Pervasive Computing Testbed

From MPC Wiki

Contents

iRobot Create Setup

Step-by-step How to Connect to Gumstix Over Serial Using Kermit

Adapted from Gumstix Wiki - Linux Serial Instructions (http://docwiki.gumstix.org/Connecting_via_Serial_-_Linux)

- remove brltty package (an Ubunutu specific fix)
- place .kermrc with config options in home directory
  - .kermrc adapted from gumstix wiki available in pctb svn repository
  - we are using USB serial adapter, so use ttyUSB0
- power on gumstix
- start kermit
- type "connect"
- then, prompt for login pops out
  - if not, try hitting enter
- ctrl-\ (and then) C to get back out to Kermit
- "connect" again to get back to gumstix

Step-by-step How to Install Buildroot

From Gumstix Wiki- Buildroot (http://docwiki.gumstix.org/Buildroot)

- type "svn co -r1161 http://svn.gumstix.com/gumstix-buildroot/trunk gumstix-buildroot" to get correct revision of Buildroot
  - we use svn revision 1161 (which came pre-installed on the gumstix), because other revisions may

or may not work

- type "svn co svn+ssh://bgrant@frisbee.ece.utexas.edu/svn/pctb pctb" to check out relevant repository code
- need to install ncurses, or you have to choose all the defaults
- install and uninstall some stuff from buildroot to save space
  - Toolchain Options
    - add build/install c++ compiler and libstdc++ -- required for player
  - Package Selection
    - remove bluez -- to prevent bluetooth error messages since we don't have bluetooth hardware
    - remove boa -- to save space
    - remove bonjour -- to save space
    - add libtool -- required for player
- type "make" in gumstix-buildroot directory
  - this takes about 35 mins on a dual core 1.83GHz laptop with a good internet connection
  - downloads all the relevant packages, constructs the filesystem for the gumstix in build_arm_nofpu/root, and builds the filesystem image from it (the .jffs2 file)
- type "cd toolchain_build_arm_nofpu/uClibc-0.9.28"
- make menuconfig
  - Networking Support
    - add Remote Procedure Call (RPC) support
      - add Full RPC Support
- go back to gumstix-buildroot
- type "grep RPC toolchain_build_arm_nofpu/uClibc-0.9.28/.config"
  - you should see:

```
UCLIBC_HAS_RPC=y
UCLIBC_HAS_FULL_RPC=y
```

- type "cp toolchain_build_arm_nofpu/uClibc-0.9.28/.config target/device/Gumstix/basix-connex/uClibc.config"
- type "rm toolchain_build_arm_nofpu/uClibc-0.9.28/.configured"
- type "make"
- when you're finished, you can double check to ensure that the xdr routines are present by doing:

```
cd build_arm_nofpu/staging_dir/lib
../bin/arm-linux-nm libc.a | grep xdr_bytes
```

and you should see:

```
U xdr_bytes
00000594 T xdr_bytes
```

Step-by-step How to Install Player Server Into Gumstix Filesystem

From Player Installation Instructions (http://playerstage.sourceforge.net/doc/Player-

2.0.0/player/group__tutorial__crosscompiling.html)

- download the player source package (make sure it's a stable version, we've tried 2.0.4 and 2.0.5)
- go to the unpacked directory
- make sure the cross compiler tools are in your path
  - they should be in gumstix-buildroot/build_arm_nofpu/staging_dir/bin
- type " ./configure --build=x86-linux --host=arm-linux --disable-shared --disable-alldrivers --enable-roomba --prefix=[path to buildroot]/build_arm_nofpu/root/usr"
  - where [path to buildroot] is of course where your own path to gumstix-buildroot
- type "make"
- type "make install"

Step-by-step How to Create Filesystem Image and Flash it to Gumstix

From Gumstix Wiki - Replacing the Filesystem Image (http://docwiki.gumstix.org/Replacing_the_filesystem_image) , Basix and Connex, pre-1326 section

- copy roomba.cfg into [buildroot]/build_arm_nofpu/root/root/
- copy player server init script into image (in pctb svn repo)
- edit the etc/network/interfaces file (also in pctb svn repo)

- type "cd [buildroot]"
- type "make"

- connect to the device with kermit:

```
run kermit
```

At kermit prompt:

```
kermit> connect
```

- turn on / plug in the Create
- stop the autoboot by pressing a key

- then flash the filesystem:

```
GUM>loadb a2000000
(CTRL-\ C to get back to the kermit prompt, usually)
kermit>send rootfs.arm_nofpu.jffs2
kermit>connect
GUM>protect on 1:0-1 && erase all && cp.b a2000000 40000 ${filesize}
(wait...)
GUM>boot
```

The Remains of Bobby's Project

- an initial dump of files (the results of Robert Grant's UbiComp project are in the svn repo at svn/pctb/irobot-create-setup. These files include:
  - a report (with lots of helpful references)
  - configuration files for the roomba and player
  - a player client written in python
  - linux config files for the gumstix
  - a kermit config file
  - a start on a step-by-step node procedure which we should wiki-ize and fill out (and correct)

Helpful Links

A pretty complete list of references is attached to Bobby's report (and a .bib file is also there for those who need it). However, the most helpful ones are these.

- The Robotics Primer Workbook (http://roboticsprimer.sourceforge.net/workbook/Main_Page) - Incomplete, but a decent starting place. Instructions for building the serial cable to interface with the iRobot Create are here.
- The Gumstix Documentation Wiki (http://docwiki.gumstix.org) - This is the best (most likely to be accurate) resource for all things Gumstix and Buildroot. Favor these instructions over any from the Robotics Primer or the University of Alabama.
  - Buildroot (http://docwiki.gumstix.org/Buildroot)
  - Buildroot on Ubuntu (http://docwiki.gumstix.org/Buildroot_on_Ubuntu)
- The University of Alabama Distributed Autonomy Lab (http://robotics.cs.ua.edu/wiki/index.php/Main_Page) - A project using the Gumstix stack and the iRobot Create in a Robotics class. This is where we got our method for powering the Gumstix from the Create's on-board battery. They have decent instructions for compiling the Player software and Buildroot, but their hardware is slightly different from ours, so you can't use their instructions verbatim.