E-puck Maintenance and Firmware Programming Documentation

M. Wahby and M. Dorigo

IRIDIA – Technical Report Series
TR/IRIDIA/2020-012
September 2020
In this report, we present a list of possible hardware issues that might arise while working with e-puck mobile robots [1], and we explain how to solve them. Then, we provide step-by-step PIC microcontroller firmware installation guides for both e-pucks and range & bearing boards (e-RandB) [2, 3].

1 E-puck hardware troubleshooting

Here, we list the common issues with the ground sensors, e-puck main board, and e-RandB.

1.1 Ground sensors

a) If you get an I²C error when trying to access the ground sensor module readings please try to clean the connectors and test the six cable connections (using a voltmeter) and resolder the pins if necessary.

![Figure 1: E-puck ground sensor module. The six connections to be tested are indicated by red circles.](image)

b) If the readings at left, middle, and right ground sensors are different, try to push the ground sensor module properly until the left, middle, and right sensors are settled at the bottom of the e-puck housing, as shown in Figure 2.
1.2 E-puck main board

a) If the robot doesn’t switch on or switches randomly between on and off, this means that the battery connectors are not correctly soldered to the basic e-puck board. Resolder the battery connectors correctly as shown in Figure 3. Note that it is very easy to short circuit the board due to the presence of many micro components around the connectors, so you have to be very careful here.

b) If one or more basic board proximity sensors don’t work properly (give improper readings). Make sure the sensors are not tilted (see Figure 4) and are soldered correctly to the board.
c) If the robot is not responding to the Linux board. Either the Linux board is not properly installed on the robot, or the e-puck firmware is incompatible. If the firmware is incompatible, then you need to reprogram the e-puck board (see Section 2).

1.3 e-RandB sensors

a) It is very easy to break the receiver modules on the e-RandB sensors (see Figure 5), therefore, handle the sensors with extreme care and use hot glue to support the receiver modules as shown in Figure 5.

b) In case a receiver module falls off the board, solder it back.

c) If you get an I^2C error when trying to access the e-RandB sensor, if the board provides unreliable communication links, or if it can communicate with other robots at range higher than 1.5m, then, reprogram the firmware of the e-RandB controller (see Section 3).
2 E-puck firmware installation

In this section, we show how to reprogram the PIC microcontroller firmware for e-puck1 using a PIC programmer and via Bluetooth. To reprogram the STM32F4 microcontroller on e-puck2, follow the guide on the wiki page of GCtronic.

2.1 Using X IDE and a PIC programmer

1. Download MPLABX from here.

2. Install the necessary libraries

```
sudo apt-get install libc6:i386 libx11-6:i386 libxext6:i386 libstdc++6:i386 libexpat1:i386
```

3. Extract the MPLAB installer, then run it as an administrator.

```
sudo ./MPLABX-v5.40-linux-installer.sh
```

4. Follow the installer instructions and make sure MPLAB IPE is also installed.

5. Download xc16 compiler from here.

6. Make the compiler executable.

```
chmod u+x xc16-v1.60-full-install-linux64-installer.run
```

7. Run the installer as administrator.

```
sudo ./xc16-v1.60-full-install-linux64-installer.run
```

8. Follow the installer instructions to install the free version.

9. Now start the MPLAB IDE. If it does not complain about the absence of a compiler, then it is installed properly.

10. Plug in the Pickit 3 into one of the PC USB ports. Go to File → Import → Hex/ELF File.

11. At the Create Prebuilt Project wizard do the following:

   (a) At Prebuild Filename, choose the desired firmware to install.

   (b) Set the device to **dsPIC30F6014A**.

   (c) At Hardware to..., Choose the PICKit3 programmer.
12. Connect the robot to the pic programmer as shown:

13. Switch on the robot (Make sure you are using a fully charged battery).

14. Click the “Make and program Device” button.

2.2 Using Bluetooth

1. Download the epuckuploadbt from here.

2. Install the libbluetooth-dev library.

```
sudo apt-get install libbluetooth-dev
```

3. Make epuckuploadbt executable.

```
chmod +x epuckuploadbt
```

4. Make sure an e-RandB module is attached to the robot.

5. Switch on the robot.

6. Pair the robot with the computer (use the board ID as pairing key).

7. Use the following command to upload a firmware.

```
./epuckuploadbt firmware.hex (board ID)
```

   e.g., "./epuckuploadbt pi-puck-e-puck1.hex 1423"

8. Press the reset button on the e-RandB sensor (see Figure 5), when prompted.
3 e-RandB firmware installation

Follow the same procedure for e-puck firmware installation using MPLAB and PICKit3 programmer but set the device to Microchip dsPIC33FJ256GP710 and connect the PICKit programmer to the connector on the eRandB board (see Figure 5).

References

