Re: PIC/Linux

Source: http://sci.tech−archive.net/Archive/sci.electronics.misc/2006−11/msg00085.html

- *From*: Chris Jones <lugnut808@xxxxxxxxxxxxxxx>

- *Date*: Tue, 14 Nov 2006 00:06:26 +0000

Allan Adler wrote:

> Suppose I want to learn to use a PIC and only have a PC running RH7.1
> Linux. In particular, I don't have a PIC development board. I hesitate to
> get anything until I know exactly what I need and how much it will
> ultimately cost. If you know exactly what I need to get to do this, and
> what it will cost, please let me know. I assume I need:
> (1) A PIC development board. Which one and what will it cost, given that
> it has to talk to my PC running RH 7.1 Linux?
> (2) Software, to talk to the PIC development board, that will run on my
> PC running RH 7.1 Linux. What exactly do I get and from where?
> (3) Reading materials that will give me certainty of learning to actually
> use the PIC development board once I have the PIC development board
> and the software running under RH 7.1 Linux. What exactly do I read
> and what does it cost?
>
> I am very easily thrown by little details that don't go right, since I
> don't have time to devote myself completely to solving them. So, I need
> for this
> undertaking to be easy and without surprises. Advice such as "read any
> book for dummies on PIC and make the necessary adjustments for Linux"
> is likely to lead to failure in my case. Once I've done this once, it will
> probably be a different story, but I need to suceed in a basic situation
> and then build on it.
>
> Also, I have very little discretionary capital; that's why I'm using an
> old PC running RH 7.1 in the first place. So the more inexpensive the
> solution, the better. (Please don't refer me to ebay, since I can't deal
> with ebay.)

I would avoid buying a development kit. If you use e.g. a PIC16F84A, then I have had success plugging those into a "solderless breadboard" (white things, about two inches by six inches and quarter inch thick, that have an array of little holes in them connected together in groups of five.) Just make sure that you put a decoupling cap very close to the PIC, and keep the

wiring of the crystal oscillator absolutely as short as you can to minimise capacitance.

To program it I used something based on this schematic:
ftp://ftp.armory.com/pub/user/rstevew/PIC/DaveTait/pp.gif
I had to add some small resistors (1kOhm) in series with the data lines from the printer port and small capacitors to ground (1nF) as a filter on these lines because I was getting glitches due to coupling between the wires of the printer cable, but if you use a very short cable then that may not be a problem for you.

I can highly recommend the PIC C compiler from HiTech, of which there is a free (but closed source) version for Linux, (though I only used the DOS one many years ago):
http://www.hitech.com.au/downloads/demos.php

In the end I became a bit frustrated with the small amount of RAM on these old PIC chips and I will one day have to get around to figuring out how to use something with more memory (but hopefully not too many pins − I don't want to have to make a PCB for very project I undertake.)

Chris
.