PC Interfacing

Fourth Level

Lecture Nine

USB to PIC Microcontroller Interface

Goals:
Up-on completing this lecture, the student should be able to:

1- Understand the main concept behind the USB
2- Comprehend the role of MCUs with USBs.
3- Build interfacing ccts for MCUs with USBs.

Universal Serial Bus (USB):

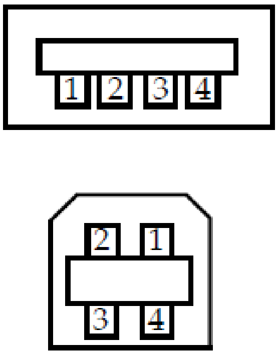
USB (Universal Serial Bus) is the most popular connection used to connect a computer to devices such as digital cameras, printers, scanners, and external hard drives. USB is a cross-platform technology that is supported by most of the major operating systems. On Windows, it can be used with Windows 98 and higher. USB is a hot-swappable technology, meaning that USB devices can be added and removed without having to restart the computer. USB is also “plug and play”. When you connect a USB device to your PC, Windows should detect the device and even install the drivers needed to use it. There are many versions of USB. The original version of USB, USB 1.0, only supported speeds of up to 11 Mbps and was used mostly to connect keyboards and mice. The second version of USB, which is known as USB 2.0, supports speeds of up to 480 Mbps. The third version of USB, which is known as USB 3.0, supports speeds of 5 Gbps. The last version of USB, which is known as USB 3.1, supports speeds of up 10 Gbps.

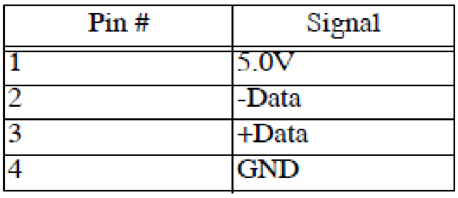
- Allows access of up to 127 different connections via a 4 wire serial connection.
- This interface is ideal for keyboards, sound cards, modems, etc.
- Sound cards can derive their power from an external (no-PC) power supply.
- Cable lengths are limited to 5 meters (for the full-speed interface).
- Maximum power is given by 100mA x 5V.

USB Connector:

It has 4 pins:

- 5 v
- -Data
- +Data
- Ground





Pin #	Signal
1	5.0V
2	-Data
3	+Data
4	GND

The +/-Data signals are 180 degrees out of phase.

Biphase signals: when +data are at 5.0V, -data are at zero volts and vice versa.

Physical Interface:

Full Duplex: data transmission can occur in both directions at the same time

Half Duplex: data transmission can go in only one direction at a time

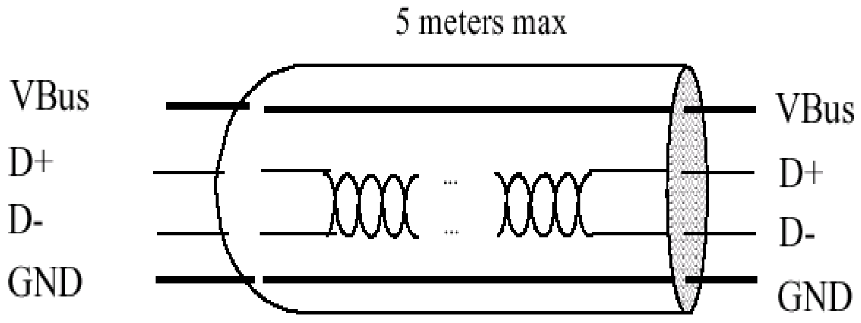


Figure 4-2. USB Cable

USB to PIC interfacing:-

It’s a low cost USB interface Board that provides very good interface to your computer and it can be used to control various devices like DC Motor, Stepper motor, Servo, relay switch etc. A small program communicates with the hardware to issue commands to set the pins of the Microcontroller. USB controlling is better than that of parallel and RS232 ports. Rather laptops now days don't come with parallel and RS232 ports.

