

Lab 3

Distributed Intelligent Systems

Chiara Ercolani

9.10.2018

What this lab is about

- Familiarize yourself with the e-puck robot
 - how to program the robot using the e-puck library
 - how to transfer data to and from the robot using serial comm.
- Basic understanding of:
 - e-puck sensors
 - stepper motor
- Implementation: Basic robot control
 - Braitenberg
 - Rule-based
- Robot Navigation:
 - Pure odometry
 - Feature aided odometry

Software

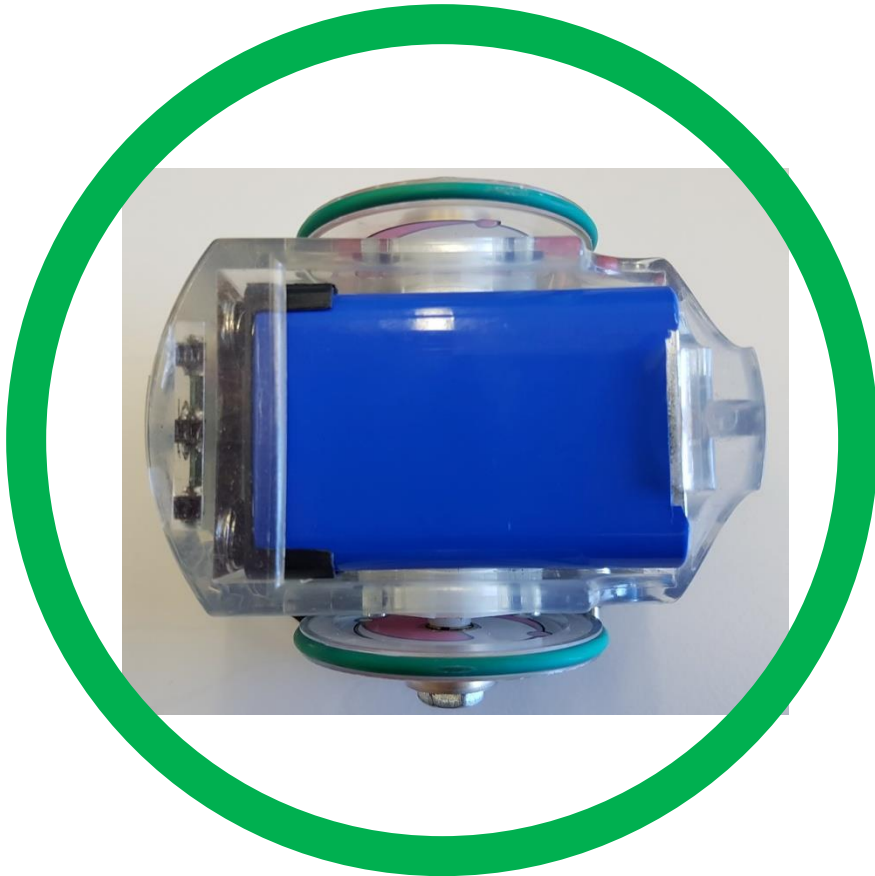
- On a GIT repository
 - download files by executing
`git clone https://disalgitn.epfl.ch/epuck/epuck.git`
- Content of folder `EpuckDevelopmentTree/`
 - e-puck library
 - test programs (not needed in this lab)
- Executables: already installed in GR B0 01 and GR C0 02
 - epuckconnect
 - epuckupload

Hardware

- You will receive:
 - 1 e-puck robot
 - 1 Battery
 - 1 USB-Bluetooth dongle
 - 3 walls
 - 3 stands



