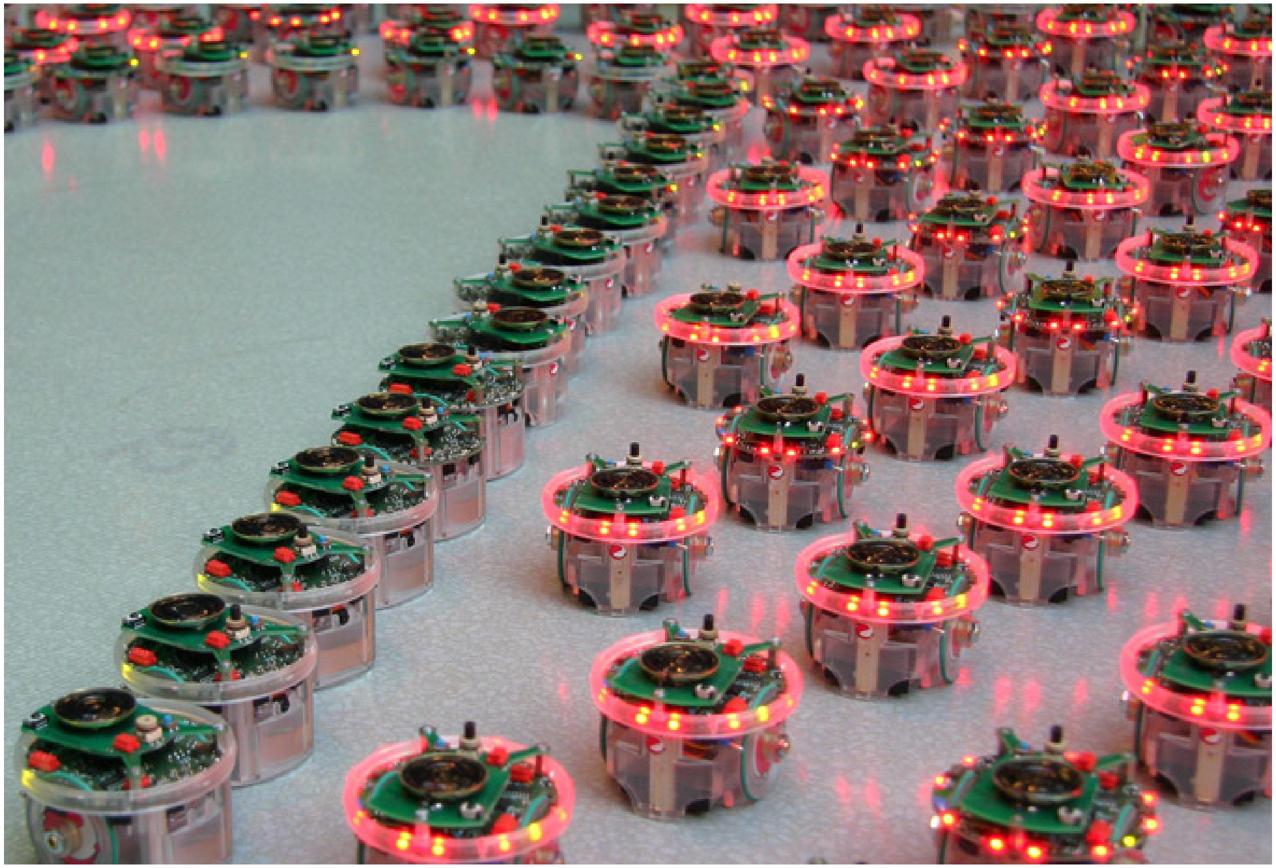
Introduction to the e-puck robot

www.e-puck.org



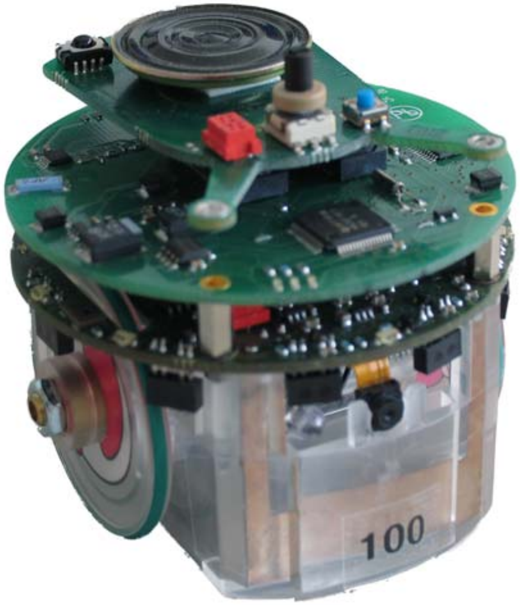
Please boot your PC under Linux

The documentation

- With the files you got from SVN, you have two PDF in the *documentation* directory
  - Intro-robot-linux.pdf
    General information about the e-puck robot.
    - How the proximity sensors work ?
    - How to design software for the robot ?
    - What to keep in mind when working close to the hardware ?
  - Linux-epuck-gettingstarted.pdf
    - How to build a project ?
    - How to upload a new software on the robot ?
- Please read them (you should already have read them), they might contain useful information for answering questions.

