Building your own avr-gcc environment with atmega328p capabilities, Linux

Abstract:

Atmel announced the atmega328p chip a while ago but it is only now that one can really order such chips. The atmega328p is the big brother of atmega88 and atmega168. It is fully pin compatible and provides 2Kb ram and 32Kb flash. Especially the larger amount of ram makes it very attractive.

You can load code compiled for atmega168 into an atmega328p and it will work but you will not be able to use the extra ram and flash memory. To use the full capabilities of the new chip a new compiler is needed.

Installing avr-gcc under Linux

Linux is a great system for software development. It is very easy to add a full avr-gcc development environment.

There is a chapter for Mac and windows at the end of this article.

The installation is very straight forward. Just follow the commands below. We will install the following versions:

```
binutils-2.19
gcc-4.3.2
avr-libc-1.6.4
avrdude-5.5
```

I have verified that those packages are working together. The final installation will give you a compiler that supports the atmega328p chip. Everything will be installed under /usr/local/avr You should add the directory /usr/local/avr/bin to your Unix search PATH environment variable after completion of the installation.

Installation step by step

Just download the files and run the commands one by one in the order shown below. The commands are in bash syntax.

Besides a C-compiler with gmake and basic unix commands you need also: texinfo, bison, flex, gawk, readline and ncurses.

```
You should switch now to bash unless you use bash already as
your default shell. Type:

bash
```

binutils-2.19

Download: ftp://ftp.gnu.org/gnu/binutils

```
tar jxvf binutils-2.19.tar.bz2
cd binutils-2.19
mkdir obj-avr
cd obj-avr
CC=gcc
export CC
../configure --target=avr --prefix=/usr/local/avr --disable-nls --enable-install-libbfd
make

make install

cd ../..
```

gcc-core-4.2.3

Download: ftp://ftp.gnu.org/gnu/gcc/gcc-4.3.2

```
tar jxvf gcc-core-4.2.3.tar.bz2
cd gcc-4.3.2
mkdir obj-avr
cd obj-avr
../configure --target=avr --prefix=/usr/local/avr --disable-nls --enable-languages=c --di
make

make install

Note: The above step requires a recent version of MPFR (http://www.mpfr.org/)
and gmp (ftp://ftp.gnu.org/gnu/gmp):  GMP 4.1+ and MPFR 2.3.0+.
Use the packages from your distribution if possible. Under Ubuntu
those packages are called libgmp3-dev and libmpfr-dev. Gentoo calls
```

them dev-libs/gmp and dev-libs/mpfr.

```
cd ../..
```

avr-libc-1.6.4

Download: http://savannah.nongnu.org/projects/avr-libc/

```
tar jxvf avr-libc-1.6.4.tar.bz2
cd avr-libc-1.6.4
PREFIX=/usr/local/avr
export PREFIX
CC=avr-gcc
export CC
PATH=/usr/local/avr/bin:${PATH}
export PATH
./configure --build=`./config.guess` --host=avr --prefix=/usr/local/avr
```

now check if configure printed the line:
 checking if avr-gcc has support for atmega328p... yes
just to be sure that you will get atmega328p support

```
make

make install

cd ..
```

avrdude-5.5

Download: http://savannah.nongnu.org/projects/avrdude

```
tar zxvf avrdude-5.5.tar.gz
cd avrdude-5.5/
CC=gcc
export CC
./configure --prefix=/usr/local/avr

make

make install
```

Now the installation is complete. It's a good idea to add some version information file so you know later on what you installed. Go back one level such that you can see the downloaded packages and their unpacked directories and then type ls:

```
cd ..
ls > /usr/local/avr/version-info.txt
```

Manual changes to avrdude.conf

The avrdude.conf file was installed during the above procedure in /usr/local/avr/etc. It needs some manual editing.

You need to add an atmega328 section (click here) as avrdude does not yet have atmega328 support in the standard configuration file. Search for atmega168 and add it before or after.

To use avrusb500 with avrdude go to the beginning of the avrdude.conf file and add this:

```
default_serial     = "/dev/ttyUSB0";

programmer
  id    = "avrusb500";
  desc  = "tuxgraphics avrusb500";
  type  =  stk500v2;
;
```

avr-libc incompatibilities

The io.h for atmega168 contains pin definitions which have been renamed in atmega328. Those will cause compile errors. In case of the tuxgraphics ethernet code you will find that you will have to change the following in the source code:

```
change PB1 to PORTB1
change PB3 to PORTB3
change PB5 to PORTB5
change PD7 to PORTD7
```

We will update the code on our website over time too but it might take a while.

Installing avr-gcc for BSD unix, Mac and windows

- FreeBSD:
  - The procedure is identical to the above description
- Mac:
  - Get the XCode tools from Apple and then follow the above instructions
  - You can also get pre-compiled versions known as avrmacpack: http://www.obdev.at/products/avrmacpack/index.html
  - Instructions for the mac with screenshots can be found at: http://www.ladyada.net/library/avrdevtut/setup-mac.html
  - The avrusb500 programmer needs the virtual com-port driver from http://www.ftdichip.com/FTDrivers.htm . The avrusb500 shows then up as a usbserial device in /dev. Typing "ls /dev/*usbserial*" will probably find the right device. There is always some funny number behind the usbserial which make it unique (in case you happen to have many of them ;-).
    To read e.g the fuse bytes of an atmega168 chip you could write a two line shell script called "readfuse168":

    ```
    #!/bin/sh -x
    avrdude -p m168 -P /dev/tty.usbserial* -c stk500v2 -v -q
    ```

    Please edit this script as needed and insert the correct usbserial name. Avrdude compiles without any problems on the Mac.

- Windows:

- Your best bet is to get a pre-compiled version at: http://winavr.sourceforge.net/index.html

  All tuxgraphics software comes with a Makefile. Most of our software consists of more than one .c file therefore remember to use the supplied Makefile during compilation.

  A problem of windows is the lack of a decent shell environment and a command interface. In Winavr you can use a batch file to setup the environment while calling the make-utility. An example of such a script is shown below. Put the batch file in the same directory where the code is and call it something.bat . To compile just double click on it form the file manager. It is also possible to call that script from avr-studio when using avr-studio. Note that avr-studio is as such is a avr assembler programming environment. To program in C you need the avr-gcc compiler.

Here is a batch file to compile C-code under windows. Please adapt the path inside the script according to your avr-gcc installation.

```
REM *** you need to edit this file and adapt it to your WinAVR
REM *** installation. E.g replace c:\avrgcc by c:\WinAVR-20090313
@echo -------- batch file for windows to call make --------

set AVR=c:\avrgcc

set CC=avr-gcc

set PATH=c:\avrgcc\bin;c:\avrgcc\utils\bin

make -f Makefile

@echo --------  end  --------
pause
```

References/Download

- The atmega328 is available in our online shop: shop.tuxgraphics.org

2009-12-30, generated by tuxgrparser version 2.57