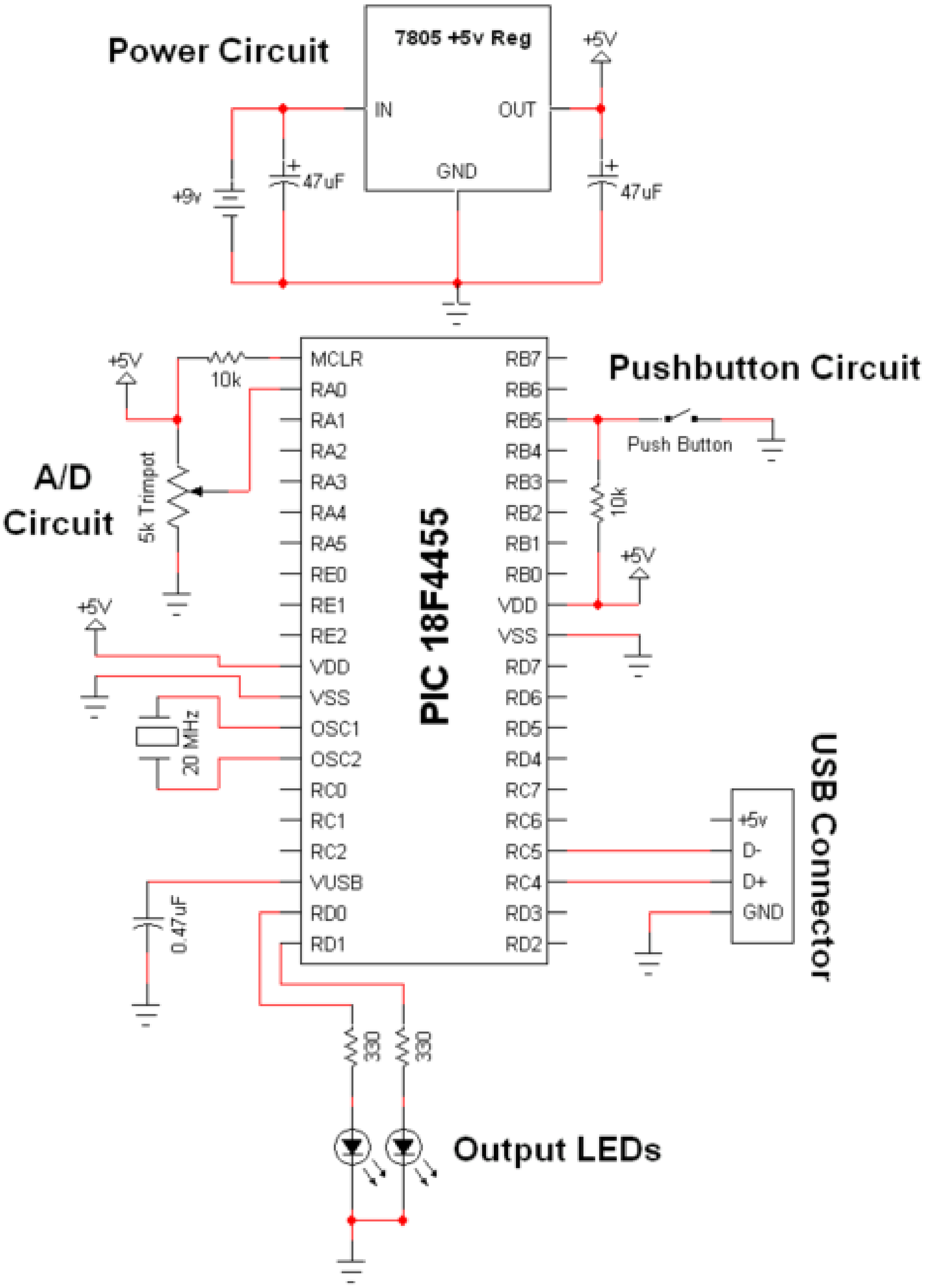
Parts

- PIC 18F4455 Microcontroller
- USB Type A Connector
- USB Cable
- 7805 +5v Regulator
- 2x 330Ω Resistors
- 2x 10kΩ Resistors
- 5kΩ Trimpot
- 2x 47uF Capacitor
- 20 MHz Crystal
- 1x 0.47uF or 0.1uF Capacitor

- Breadboard
- Jumper Wire
- 9v Connector
- Push Button Switch
- 2x LEDs

Schematic Overview

Hardware design for USB is actually quite minimal, which is a big plus for us. However, what you quickly find out with USB is that the easy hardware design means the communication and control software is very complex. The main devices used in the circuit are the PIC18F4455, USB Connector and LM7805.



