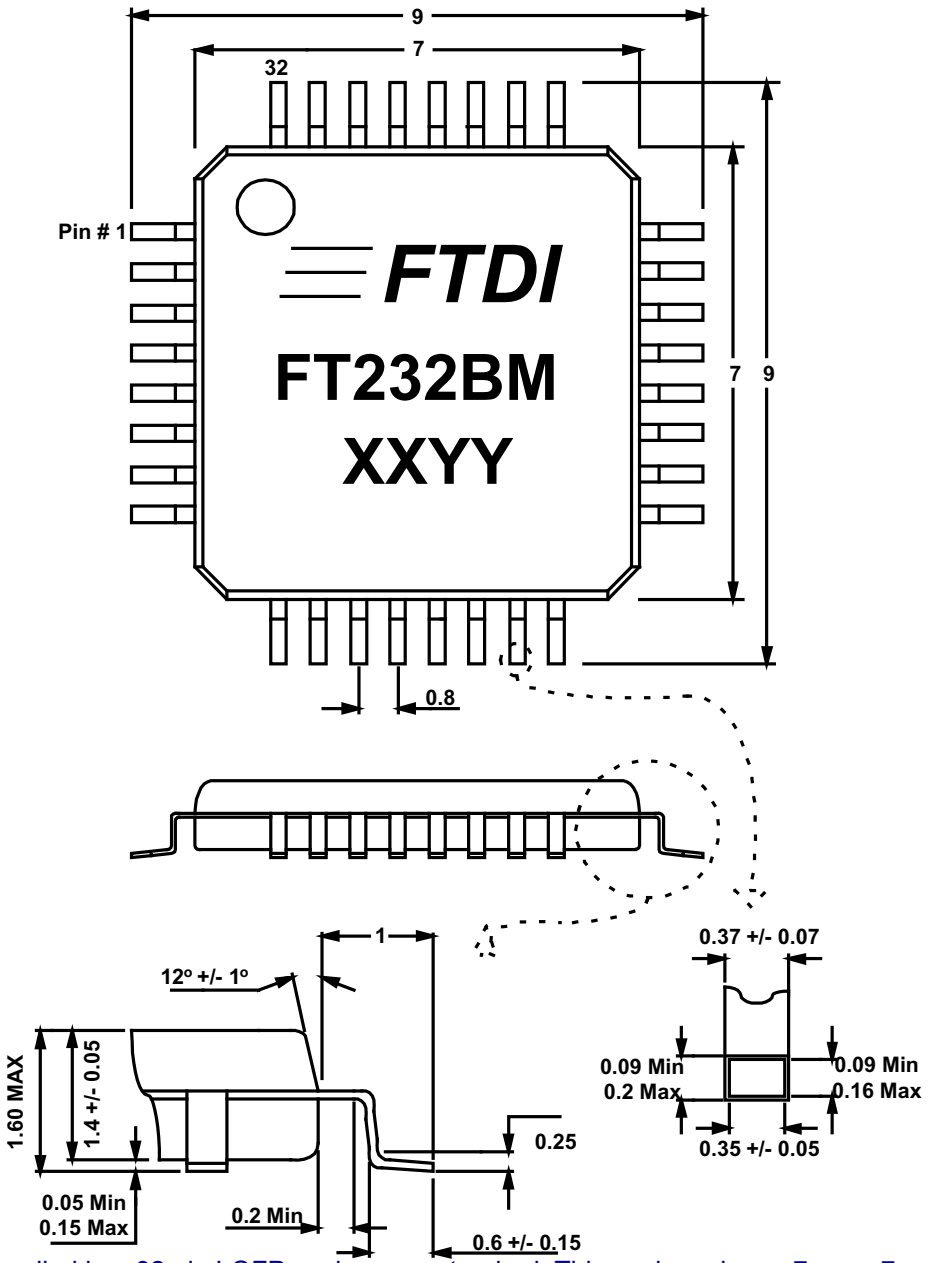
POWER AND GND GROUP

Pin#	Signal	Type	Description
6	3V3OUT	OUT	3.3 volt Output from the integrated L.D.O. regulator This pin should be decoupled to GND using a 33nF ceramic capacitor in close proximity to the device pin. Its prime 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.
3,26	VCC	PWR	+4.35 volt to +5.25 volt VCC to the device core, LDO and non-UART interface pins.
13	VCCIO	PWR	+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 in a bus powered design connect VCCIO to a 3.3V supply generated from the USB bus. When interfacing with 3.3V external logic in a self powered design connect VCCIO to the 3.3V supply of the external logic. Otherwise connect to VCC to drive out at 5V CMOS level.
9,17	GND	PWR	Device - Ground Supply Pins
30	AVCC	PWR	Device - Analog Power Supply for the internal x8 clock multiplier
29	AGND	PWR	Device - Analog Ground Supply for the internal x8 clock multiplier

****Note 1 -** During device reset, these pins are tri-state but pulled up to VCC via internal 200K resistors.

5.0 Package Outline

Figure 3 – 32 LD LQFP Package Dimensions



The FT232BM is supplied in a 32 pin LQFP package as standard. This package has a 7mm x 7mm body (9mm x 9mm including leads) with leads on a 0.8mm pitch. The above drawing shows the LQFP-32 package – all dimensions are in millimetres. Note that there are two date code formats used - XXYY = Date Code where XX = 2 digit year number, YY = 2 digit week number; or XYY-N where X = 1 digit year. number, YY = 2 digit week number, and -N is a integer.

A lead free version is available, part number FT232BL.

An alternative 5mm x 5mm leadless QFN32 package is also available for projects where package area is critical. Part number for this version is FT232BQ. The FT232BQ is also a lead free package. See their seperate datasheets for package dimensions.

6.0 Absolute Maximum Ratings

These are the absolute maximum ratings for the FT232BM device in accordance with the Absolute Maximum Rating System (IEC 60134). Exceeding these may cause permanent damage to the device.

Parameter	Value	Units
Storage Temperature	–65°C to + 150°C	Degrees C
Floor Life (Out of Bag) at Factory Ambient (30°C/60% Relative Humidity)	192 Hours (Level 3 Compliant) **Note 2	
Ambient Temperature (Power Applied)	0°C to + 70°C	Degrees C
M.T.B.F. (at 35°C)	247484 Hours ≈ 28 Years	
VCC Supply Voltage	-0.5 to +6.00	V
D.C. Input Voltage - USBDP and USBDM	-0.5 to +3.8	V
D.C. Input Voltage - High Impedance Bidirectionals	-0.5 to +(Vcc +0.5)	V
D.C. Input Voltage - All other Inputs	-0.5 to +(Vcc +0.5)	V
DC Output Current – Outputs	24	mA
DC Output Current – Low Impedance Bidirectionals	24	mA
Power Dissipation (VCC = 5.25V)	500	mW
Electrostatic Discharge Voltage (Human Body Model) (I < 1uA)	+/- 3000	V
Latch Up Current (Vi = +/- 10V maximum, for 10 ms)	+/-200	mA

**Note 2 – *If devices are stored out of the packaging beyond this time limit the devices should be baked before use. The devices should be ramped up to a temperature of 110°C and baked for 8 to 10 hours.*

6.1 D.C. Characteristics

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

Operating Voltage and Current

Parameter	Description	Min	Typ	Max	Units	Conditions
Vcc1	VCC Operating Supply Voltage	4.35	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	-	180	200	uA	USB Suspend **Note 3

**Note 3 – *Supply current excludes the 200uA nominal drawn by the external pull-up resistor on USBDP.*

UART IO Pin Characteristics (VCCIO = 5.0V)

Parameter	Description	Min	Typ	Max	Units	Conditions
Voh	Output Voltage High	3.2	4.1	4.9	V	I source = 2mA
Vol	Output Voltage Low	0.3	0.4	0.6	V	I sink = 2mA
Vin	Input Switching Threshold	1.3	1.6	1.9	V	**Note 4
VHys	Input Switching Hysteresis	50	55	60	mV	

UART IO Pin Characteristics (VCCIO = 3.0 - 3.6V)

