Re: PIC/Linux

Source: http://sci.tech−archive.net/Archive/sci.electronics.misc/2006−11/msg00055.html

- *From*: Allan Adler <ara@xxxxxxxxxxxxxxxxxxx>
- *Date*: 10 Nov 2006 00:00:59 −0500

Frank−Christian Kruegel <dontmailme@xxxxxxxxxxxx> writes:

> Is there a reason why you choose PICs?

Not really.

> Atmel AVR microcontrollers are
> similar or even a bit better in terms of features, pincount, memory, speed,
> and they are easier to program. There is a gcc port for AVR controllers
> (yes, it's the same gcc you'd use on your Redhat system, but this one spills
> out code for AVR instead of x86). There are free programming tools like
> avrdude for programming the controllers via their ISP interface.
> So AVR is a better choice for you.

I considered AVR earlier and tried to find some materials on it that didn't
assume a Windows environment. The stuff I downloaded contained the following
files with no documentation, and also looked BSD specific, so I gave up:
AT90CAN128.BSD ATMEGA169.BSD atmega323.bsd ATMEGA329.BSD ATMEGA6490.BSD
ATMEGA128.BSD ATMEGA16.BSD ATMEGA3250.BSD atmega32.bsd ATMEGA649.BSD
ATMEGA162.BSD ATMEGA2560.BSD ATMEGA325.BSD ATMEGA6450.BSD atmega64.bsd
ATMEGA165.BSD ATMEGA2561.BSD ATMEGA3290.BSD ATMEGA645.BSD

> Start reading here:
> http://www.linuxjournal.com/article/7289
> http://cdk4avr.sourceforge.net/
> http://www.tuxgraphics.org/electronics/200411/article352.shtml
> Mit freundlichen Gruessen

Vielen Danke. I looked at these links and will have to read the first and
last a few more times before I can get an overview. Meanwhile, I went to
the sourceforge link and downloaded everything that I couldn't prove I
didn't need, except when there were long sublists of downloads for very
specific interfaces, etc. Here is what I wound up with:

cdk−avr−atmelprg−20060228.tgz
cdk−avr−ava−0.3b_0815_E.tgz
cdk−avr−avarice−2.4.tgz
cdk−avr−avra−1.1.1−20060912.tgz
cdk−avr−avrctrl−isp−20020506.tgz
cdk−avr−avrdis−0.01.tgz
cdk−avr−avrp−1.0.tgz
cdk−avr−AVRprog−20060215−20060626.tgz
cdk−avr−avrprog−pzn−0.2.1.tgz
cdk−avr−base−0.5.tgz
cdk−avr−base−libconfuse2−devel−2.5.tgz
cdk−avr−base−libftdi0−devel−0.7.tgz
cdk−avr−binutils−2.16.1−20060708.tgz
cdk−avr−cisp−1.0.4.tgz
cdk−avr−example−butterfly−0.6.6.tgz
cdk−avr−example−simulavr−ddd−1.0−20060709.tgz
cdk−avr−gavrasm−2.0−20061001.tgz
cdk−avr−gcc−3.4.5−20060708.tgz
cdk−avr−gdb−6.4.tgz
cdk−avr−geany−0.9−20061001.tgz
cdk−avr−jtagice−0.0.3.tgz
cdk−avr−libavr−20051024.tgz
cdk−avr−libavrhal−20030717.tgz
cdk−avr−libc−1.4.4.tgz
cdk−avr−picoweb−pppt−20030107.tgz
cdk−avr−revava−0.4.tgz
cdk−avr−simulavr−0.1.2.2−20060709.tgz
cdk−avr−sp12−2.1.0.tgz
cdk−avr−tavrasm−1.22.tgz
cdk−avr−uisp−20050207.tgz

I didn't download the separate doc files. If it turns out that I need them,
I can always go back.

Since I don't possess an AVR and don't know yet exactly what to get, I think
I need to try to figure out which of these tgz files to open and explore.
For example, cdk−abr−simulavr−0.1.2.2−20060709.tgz, by its name, sounds as
though it will let me simulate the AVR without actually having one.

The third link, to tuxgraphics.org, is fairly specific about binutils−2.15,
while I have binutils−2.10, and gcc−core−3.42, while I have gcc−2.96. So,
I'm not sure yet what ought to work on my RH 7.1 Linux PC. They also sell
some hardware at various prices in Euros, any one of which might represent
approximately the total of what I'm prepared to spend on hardware for this
learning experience, and I don't know how many items I might actually need.
I do know that I can't buy anything until I have very detailed knowledge of
exactly what I will need in the way of hardware, software, documentation
and other educational materials.
−−
Ignorantly,

Allan Adler <ara@xxxxxxxxxxxxxxxxxxx>
* Disclaimer: I am a guest and *not* a member of the MIT CSAIL. My actions and
* comments do not reflect in any way on MIT. Also, I am nowhere near Boston.
.