Schematic Specifics:-

Power Circuit

The +5v output from the power circuit comes from the LM7805 regulator. Notice the 47uF capacitors on the input and output. These are meant to be DC filtering capacitors, which smooth out the constant DC voltage being fed to the microcontroller from the 7805 regulator.

USB Connection and Output LEDs

Make sure you double check your USB pin out. A common mistake when wiring the PIC to the USB connector is getting the D + and D - signals backwards. So if you're sure that the PIC is running your perfect code, but the USB device isn't coming up properly, switch D+ and D-, it might just magically fix your problem! The output LEDs will be simple 'toggle' LEDs. The program running on our laptop will be able to toggle them on and off with the push of a button.

A/D and Push Button Circuits

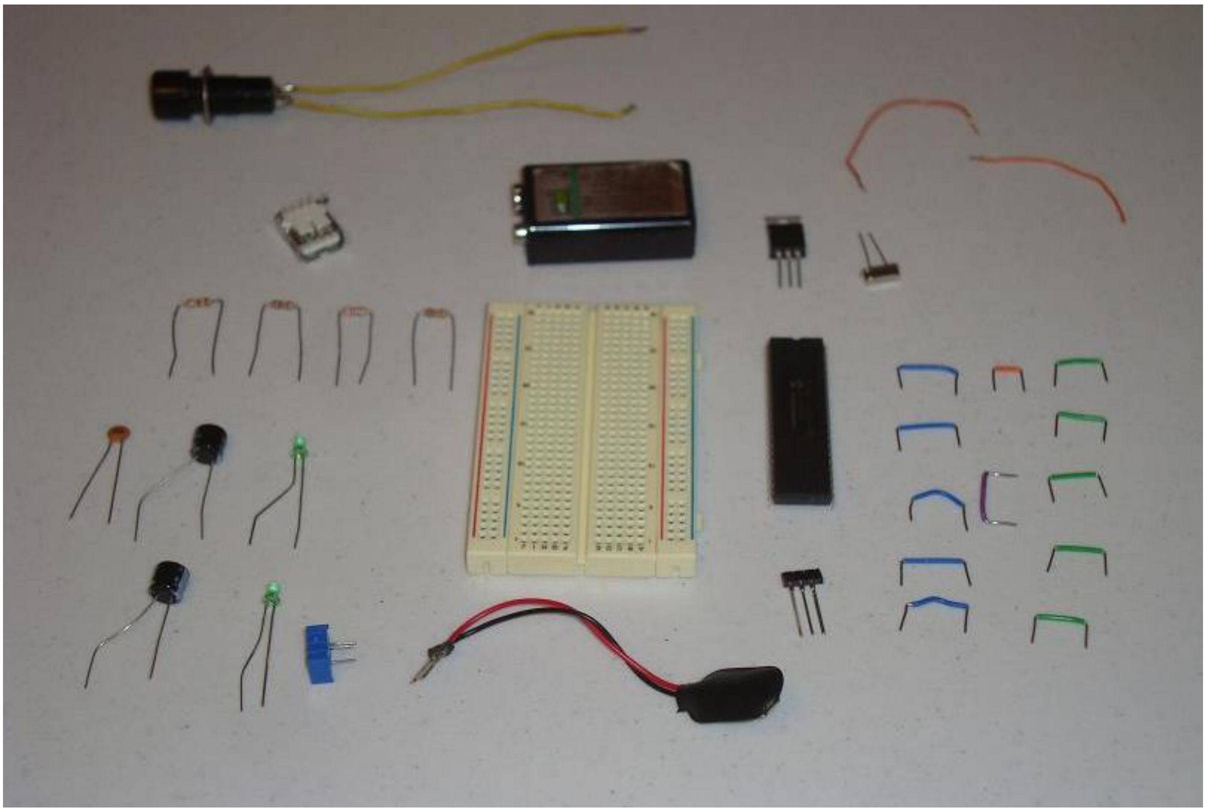
The A/D circuit is a standard 3 pin, Connected to Power, Signal Out and Ground circuit. The signal output goes into RA0 which is the Analog to Digital converter. After the PIC converts this signal it should send the data to the Laptop via USB. The laptop will the visually display the trimpot's value. The push button will do a similar thing, when the button is pushed, the laptop application should update with a notification that it has been pressed. These are simple ideas, but when done over USB they become rather complicated as we'll see in the theory section.