Parameter	Description	Min	Typ	Max	Units	Conditions
Voh	Output Voltage High	2.2	2.7	3.2	V	I source = 1mA
Vol	Output Voltage Low	0.3	0.4	0.5	V	I sink = 2 mA
Vin	Input Switching Threshold	1.0	1.2	1.5	V	**Note 4
VHys	Input Switching Hysteresis	20	25	30	mV	

**Note 4 – *Inputs have an internal 200K pull-up resistor to VCCIO.*

XTIN / XTOUT Pin Characteristics

Parameter	Description	Min	Typ	Max	Units	Conditions
Voh	Output Voltage High	4.0	-	5.0	V	Fosc = 6MHz
Vol	Output Voltage Low	0.1	-	1.0	V	Fosc = 6MHz
Vin	Input Switching Threshold	1.8	2.5	3.2	V	

RESET#, TEST, EECS, EESK, EEDATA Pin Characteristics

Parameter	Description	Min	Typ	Max	Units	Conditions
Voh	Output Voltage High	3.2	4.1	4.9	V	I source = 2mA
Vol	Output Voltage Low	0.3	0.4	0.6	V	I sink = 2 mA
Vin	Input Switching Threshold	1.3	1.6	1.9	V	**Note 5
VHys	Input Switching Hysteresis	50	55	60	mV	

**Note 5 – *EECS, EESK and EEDATA pins have an internal 200K pull-up resistor to VCC*

RSTOUT Pin Characteristics

Parameter	Description	Min	Typ	Max	Units	Conditions
Voh	Output Voltage High	3.0	-	3.6	V	I source = 2mA
Vol	Output Voltage Low	0.3	-	0.6	V	I sink = 2mA

USB IO Pin Characteristics

Parameter	Description	Min	Typ	Max	Units	Conditions
UVoh	IO Pins Static Output (High)	2.8		3.6	V	RI = 1.5K to 3V3Out (D+) RI = 15K to GND (D-)
UVol	IO Pins Static Output (Low)	0		0.3	V	RI = 1.5K 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 5

**Note 5 – *Driver Output Impedance includes the external 27R series resistors on USBDP and USBDM pins.*

7.0 Device Configuration Examples

7.1 Oscillator Configurations

Figure 4
3-Pin Ceramic Resonator Configuration

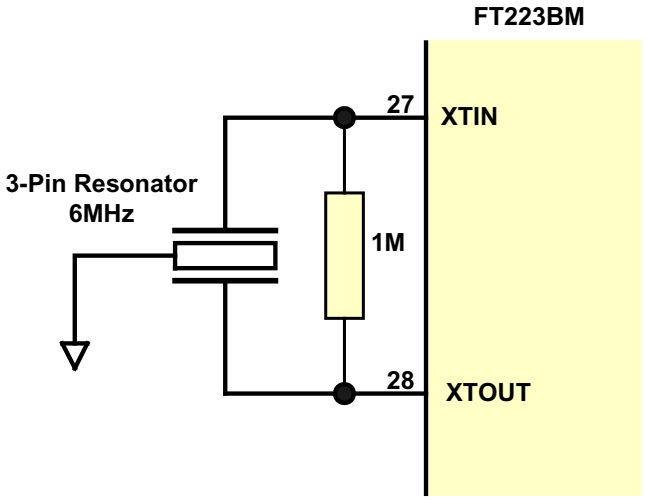


Figure 5
Crystal or 2-Pin Ceramic Resonator Configuration

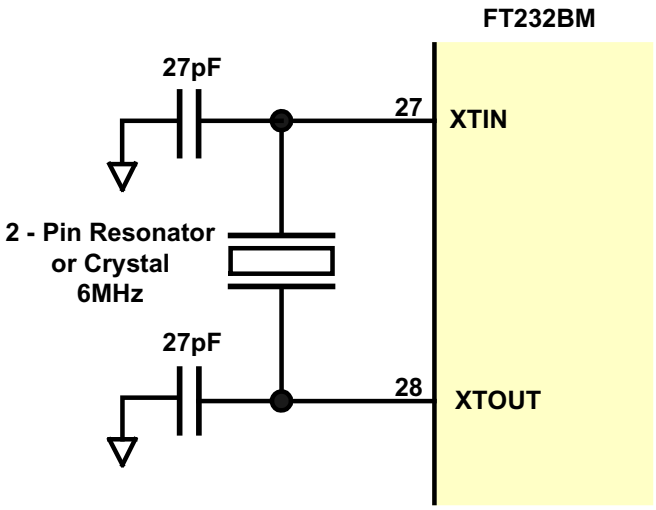


Figure 4 illustrates how to use the FT232BM with a 3-Pin Ceramic Resonator. A suitable part would be a ceramic resonator from Murata's CERALOCK range. (Murata Part Number CSTCR6M00G15), or equivalent. 3-Pin ceramic resonators have the load capacitors built into the resonator so no external loading capacitors are required. This makes for an economical configuration. The accuracy of this Murata ceramic resonator is +/- 0.1% and it is specifically designed for USB full speed applications. A 1 MegaOhm loading resistor across XTIN and XTOUT is recommended in order to guarantee this level of accuracy.

Other ceramic resonators with a lesser degree of accuracy (typically +/- 0.5%) are technically out-with the USB specification, but it has been calculated that using such a device will work satisfactorily in practice with a FT232BM design.

Figure 5 illustrates how to use the FT232BM with a 6MHz Crystal or 2-Pin Ceramic Resonator. In this case, these devices do not have in-built loading capacitors so these have to be added between XTIN, XTOUT and GND as shown. A value of 27pF is shown as the capacitor in the example – this will be good for many crystals and some resonators but do select the value based on the manufacturers recommendations wherever possible. If using a crystal, use a parallel cut type. If using a resonator, see the previous note on frequency accuracy.

7.2 EEPROM Configuration

Figure 6
EEPROM Configuration

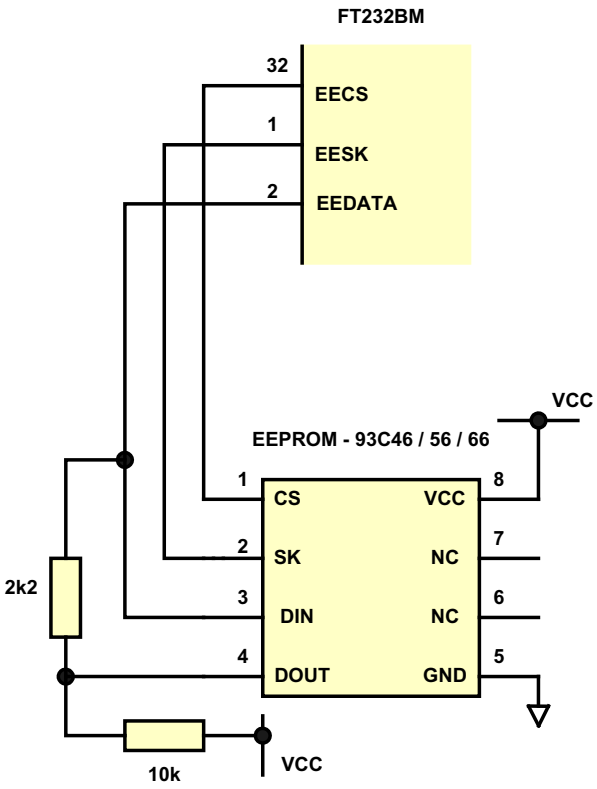


Figure 6 illustrates how to connect the FT232BM to the 93C46 (93C56 or 93C66) EEPROM. EECS (pin 32) is directly connected to the chip select (CS) pin of the EEPROM. EESK (pin 1) is directly connected to the clock (SK) pin of the EEPROM. EEDATA (pin 2) is directly connected to the Data In (Din) pin of the EEPROM. There is a potential condition whereby both the Data Output (Dout) of the EEPROM can drive out at the same time as the EEDATA pin of the FT232BM. To prevent potential data clash in this situation, the Dout of the EEPROM is connected to EEDATA of the FT232BM via a 2.2K resistor.

