1 Working with Webots

This guide gives instructions on how to upload code from the Webots environment (fresh installation) to the e-puck robot. Before proceeding, please note that the default installation directory of the Webots program is problematic, and it is important to install the application directly to the C: under windows, or some other controlled directory in linux, at least for the purposes of this guide.

1.1 Installing Webots

Install the webots program to the C: and NOT to the default installation directory. The latest version is always available from their website. If you install webots directly into C: \Webots, the cross-compilation steps detailed below will be drastically easier, since the configuration provided in the e-puck resources folder will already work.

1.1.1 A Note on Dongles

The dongles included with the e-puck robot contain valid Webots licensees. If a dongle is in a USB drive on your computer, you will have complete access to all of the features of Webots .edu edition; if the dongle is not present then you will only be able to use the free version.

1.2 Loading the World

A world has been created for webots, and is located in the e-puck resources folder (The folder where this document is located). It can be found in Webots/my_project/worlds and it is called unmEpuckDevelopment.wbt.

To load the world, simply boot Webots, go to file - open world, and then navigate to the directory that contains the e-puck resources folder.

1.3 The Controllers

There are two controllers loaded into the project you have just loaded, lightFollower.c and movingBlinker.c. For information about what they are capable of doing, please refer to the .c files associated with each controller, as they are well documented. You can plug controllers into epucks in the world by selecting the puck in the Scene Tree, and changing its controller variable to the controller of your choosing.

You can also modify and create controllers usings webots. Please refer to the webots tutorial on their website for more information about how to do this.

1.4 Cross - Compliation

In order to get your code working on the e-puck, your source code MUST be written in C and a couple of steps have to be taken to ensure the process succeeds. Basically, for cross compilation to be possible, your controller must have a file called 'Makefile.e-puck'

stored in the same directory as the controller's source code. In the case of lightFollower and movingBlinker, the Makefile.e-puck already exists. The makefile tells the compiler where the Include list for compiling webots code is located. This file is part of the Webots install, and is always located in the Webots installation directory. If you installed Webots to C: \Webots, then you shouldn't have to do anything else. If, however, it is installed in another location or you're running linux, you will need to edit the Makefile.e-puck file to point to the correct location. It needs to have the following line:
include [webots-directory] \transfer \e-puck \libepuck \Makefile.include
where webots-directory is the path to your webots installation.