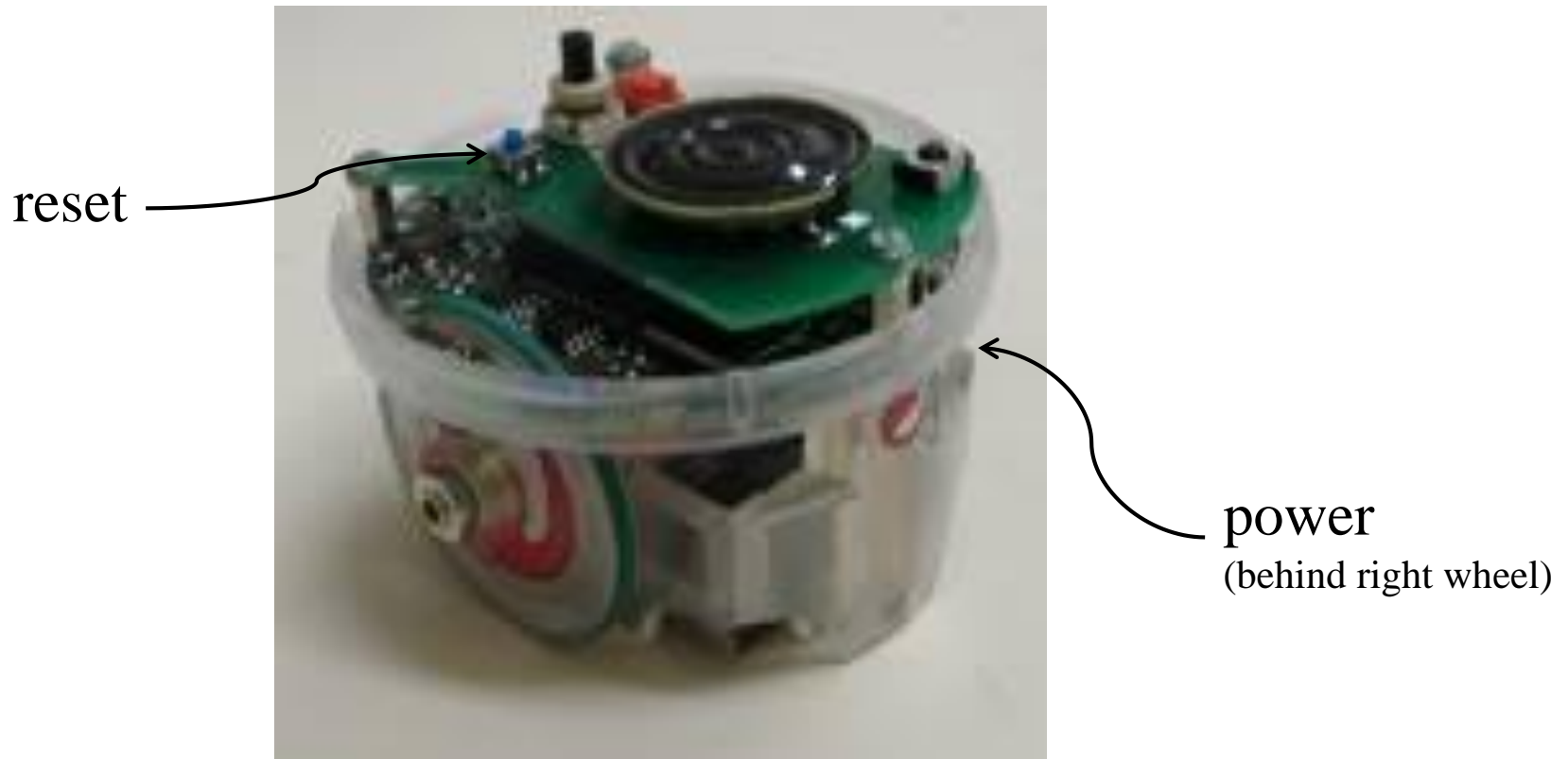
e-puck: insert battery



If you can see the serial number it's wrong

e-puck: Reminder

You will need to use 2 buttons on the robot:



Also: watch for the power (green) and the low-power LED (red)

Programming an e-puck

- Turn on robot, plug USB dongle into computer
- Upload program abc.hex
`epuckupload -f abc.hex 123`

Important Notes:

- In the computer room, type the following **before** the epuckupload command:
`ssh localhost`
You only need to do this once every time you open the terminal you use for the epuckupload, epuckconnect and minicom commands.
- Before compiling a C file (abc.c), edit your Makefile:

```
EPUCKLIBROOT=/home/user/mydocs/epuck/EpuckDevelopmentTree/library
```

Bluetooth issues

If you are having problems connecting and/or uploading code, do the following:

- Unplug the USB-Bluetooth dongle
- Plug in the USB-Bluetooth dongle
- Left click on the Bluetooth icon on the top right screen
 - Disable
 - Enable

You cannot upload code while the minicom is used!