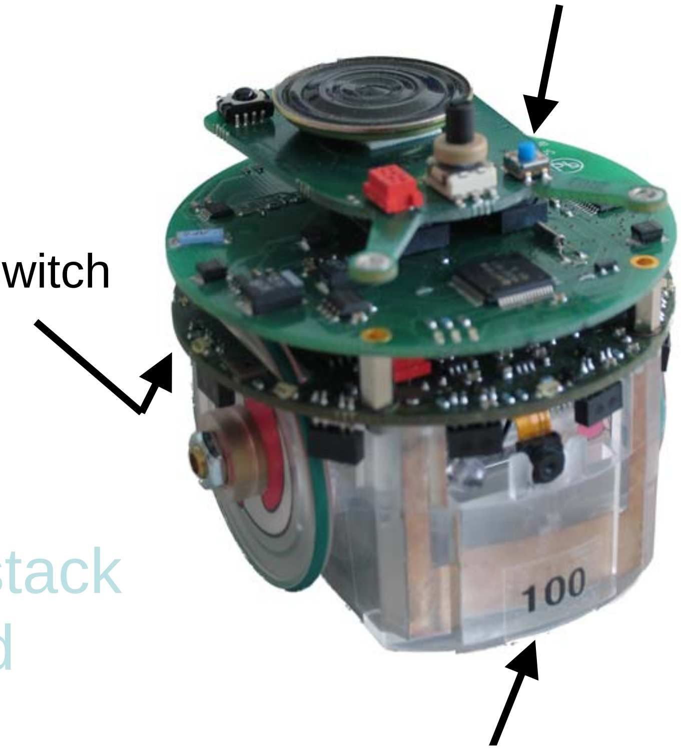
The robot

- DsPIC30F6014A platform up to 30Mips
- 2 step-motors
- 8 IR sensors
- 3 microphones
- Color camera
- 3 axis accelerometer
- Bluetooth serial transmission
- A light ring around the robot
- Bus connectors to allow board stack
- Area to add a floor sensor board
- Robot size is Ø 7cm x 5cm
- Power on indicator, low battery indicator, removable battery, program selector, lightened transparent body, RS232 connector with external power supply ability, de-bounced reset button, design to allow plugged board to access onboard hardware.



The robot

- DsPIC30F6014A platform up to 30Mips
- 2 step-motors
- 8 IR sensors
- 3 microphones
- Color camera
- 3 axis accelerometer
- Bluetooth serial transmission
- A light ring around the robot
- Bus connectors to allow board stack
- Area to add a floor sensor board
- Robot size is Ø 7cm x 5cm
- Power on indicator, low battery indicator, removable battery, program sele transparent body, RS232 connector with external power supply ability, de-bounced reset button, design to allow plugged board to access onboard hardware.



“How to use it ?” part 1 : remote control

- The robot runs a command interpreter : the *sercom.*
- You connect your PC to the robot using a Bluetooth link.
- You send commands to the robots and it executes them.
- Step by step :
  - Open a terminal window
  - Assuming you have e-puck 24, type:
    minicom epuck24
  - Wait until the serial link is established (little yellow LED)
  - Press the reset button of the e-puck (little blue button)
  - You should get : “WELCOME to the SerCom…”
  - Type “H” to get help on the available commands.
- That’s it !

“How to use it ?” part 2 : embedded software

- The robot runs your program from its internal Flash memory.
- Use Bluetooth to upload the new software.
- It uses a bootloader to program itself.
- Bootloader ?
  - A software with two parts:
    - On the robot : The software is executed after reboot or reset, waits for data from the BT link, programs the flash with the new data.
    - On the PC : Establish the BT link with the robot, and send the new program.
  - Remember : first execute the PC part, then press the reset button on the robot to execute the bootloader.

The lab itself

- 3 parts :
  - Warm up part : You will play with the robot using remote control. The purpose of this part is that you get a feeling about the robot. Don’t spend too much on this part.
  - Remote control using Webots : You will use Webots to run a Braitenberg algorithm on the robot.
  - Wall follow implementation on the robot itself : you will write an embedded software for the robot and upload it.

What to keep in mind ?

- You have two PDF files available with a lot of information for questions like :
  - How to use the bootloader ?
  - How to build a project ?
  - Where is the Bluetooth LED located ?
  - What is the numbering of the LED ring ?
- We are in the real world ! There is noise, differences between sensors, hardware limitations.
- The CPU is slow and has no hardware floating point support, act accordingly.