Following a power-on reset or a USB reset, the FT232BM will scan the EEPROM to find out (a) if an EEPROM is attached to the Device and (b) if the data in the device is valid. If both of these are the case, then the FT232BM will use the data in the EEPROM, otherwise it will use its built-in default values. When a valid command is issued to the EEPROM from the FT232BM, the EEPROM will acknowledge the command by pulling its Dout pin low. In order to check for this condition, it is necessary to pull Dout high using a 10K resistor. If the command acknowledge doesn't happen then EEDATA will be pulled high by the 10K resistor during this part of the cycle and the device will detect an invalid command or no EEPROM present.

There are two varieties of these EEPROM's on the market – one is configured as being 16 bits wide, the other is configured as being 8 bits wide. These are available from many sources such as Microchip, STMicro, ISSI etc. The FT232BM requires EEPROM's with a 16-bit wide configuration such as the Microchip 93LC46B device. The EEPROM must be capable of reading data at a 1Mb clock rate at a supply voltage of 4.35V to 5.25V. Most available parts are capable of this.

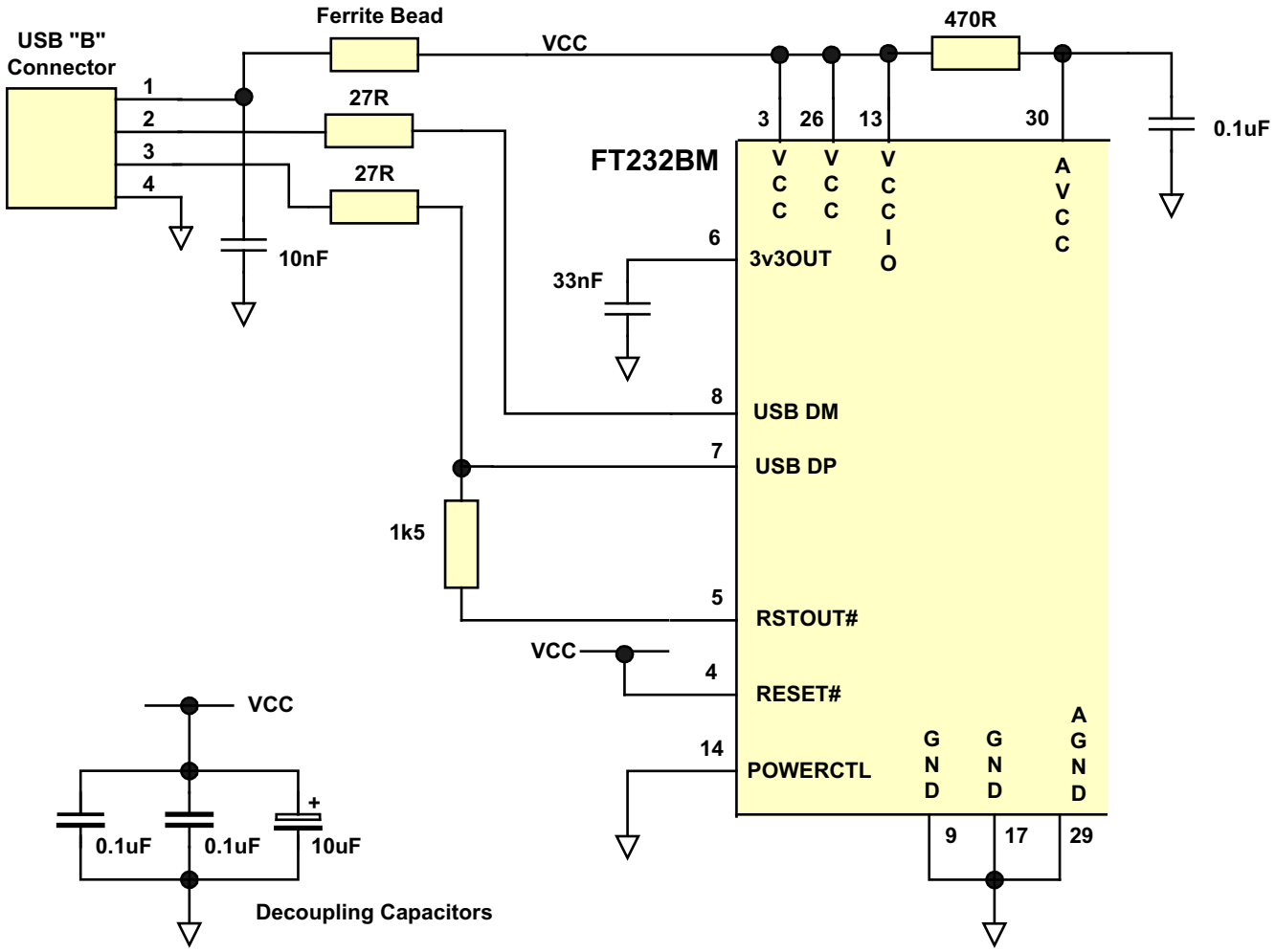
Check the manufacturers data sheet to find out how to connect pins 6 and 7 of the EEPROM. Some devices specify these as no-connect, others use them for selecting 8 / 16 bit mode or for test functions. Some other parts have their pinout rotated by 90° so please select the required part and its options carefully.

It is possible to "share" the EEPROM between the FT232BM and another external device such as an MCU. However, this can only be done when the FT232BM is in its reset condition as it tri-states its EEPROM interface at that time. A typical configuration would use four bit's of an MCU IO Port. One bit would be used to hold the FT232BM reset (using RESET#) on power-up, the other three would connect to the EECS, EESK and EEDATA pins of the FT232BM in order to read / write data to the EEPROM at this time. Once the MCU has read / written the EEPROM, it would take RESET# high to allow the FT232BM to configure itself and enumerate over USB.

7.3 USB Bus Powered and Self Powered Configuration

Figure 7
USB Bus Powered Configuration

Figure 7 illustrates a typical USB bus powered configuration. A USB Bus Powered device gets its power from the USB



bus. Basic rules for USB Bus power 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 PWREN# pin to keep the current below 100mA on plug-in and 500uA on USB suspend.
d) A device that consumes more than 100mA can not be plugged into a USB Bus Powered Hub
e) No device can draw more that 500mA from the USB Bus.

PWRCTL (pin 14) is pulled low to tell the device to use a USB Bus Power descriptor. The power descriptor in the EEPROM should be programmed to match the current draw of the device.

A Ferrite Bead is connected in series with USB power to prevent noise from the device and associated circuitry (EMI) being radiated down the USB cable to the Host. The value of the Ferrite Bead depends on the total current required by the circuit – a suitable range of Ferrite Beads is available from Steward (www.steward.com) for example Steward Part # **MI0805K400R-00 also available as DigiKey Part #** 240-1035-1.

