6.095 Humanoid Robot Competition

Massachusetts Institute of Technology

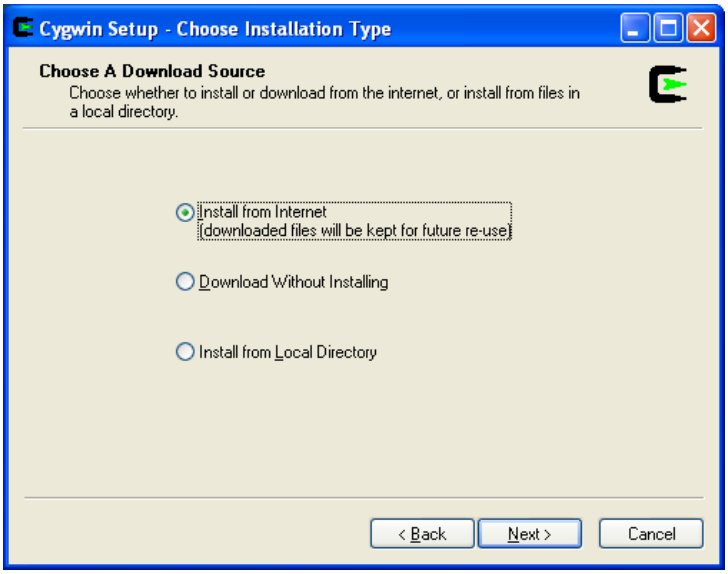
Software Installation

Overview

This handout walks you through the steps to install the software needed to run the robot. By the end, you will download a program to the robot: test_and_calibration. This program will allow you to set specific servos to determined angles. You need to do this before finishing the assembly of the robot. Since the software installation can be time consuming, it is best if a teammate starts the software installation while the other continues the hardware assembly.

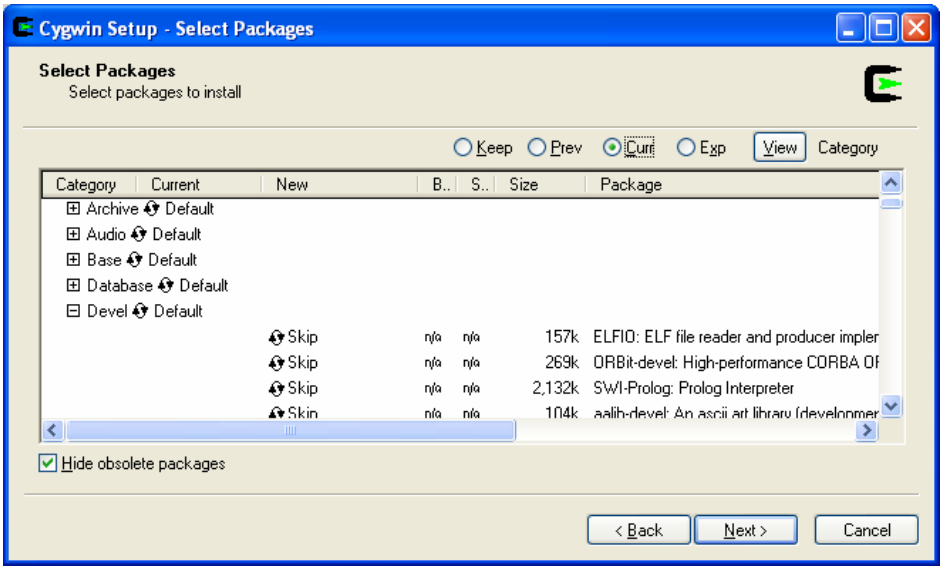
1. Installing cygwin

Go to **http://cygwin.com/**, download and run cygwin's **setup.exe**. Choose the option to install from the internet.



Then continue the setup with the default settings, until reaching the **Select Packages** window. In this window, you have to make sure you are installing **gcc** (the C compiler) and **make**. To do this, expand **Devel** and make sure the following packages are selected:

- gcc-core: C compiler
- make: The GNU version of the 'make' utility



Finish the setup making sure to create an icon on the Desktop.