Hardware Design

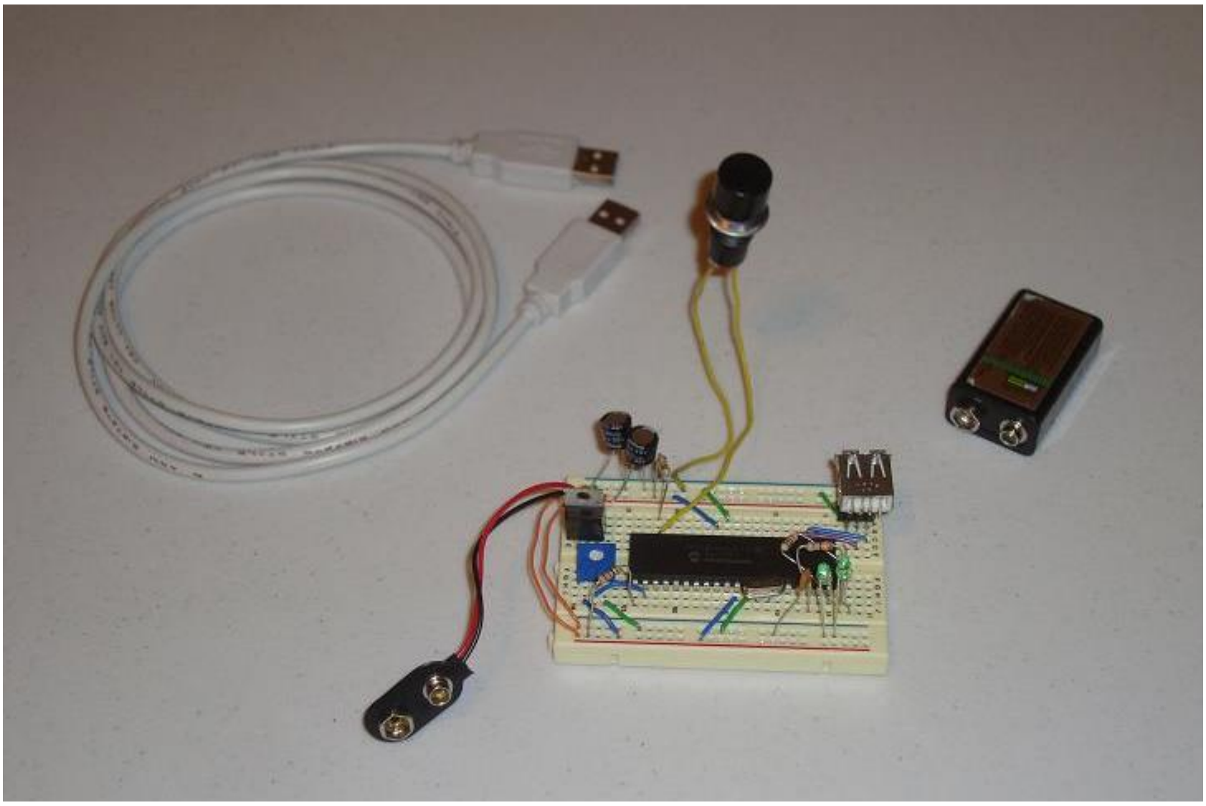
When it comes to USB, the hardware design is as simple as it gets. A few wires and the external input/output parts are all that you need. However, this ease, is made up for by the complexity of the protocol seen in the theory section, and in the software.