Figure 8
USB Self Powered Configuration

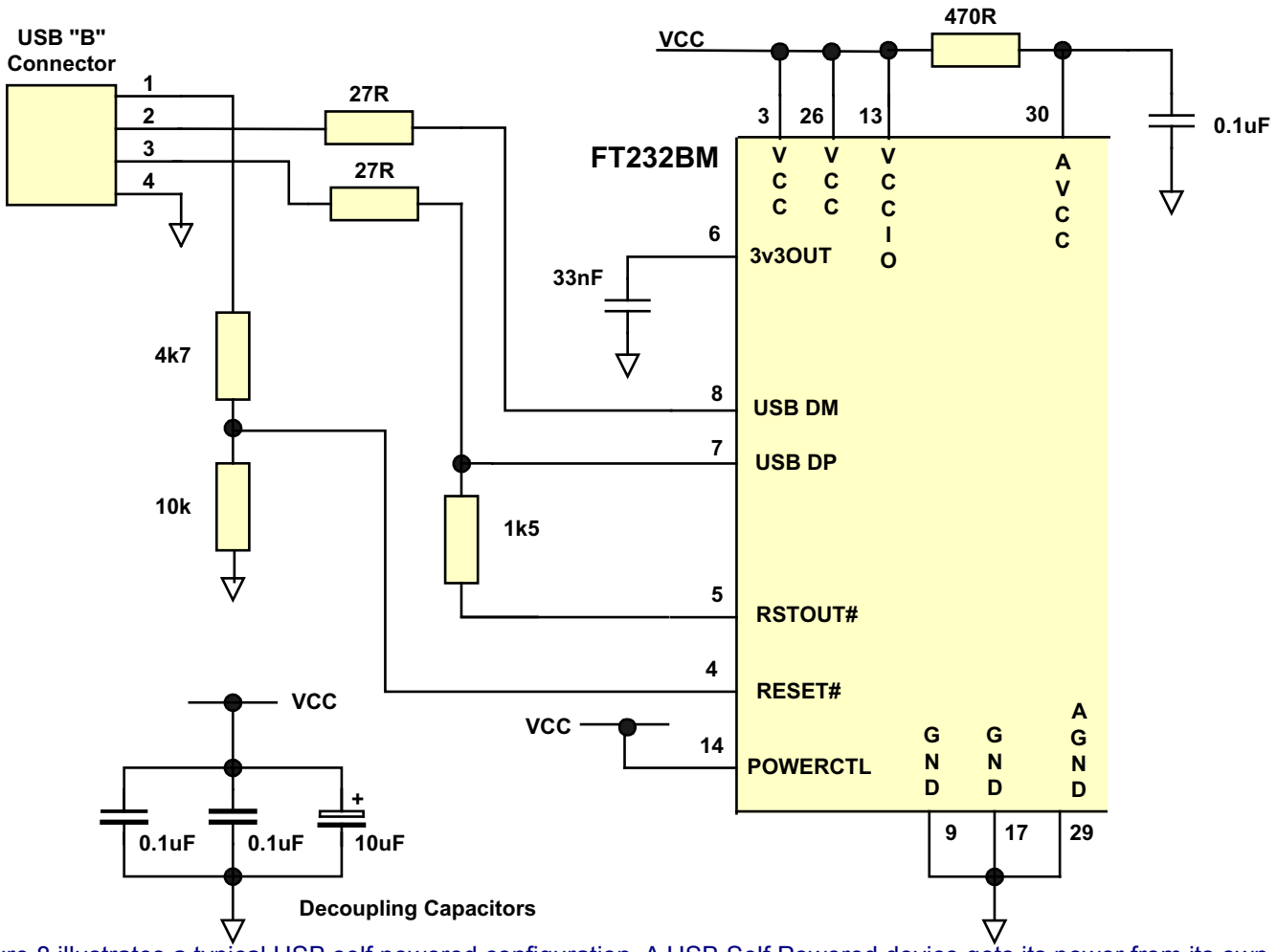


Figure 8 illustrates a typical USB self powered configuration. A USB Self Powered device gets its power from its own POWER SUPPLY and does not draw current from the USB bus. The basic rules for USB Self power devices are as follows –

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

PWRCTL (pin 14) is pulled high to tell the device to use a USB Bus Power descriptor. The power descriptor in the EEPROM should be programmed to a value of zero. The USB power descriptor option in the EEPROM should be programmed to a value of zero (self powered).

To meet requirement a) the 1.5K pull-up resistor on USBDP is connected to RSTOUT# as per the bus-power circuit. However, the USB Bus Power is used to control the RESET# Pin of the FT232BM device. When the USB Host or Hub is powered up RSTOUT# will pull the 1.5K resistor on USBDP to 3.3V, thus identifying the device as a full speed device to USB. When the USB Host or Hub power is off, RESET# will go low and the device will be held in reset. As RESET# is low, RSTOUT# will also be low, so no current will be forced down USBDP via the 1.5K pull-up resistor when the host or hub is powered down. Failure to do this may cause some USB host or hub controllers to power up erratically.

Note : When the FT232BM is in reset, the UART interface pins all go tri-state. These pins have internal 200K pull-up resistors to VCCIO, so they will gently pull high unless driven by some external logic.

7.4 UART Interface Configuration

Figure 9
USB <=> RS232 Converter Configuration

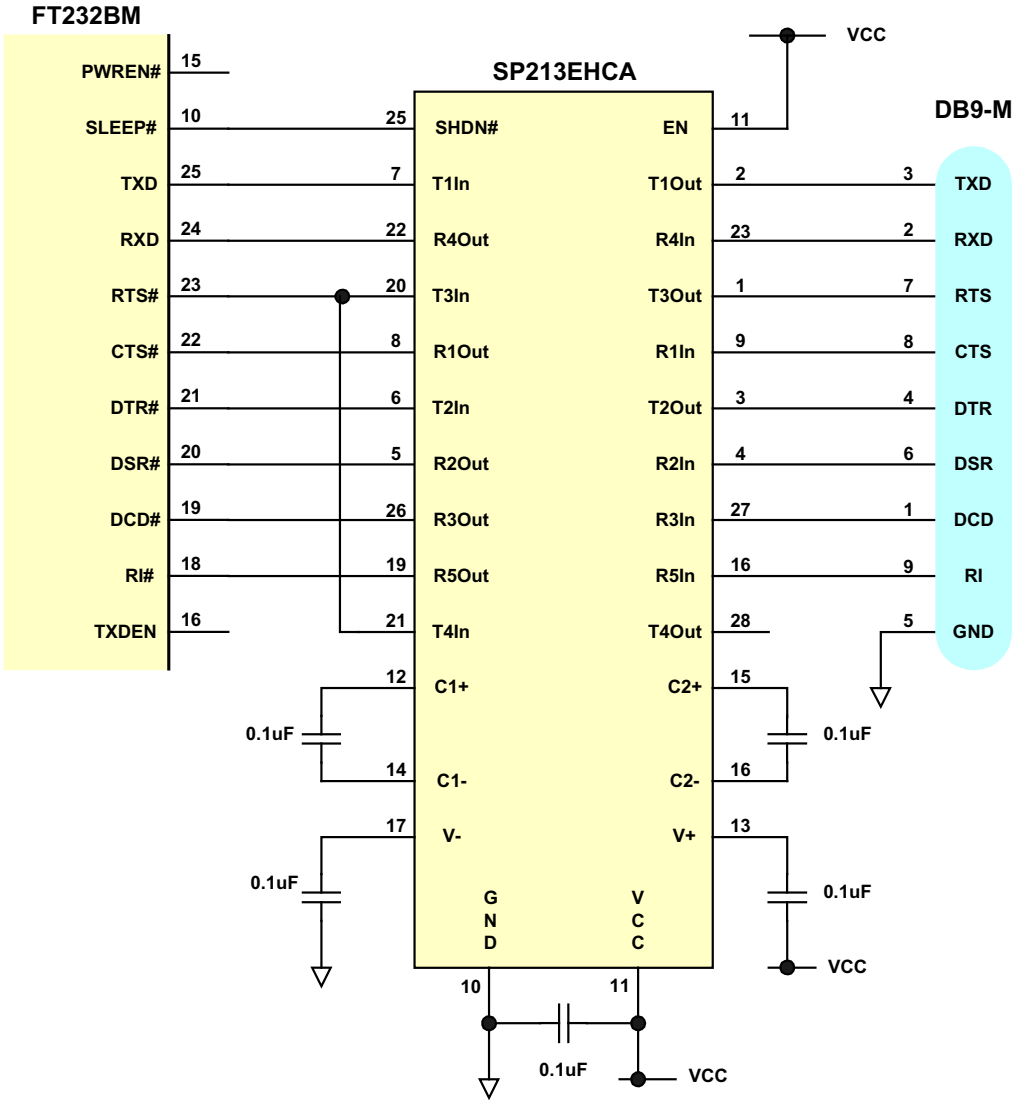


Figure 9 illustrates how to connect the UART interface of the FT232BM to a TTL – RS232 Level Converter I.C. to make a USB <=> RS232 converter using the popular “213” series of TTL to RS232 level converters. These devices have 4 transmitters and 5 receivers in a 28 LD SSOP package and feature an in-built voltage converter to convert the 5v (nominal) VCC to the +/- 9 volts required by RS232. An important feature of these devices is the SHDN# pin which can power down the device to a low quiescent current during USB suspend mode

The device used in the example is a Sipex SP213EHCA which is capable of RS232 communication at up to 500K baud. If a lower baud rate is acceptable, then several pin compatible alternatives are available such as Sipex SP213ECA , Maxim MAX213CAI and Analog Devices ADM213E which are good for communication at up to 115,200 baud. If a higher baud rate is desired, use a Maxim MAX3245CAI part which is capable of RS232 communication at rates of up to 1M baud. The MAX3245 is not pin compatible with the 213 series devices, also its SHDN pin is active high so connect this to PWREN# instead of SLEEP#.