Commands in Cygwin

Cygwin behaves as most Unix systems. Tasks such as copying or deleting files are achieved with commands. This is a list, for future reference, of basic commands that can be used in Cygwin.

- `pwd` prints the current directory
- `cd <subfolder>` moves to the subfolder with that name
- `cd ..` moves up one folder
- `ls` displays the content of the current folder
- `cp <targetfile> <targetdirectory>` copies the target file into the target directory
- `mkdir <directorypath>` creates a new directory
- `tar <targetfile>` uncompresses the target file
- `make` executes the Makefile in the current path

2. Downloads

Download all the files in **http://courses.csail.mit.edu/6.095/current/software/**
Locate all files in cygwin's home directory. The download should contain these files:

Makefile
binutils-2.16.1.tar.gz
gcc-core-3.4.6.tar.gz
newlib-1.9.0.tar.gz
lib.tgz

ttermp23.zip
test_and_calibration.rar

3. Compiling cross compiler for mips CPU

Run cygwin and write these commands. Do not hesitate to call a TA if you see an error message.

binutils (version 2.16.1)

```
$ mkdir /usr/local/src
$ cd /usr/local/src
$ tar xvfz ~/binutils-2.16.1.tar.gz
$ mkdir binutils_robin
$ cd binutils_robin
$ ../binutils-2.16.1/configure --target=mipsel-robin-elf --
prefix=/usr/local
$ make
$ make install
```

gcc (version 3.4.6)

```
$ cd /usr/local/src
$ tar xvfz ~/gcc-core-3.4.6.tar.gz
$ mkdir gcc_robin
$ cd gcc_robin
$ ../gcc-3.4.6/configure --target=mipsel-robin-elf --
prefix=/usr/local --with-newlib --enable-languages=c --
disable-threads --disable-shared
$ make
$ make install
```

newlib (version 1.9.0)

```
$ cd /usr/local/src
$ tar xvfz ~/newlib-1.9.0.tar.gz
$ mkdir newlib_robin
$ cd newlib_robin
$ ../newlib-1.9.0/configure --target=mipsel-robin-elf --
prefix=/usr/local
$ make
$ make install
```

4. Revising specs

Open the file ***/usr/local/lib/gcc/mipsel-robin-elf/3.4.6/specs*** with a text editor. Find the labels called *lib and *startfile and replace their values with the following. Save the file with this content.

```
*lib:
-lsemb -lc -lnullmon -lc

*startfile:
crt0%O%s crti%O%s crtbegin%O%s
```

5. Installing semb1200a-rom.ld

```
$ cd ~
$ cp semb1200a-rom.ld /usr/local/mipsel-robin-elf/lib/.
```

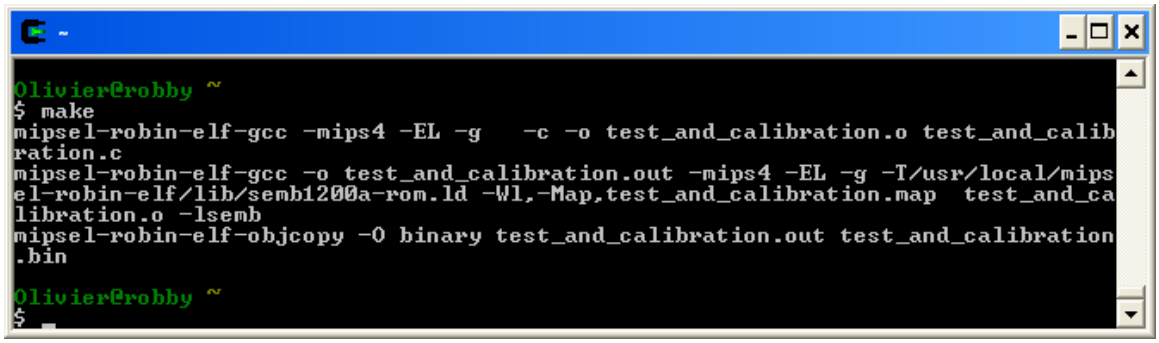
6. Installing lib, header files, and so on

```
$ cd ~
$ tar xfvz lib.tgz
$ cp -r lib/semb1200a /usr/local/mipsel-robin-elf/include/.
$ cp lib/libsemb.a /usr/local/mipsel-robin-elf/lib/.
$ cp lib/crt0.o /usr/local/mipsel-robin-elf/lib/.
```

