

USB interface for PsoC

Objective

To implement a USB interface for the PsoC microcontroller.

Part Selection

The PsoC microcontroller does not have a built in USB interface. I needed to use USB for debug and other communication between the PsoC device and the PC. There were two solutions, I am familiar with from my past experience. It is assumed that the reader is familiar with the USB interface (specs. can be found at www.usb.org)

One was the EZUSB 8051 microcontroller from Cypress. This supported all four kinds of USB transfers and the 4-cycle, 48MHz packed a mean punch. The development kit downloadable from Cypress website includes all kinds of examples and drivers to get you going in a few hours. However the device itself cost more and I did not need all that flexibility and super power available. There are many other solutions from Cypress but I was not familiar with them.

So I went with the second solution. This is the solution from FTDI (www.ftdichip.com or www.ftdi.co.uk). They have 2 simple solutions. One is a rs232 to USB converter with speeds up to 1Mbps and a 384*8 bit FIFO to USB solution with speed up to 8Mbps. Drivers are provided free of cost for both solutions. There are two kinds of drivers for the rs232 to USB conversion. One maps the USB transfers to COM ports, this allows you to use existing serial communication programs without any changes to the PC program. The direct driver allows you to have direct access ,but needs more programming.

Working

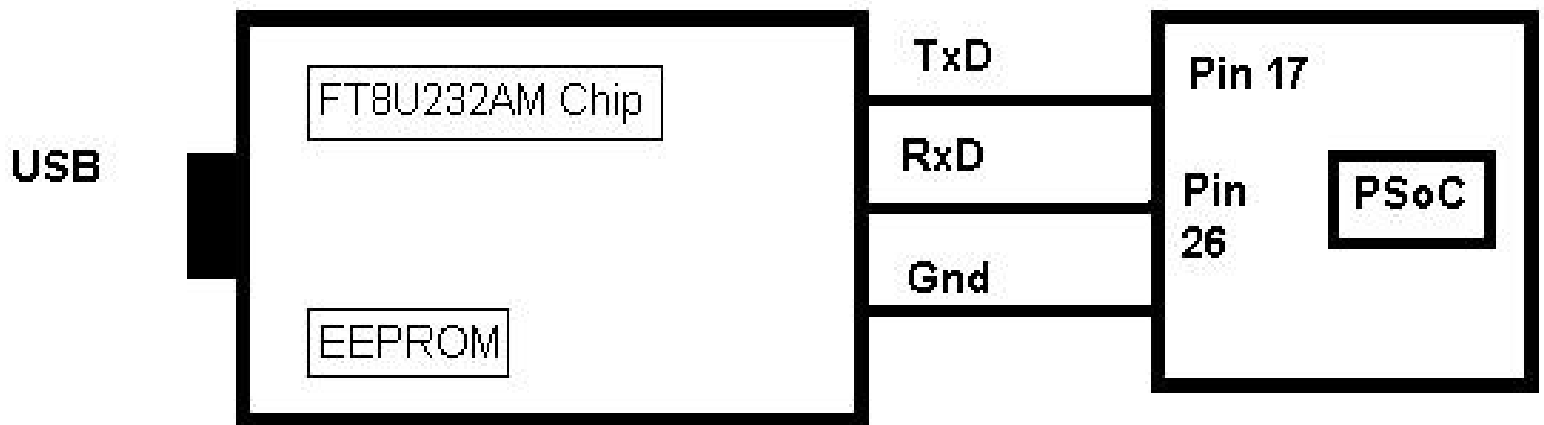
I went with the RS232 to USB converter chip **FT8U232AM** from FTDI. I was communicating with the PsoC from the PC using hyperterminal on com port 2 at 115K baud in less than 1hr flat. I had a working board from FTDI with the converter chip. I only had to bridge 3 wires (RxD, TxD and Gnd) from the PsoC over to that board.

The chip also supports non-standard baud rates. See the documentation at the FTDI website, there is a lot of support there. Since the IC supports 1Mbps the capability of the PSoC to operate the UART at >115Kbps can be exploited. Few microcontrollers support speeds above 115Kbps.

The picture below shows the arrangement needed, a FTDI board and a PsoC PUP board with ZIP socket for easier device insertion.



RS232 <> USB Converter board



The schematic for the project is shown above. The detailed schematic for the RS232 to USB converter board can be found at the FTDI website. The power for the whole system may be derived from the USB bus itself.

To get it working first a simple project was created in PsoC designer. This can be found attached as 'Project Number 246.zip'.

The project includes a UART module and a Counter8 module. The counter is used to provide the clock for UART module. This is obtained by dividing the Counter clock by the period value of the Counter. The Counter clock is obtained by dividing 24V1 by 1 (24V1). The UART module clock determines the baud rate . The clock must be at least 8 times the baud rate. The period value for the Counter is set to 25. Thus the baud rate of 115200 is obtained by dividing the Counter clock by period and then by 8 (i.e. $24V1 / 25 / 8$). This is all there is to hardware.

As for the firmware. The two modules are started at first. Then the program waits for a Carriage return on the RxD line or for 64 bytes. The received bytes are sent back on the TxD line for verification. The PsoC project is available in 'Project Number 246.zip'

Driver Installation

Now plug in the USB cable. The hardware will autodetect the device as the default FTDI converter. Direct the installation to the location of the drivers downloaded. The driver installation is now complete

Testing

Now open up a windows communication program like Hyperterminal. Since the driver is installed you will see an additional comm. port in the selection list,select this. Select the baud rate at 115K, 8 bits,no parity,1 stop bit. Save and reopen the setting.

Now type in a few characters and press 'Enter'. You should see those characters echoed back from the PsoC via the USB interface. The driver takes care of mapping the new comm. port to USB.

The FTDI chip allows user to use the 64 to 4K bytes 'bulk transfer' with automatic baud rate detection. They also provide a EEPROM programmer software to allow custom configuration table into the EEPROM. Consult the FTDI documentation for more details. The user may change for e.g.:, the PID and VID and also descriptor string into something that resembles the device like "PSOC device", so that the next time the device enumerates as your custom USB peripheral. This cab be verified in the windows device manager list under the USB section.

The setup provided very reliable communication for the PsoC device.