Figure 10
USB <=> RS422 Converter Configuration

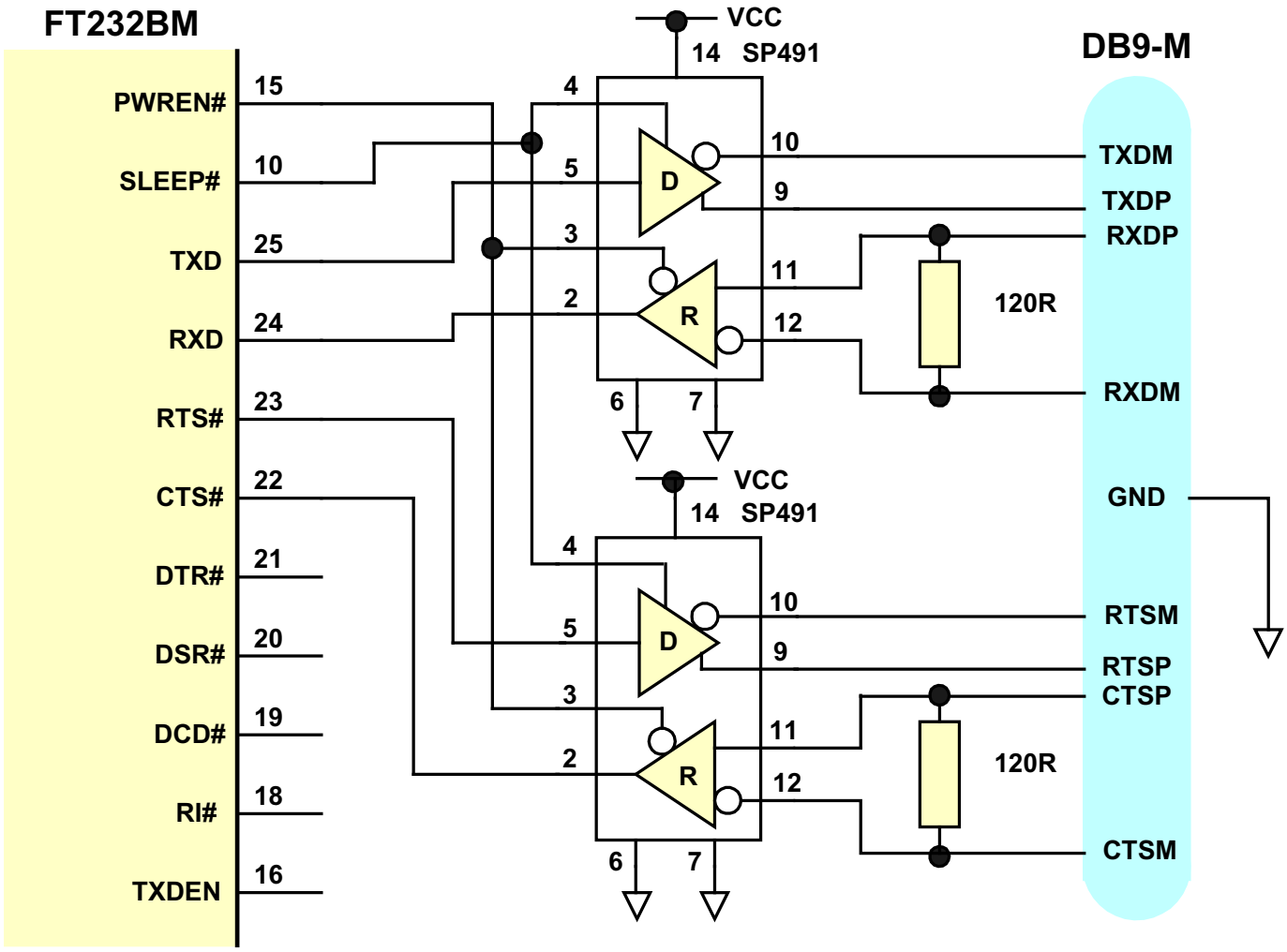


Figure 10 illustrates how to connect the UART interface of the FT232BM to a TTL – RS422 Level Converter I.C. to make a USB <=> RS422 converter. There are many such level converter devices available – this example uses Sipex SP491 devices which have enables on both the transmitter and receiver. Because the transmitter enable is active high, it is connected to the SLEEP# pin. The receiver enable is active low and is connected to the PWREN# pin. This ensures that both the transmitters and receivers are enabled when the device is active, and disabled when the device is in USB suspend mode. If the design is USB BUS powered, it may be necessary to use a P-Channel logic level MOSFET (controlled by PWREN#) in the VCC line of the SP491 devices to ensure that the USB standby current of 500uA is met.

The SP491 is good for sending and receiving data at a rate of up to 5M Baud – in this case the maximum rate is limited to 3M Baud by the FT232BM.