7. Cross-compiling our first program

Extract the contents of the file **test_and_calibrations.rar**. You should have the files **test_and_calibration.c** and **Makefile**. **test_and_calibration.c** contains the source code of the program in C and **Makefile** contains the instructions on how to compile the code.

In cygwin, go to the folder where **test_and_calibration.c** and **Makefile** are located. Write the command **make**. If the installation is correct, you should see these messages.



The compilation created the following files:

- test_and_calibration.out
- test_and_calibration.o
- test_and_calibration.map
- test_and_calibration.bin

The file **test_and_calibration.bin** is the one that we will be downloading to the robot.

When you will be creating your code, it is often useful to erase all the compiled files. This can be done with the command **make clean** in cygwin. This will remove the four files listed above.

8. Installing Tera Term

The robot is connected to the PC through a USB adapter. In order to communicate with the robot, we need to install a terminal emulator. We will use Terra Term in this class, an open source terminal emulator. Extract the file **ttermp23.zip** and run the file called **setup.exe**. Follow the installer's instructions.

Note: setup.exe will result in an error if the path to the file contains spaces. If this occurs, move the whole folder to a location where the path does not have spaces and run setup.exe again.

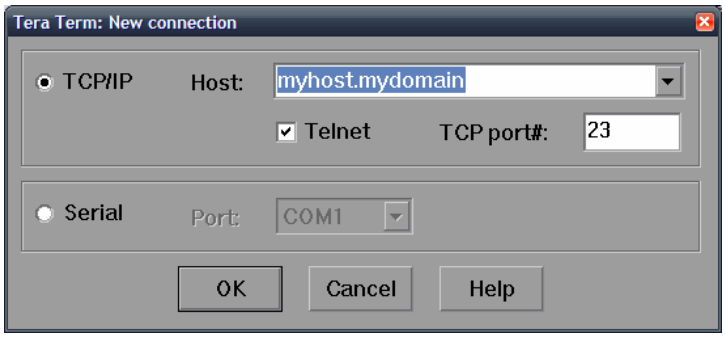
9. Installing the USB Adapter Driver

The driver for the USB Adapter is included in the CD that we gave you in your kit. Connect the robot to one of your computer's USB ports and follow the wizard until prompted about where to search for the driver. At this point insert the CD and resume the installation.