Building the Circuit

Here I have all the parts used in this project laid out for you to see. They were also listed in the parts section of this write-up.



Now, we start assembling the circuit, and connecting it together, we obtain:



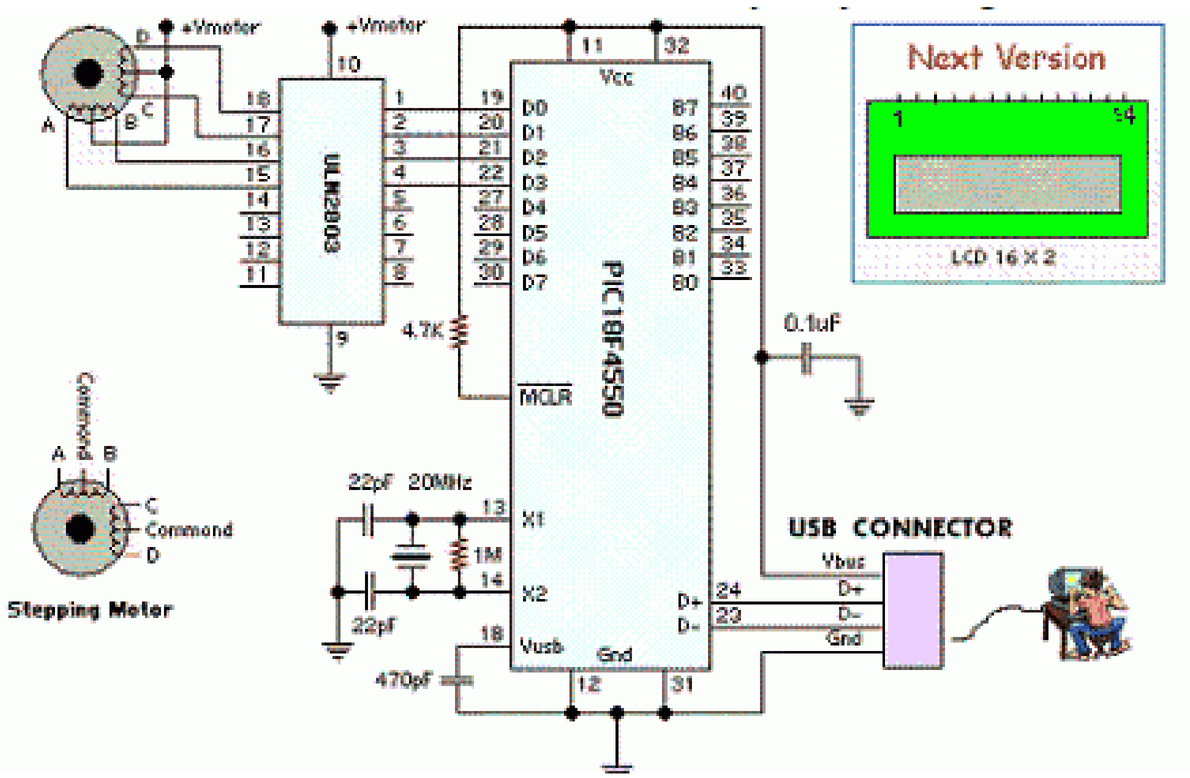
Software

Now since the hardware is ready we have to write the software for the PIC microcontroller. The different compiler vendors provide different ways to setup the PIC.

Control stepper motor via USB:

How to control stepper motor device via USB interface?

Simple solution is to take microcontroller with USB interface built in. PIC15F4550 has full speed USB interface with full speed capability. Microcontroller uses on chip USB driver which is pretty easy to program.



You may want to edit source in order to interface different devices like LCD, relays etc.

Summary:

1- USBs play a crucial rule in interfacing
2- An MCU with a USB hugely changes the design possibilities
3- Most of today's modern computers are equipped only with the USB.

Questions:

1- Design a cct to read the temperature from an LM35 sensor to the PC using the USB port.
2- What are the models and pin layout of the USB port?