Figure 11
USB <=> RS485 Converter Configuration

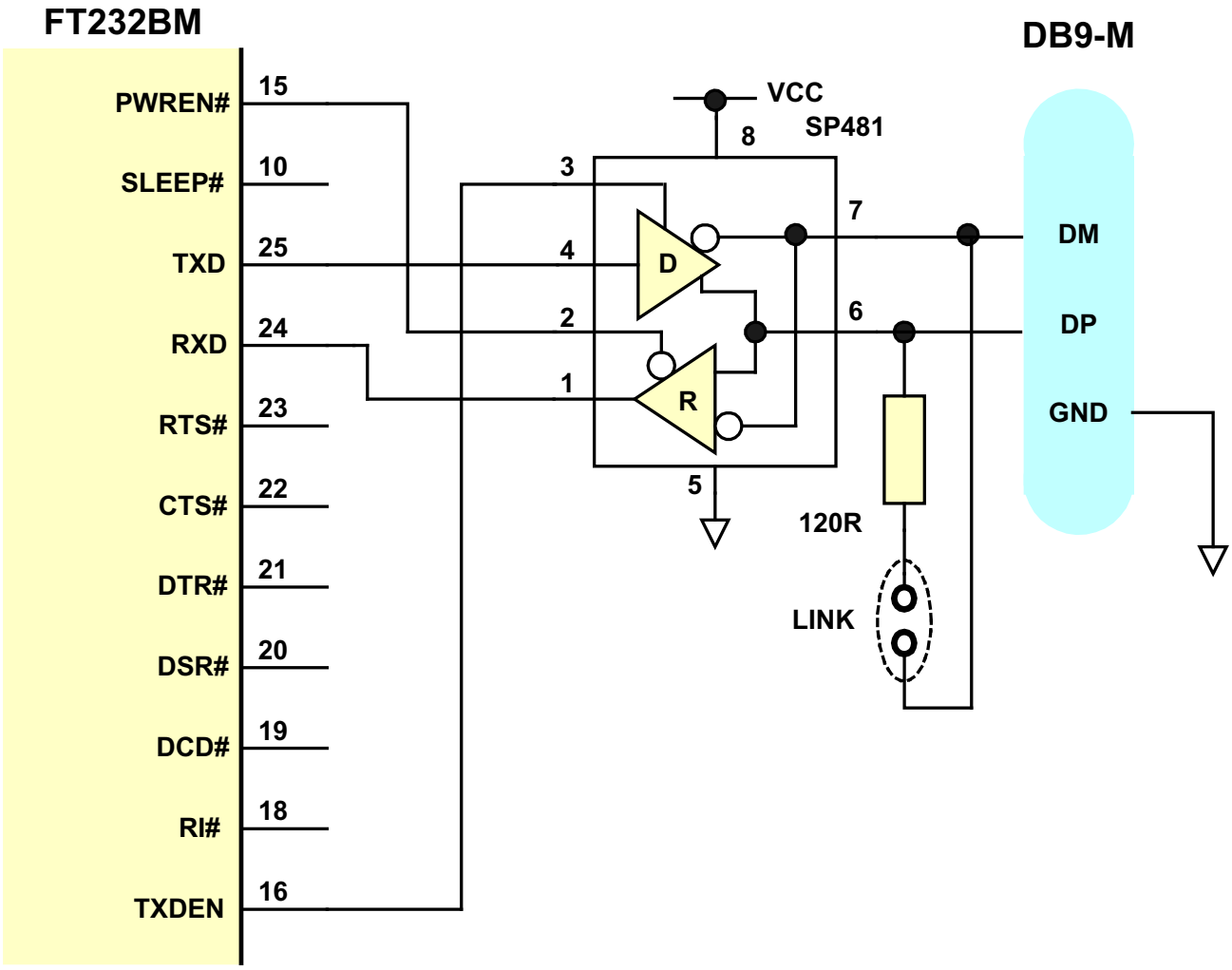


Figure 11 illustrates how to connect the UART interface of the FT232BM to a TTL – RS485 Level Converter I.C. to make a USB => RS485 converter. This example uses the Sipex SP481 device but there are similar parts available from Maxim and Analog Devices amongst others. The SP481 is a RS485 device in a compact 8 pin SOP package. It has separate enables on both the transmitter and receiver. With RS485, the transmitter is only enabled when a character is being transmitted from the UART. The TXDEN pin on the FT232BM is provided for exactly that purpose and so the transmitter enable is wired to TXDEN. The receiver enable is active low, so it is wired to the PWREN# pin to disable the receiver when in USB suspend mode.

RS485 is a multi-drop network – i.e. many devices can communicate with each other over a single two wire cable connection. The RS485 cable requires to be terminated at each end of the cable. A link is provided to allow the cable to be terminated if the device is physically positioned at either end of the cable.

In this example the data transmitted by the FT232BM is also received by the device that is transmitting. This is a common feature of RS485 and requires the application software to remove the transmitted data from the received data stream. With the FT232BM it is possible to do this entirely in hardware – simply modify the schematic so that RXD of the FT232BM is the logical OR of the SP481 receiver output with TXDEN using an HC32 or similar logic gate.

Figure 12
Dual LED Configuration

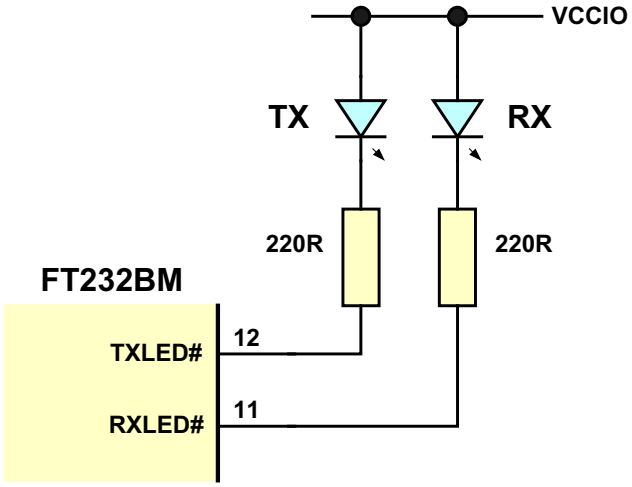
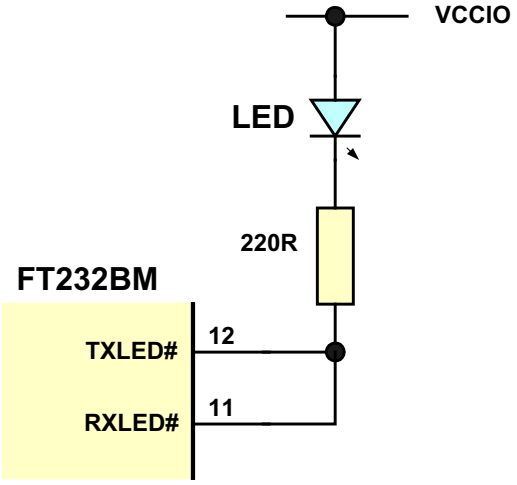


Figure 13
Single LED Configuration



The FT232BM has two IO pins dedicated to controlling LED status indicators, one for transmitted data the other for received data. When data is being transmitted / received the respective pins drive from tri-state to low in order to provide indication on the LEDs of data transfer. A digital one-shot timer is used so that even a small percentage of data transfer is visible to the end user. Figure 12 shows a configuration using two individual LED's – one for transmitted data the other for received data. In Figure 13, the transmit and receive LED indicators are wire-OR'ed together to give a single LED indicator which indicates any transmit or receive data activity.
Another possibility (not shown here) is to use a 3 pin common anode tri-color LED based on the circuit in Figure 13 to have a single LED that can display activity in a variety of colors depending on the ratio of transmit activity compared to receive activity.

Note that the LED's are connected to VCCIO.

7.6 Interfacing to 3.3v Logic

Figure 14
Bus Powered Circuit with 3.3V logic drive / supply voltage

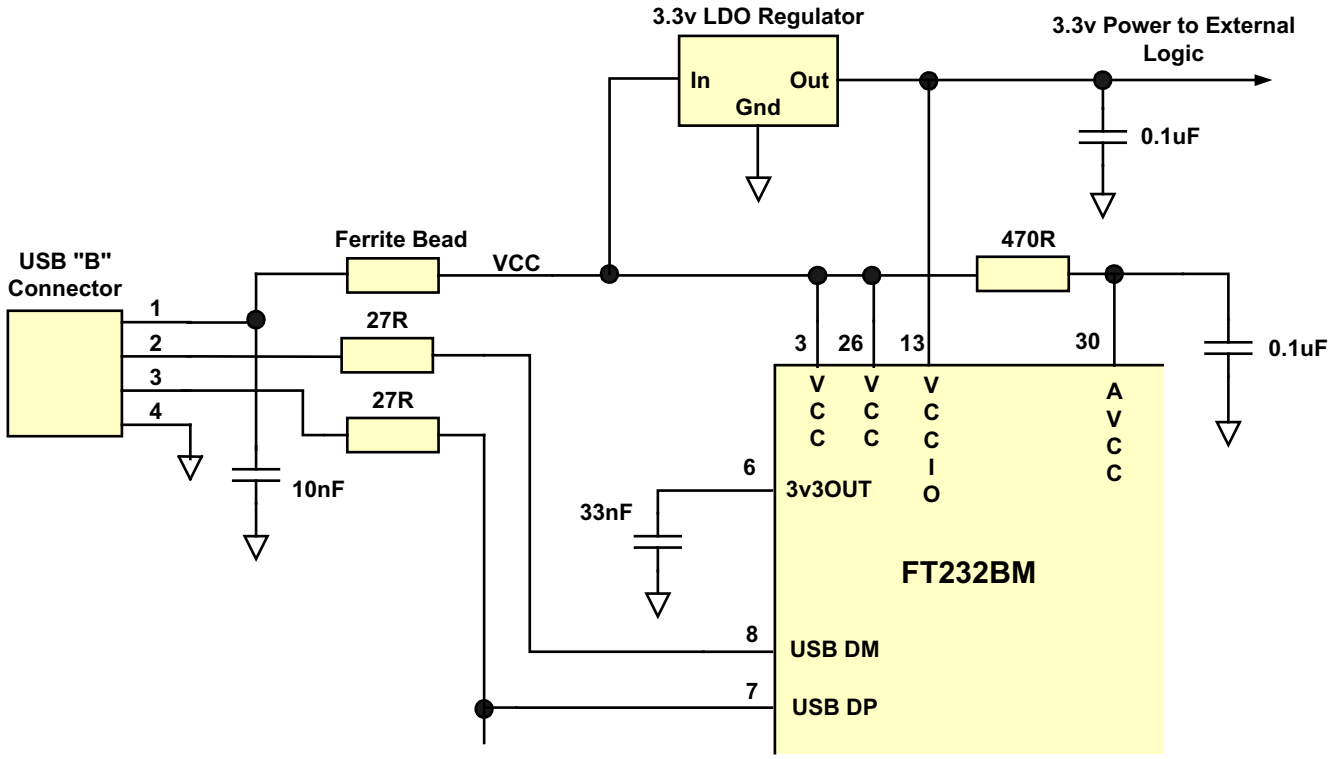


Figure 14 shows how to configure the FT232BM to interface with a 3.3V logic device. In this example, a discrete 3.3V regulator is used to supply the 3.3V logic from the USB supply. VCCIO is connected to the output of the 3.3V regulator, which in turn will cause the UART interface IO pins to drive out at 3.3V level. For USB bus powered circuits some considerations have to be taken into account when selecting the regulator –

a) The regulator must be capable of sustaining its output voltage with an input voltage of 4.35 volts. A Low Drop Out (LDO) regulator must be selected.
b) The quiescent current of the regulator must be low in order to meet the USB suspend total current requirement of <= 500uA during USB suspend.