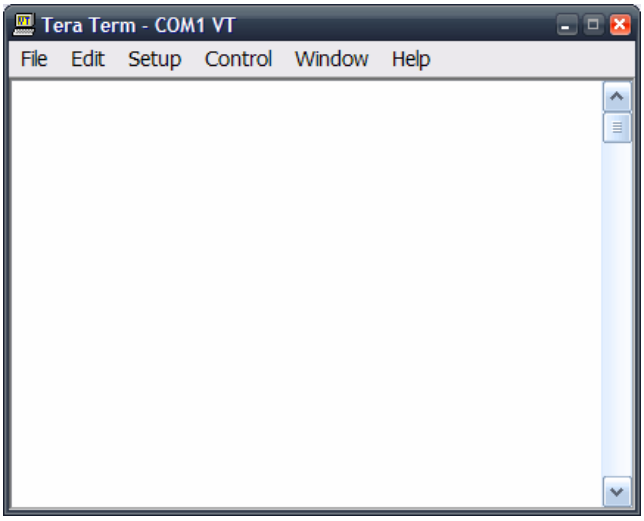
10. Running Tera Term

10.1 Preliminary setup

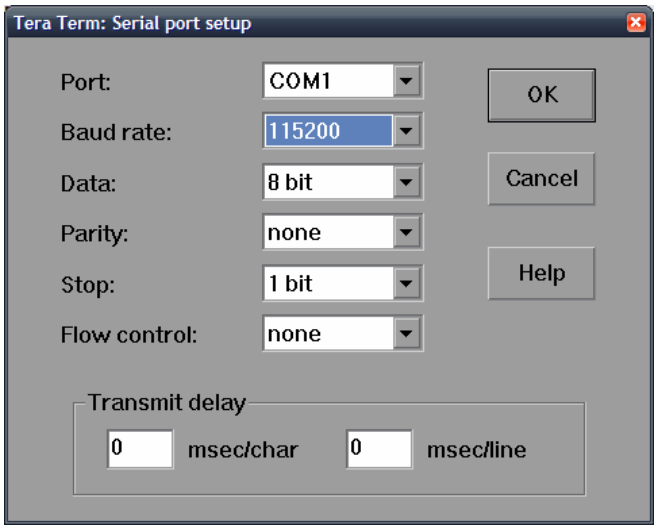
Once you have installed Tera Term, connect the robot to one of your computer's USB ports and run it by clicking the **ttermpro.exe**. You should see this window appear.



Choose the **serial** option and choose the port where the robot is connected (COM 1 to 4). If you do not know which port the robot is connected to, call a TA for assistance. Once you have chosen the correct port, this window should appear.



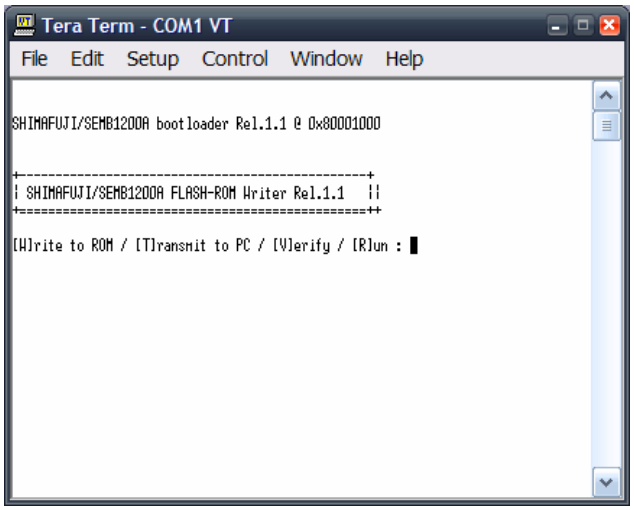
Now, we have to set up the port's baud rate to match the one used by SEMB1200A. To do this, click **Setup** and then **Serial Port…** Change the baud rate of the port you are using to 115200, and click **OK**.



You will have to do this each time you start Tera Term. Once this is done, we are ready to download programs to the robot's board.

10.2 Downloading our first program

Start by turning on the robot's board. If the connection is done correctly, you should see this message appear.



Now press W, and then 1 to tell the program that we want to write a program in the first slot. Now go to the File tab and choose File>Transfer>XMODEM>Send… Once the search dialog appears, find **test_and_calibration.bin**, one of the files the cross-compiler created.

Once you have downloaded the file to the robot, press R, and then 1. This will run the program.

Note: As a result of the very high baud rate, there are usually communication programs between PCs and the robot during the first times programs are downloaded to it. If you see an exception appear right after running the program, it is the result of the corruption of the bin file during the transfer. If this happens, try downloading the file again once or twice.

test_and_calibration.bin is a simple program that allows you to set individual servos to specific angles. Try connecting one of the servos to channel 0 in the distribution boards and setting that servo to various angles.

Use this program to set the servos to the angles specified in the assembly manual.