An example of a regulator family that meets these requirements is the MicroChip (Telcom) TC55 Series. These devices can supply up to 250mA current and have a quiescent current of under 1uA.

In some cases, where only a small amount of current is required (< 5mA) , it may be possible to use the in-built regulator of the FT232BM to supply the 3.3v without any other components being required. In this case, connect VCCIO to the 3v3OUT pin of the FT232BM.

Note : It should be emphasised that the 3.3V supply for VCCIO in a bus powered design with a 3.3V logic interface should come from an LDO which is supplied by the USB bus, or from the 3V3OUT pin of the FT232BM, and not from any other source.

Figure 15
Self Powered Circuit with 3.3V logic drive / supply voltage

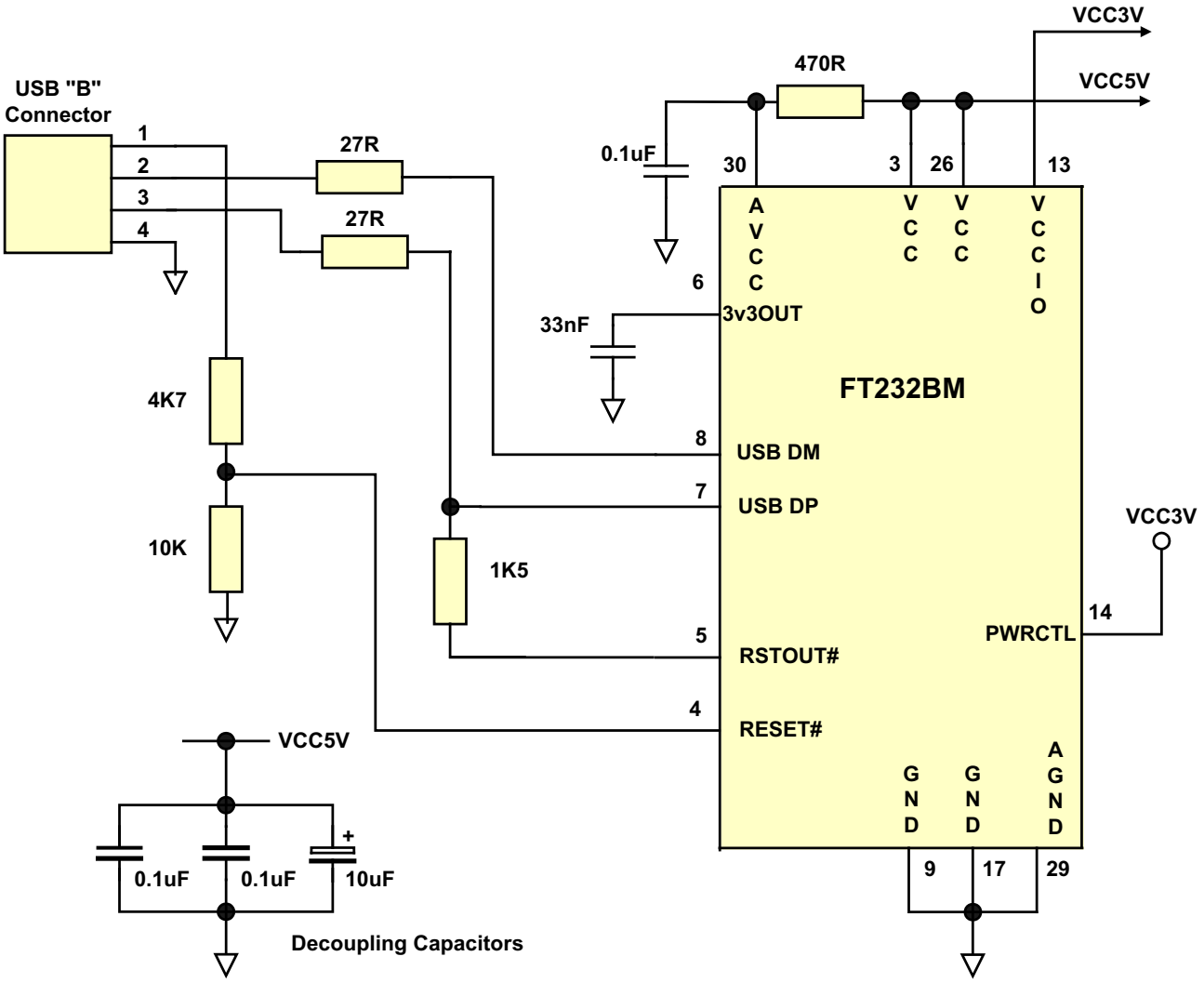


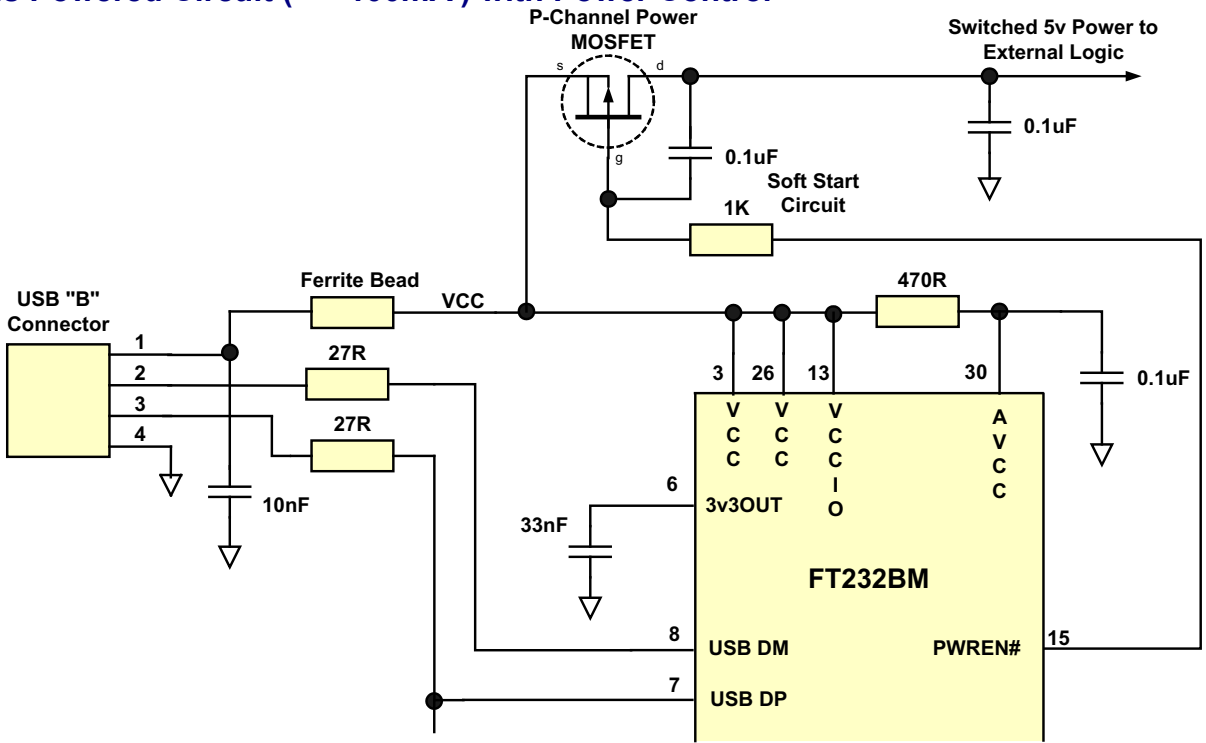
Figure 15 is an example of a USB self powered design with 3.3V interface. In this case VCCIO is supplied by an external 3.3V supply in order to make the device IO pins drive out at 3.3V logic level, thus allowing it to be connected to a 3.3V MCU or other external logic. A USB self powered design uses its own power supplies, and does not draw any of its power from the USB bus. In such cases, no special care need be taken to meet the USB suspend current (0.5 mA) as the device does not get its power from the USB port.

As with bus powered 3.3V interface designs, in some cases, where only a small amount of current is required (<5mA), it may be possible to use the in-built regulator of the FT232BM to supply the 3.3V without any other components being required. In this case, connect VCCIO to the 3v3OUT pin of the FT232BM.

Note that in this case PWRCTL is pulled up to VCCIO, not VCC.

7.7 Power Switching

Figure 16
Bus Powered Circuit (<= 100mA) 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). Some external logic can power itself down into a low current state by monitoring the PWREN# pin. For external logic that cannot power itself down in that way, the FT232BM provides a simple but effective way of turning off power to external circuitry during USB suspend.

Figure 16 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 International Rectifier IRLML6402, or equivalent. It is recommended that a “soft start” circuit consisting of a 1K series resistor and a 0.1 uF capacitor are used to limit the current surge when the MOSFET turns on. Without the soft start circuit there is a danger that the transient power surge of the MOSFET turning on will reset the FT232BM, or the USB host / hub controller. The values used here allow attached circuitry to power up with a slew rate of ~12.5 V per millisecond, in other words the output voltage will transition from GND to 5V in approximately 400 microseconds.

Alternatively, a dedicated power switch i.c. with inbuilt “soft-start” can 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 re-applied on coming out of suspend.
b) Set the Pull-down on Suspend option in the FT232BM’s EEPROM.
c) For USB high-power bus powered device (one that consumes greater than 100 mA, and up to 500 mA of current from the USB bus), the power consumption of the device should be set in the max power field in the EEPROM. A high-power bus powered device must use this descriptor in the EEPROM to inform the system of its power requirements.
d) For 3.3V power controlled circuits VCCIO must not be powered down with the external circuitry (PWREN# gets its VCC supply from VCCIO). Either connect the power switch between the output of the 3.3V regulator and the external 3.3V logic OR if appropriate power VCCIO from the 3V3OUT pin of the FT232BM.

8.0 Document Revision History

DS232B Version 1.0 – Initial document created 30 April 2002.

DS232B Version 1.1 – Updated 04 August 2002

- Section 4.1 RESET# Pin description corrected (RESET# does not have an internal 200k pull-up to VCC as previously stated).
- Figure 2 pin-out corrected (EECS = Pin 32).

DS232B Version 1.2 – Updated 27 October 2003

- Pin and package naming made consistent throughout data sheet.
- Section 1.0 Updated to reflect availability of Mac OS X driver.
- Section 2.0 Minor corrections.
- Section 3.1 Minor changes to functional block descriptions of SIE, RESET Generator, and EEPROM interface.
- Section 4.1 Note added to EEPROM interface group.
- Section 4.1 RSTOUT# Pin description amended.
- Section 6.1 Minimum operating supply voltage adjusted.
- Section 6.1 EESK added to Note 3.
- Section 6.1 UART IO pin characteristics amended.
- Section 6.1 RESET#, TEST, EECS, EESK, and EEDATA pin characteristics amended.
- Section 6.1 RSTOUT pin characteristics amended.
- Section 7.1 Updated recommended ceramic resonator part number and circuit configuration.
- Section 7.3 “USB Self Powered Configuration (1)” (original figure 8 removed). Recommended circuit for USB self powered designs updated. Subsequent figure numbers have changed as a result.
- Section 7.6 Note added to description of Bus powered circuit with 3.3V logic drive / supply voltage.
- Section 7.6 Self Powered Circuit with 3.3V logic drive / supply voltage added (new figure 16).

DS232B Version 1.3 – Updated 10 December 2003

- Section 5.0 Package drawing amended
- Section 6.0 Floor Life / Relative Humidity specification added. ESD and Latch Up specifications amended.
- Section 7.1 Required resonator / crystal accuracy corrected.

DS232B Version 1.4 – Updated 10 February 2004

- Grammar Correcttions
- Section 10.0 FTDI Address Updated
- Section 2.0 Extended EEPROM Support corrected
- Section 4.1 VCCIO Pin description amended.
- Section 7.4 RS485 Example Sipex SP481 part number corrected.

DS232B Version 1.5 – Updated April 2004

- Section 4.1 EESK Pin Description amended.
- Section 7.6 Figure 16 PWRCTL Pin number corrected.
- Section 7.7 Figure 15 PWREN# Pin number corrected.

DS232B Version 1.6 – Updated November 2004

- Section 1.0 WinCE drivers now available.
- Section 5.0 Date code format updated.
- Section 6.0 Absolute Maximum Ratings table reformated.

DS232B Version 1.7 – Updated February 2005

- Section 1.0 D2XX drivers for Linux and Windows CE now available.
- Section 5.0 FT232BL (lead Free) and FT232BQ (lead free QFN package) now available.

DS232B Version 1.8 – Updated December 2005

- Section 6.1 USB dataline Input voltage absolute maximum voltage added.
- Section 1.0 Driver OS support updated.

9.0 Disclaimer

© Future Technology Devices International Limited , 2005

Neither the whole nor any part of the information contained in, or the product described in this manual, 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.

Future Technology Devices International Ltd. 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.

10.0 Contact Information

Head Office -

Future Technology Devices International Ltd.
373 Scotland Street,
Glasgow G5 8QB,
United Kingdom

Tel. : +(44) 141 429 2777
Fax. : +(44) 141 429 2758

E-Mail (Sales) : **sales1@ftdichip.com**
E-Mail (Support) : **support1@ftdichip.com**
E-Mail (General Enquiries) : **admin1@ftdichip.com**

Regional Sales Offices -

Future Technology Devices International Ltd. (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.sales@ftdichip.com**
E-Mail (Support): **tw.support@ftdichip.com**
E-Mail (General Enquiries): **tw.admin@ftdichip.com**

Future Technology Devices International Ltd. (USA)
5285 NE Elam Young
Parkway, Suite B800
Hillsboro,
OR 97124-6499
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**

Website URL : **http://www.ftdichip.com**

Agents and Sales Representatives

At the time of writing our Sales Network covers over 40 different countries world-wide. Please visit the Sales Network page of our Web site for the contact details our distributor(s) in your country.

ООО «НИОКРсистемс» - это оперативные поставки широкого спектра электронных компонентов отечественного и импортного производства напрямую от производителей и с крупнейших мировых складов. Реализуемая нашей компанией продукция насчитывает более полумиллиона наименований.

Благодаря этому наша компания предлагает к поставке практически не ограниченный ассортимент компонентов как оптовыми, мелкооптовыми партиями, так и в розницу.

Благодаря развитой сети поставщиков, помогаем в поиске и приобретении экзотичных или снятых с производства компонентов.

Наша компания это:

- Гарантия качества поставляемой продукции
- Широкий ассортимент
- Минимальные сроки поставок
- Техническая поддержка
- Подбор комплектации
- Индивидуальный подход
- Гибкое ценообразование
- Работаем по 275 ФЗ

ООО «НИОКР СИТЕМС»
Москва, 125362, Россия, ул. Вишневая, д. 9, к. 1, Блок 104 а и 104 в

Телефон: 8 (495) 268-14-82
Email: n@nsistems.ru

ИНН: 7735154786
ОГРН: 1167746717709