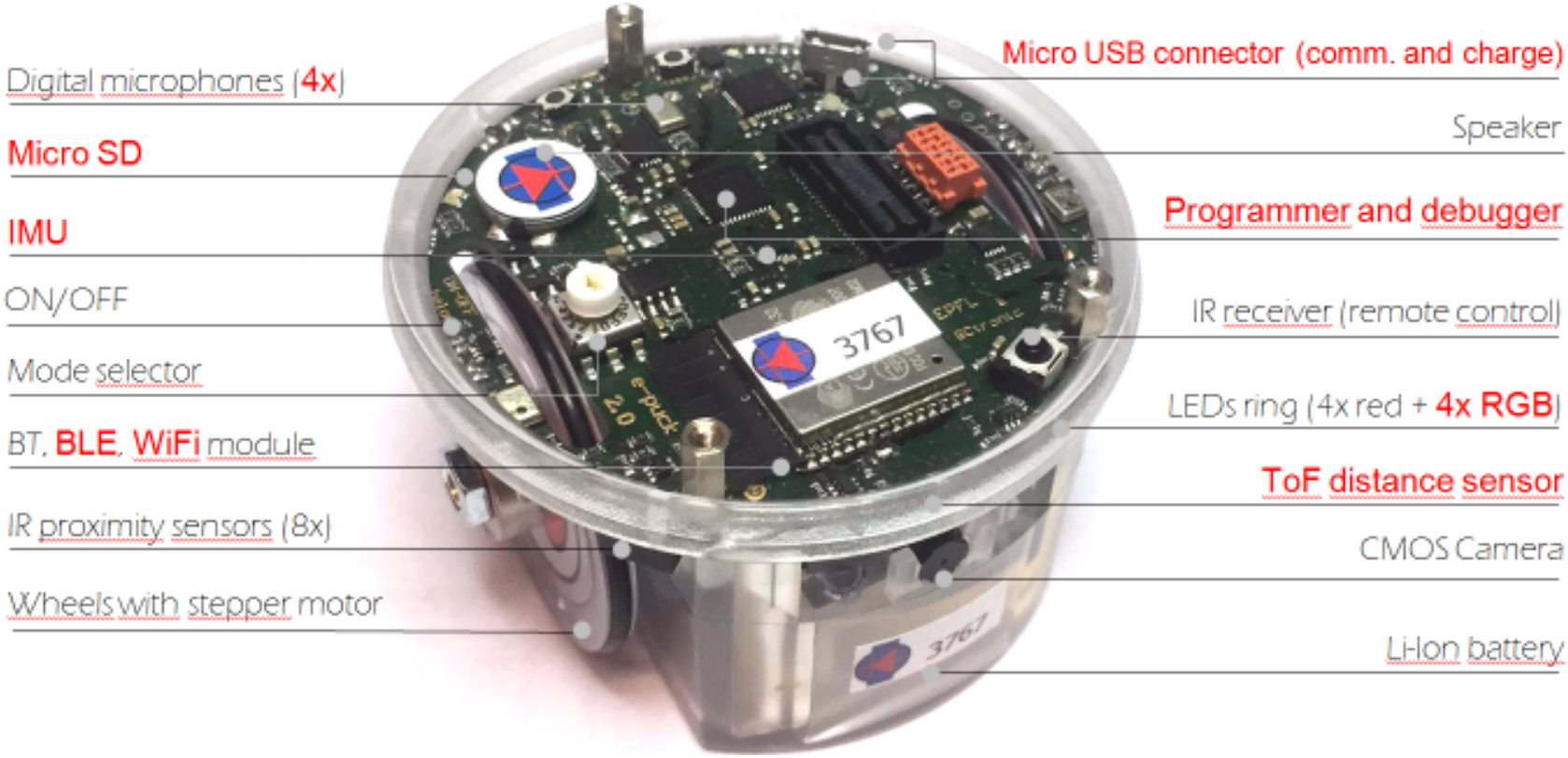
e-puck Lab overview

Roderich Gross

The University of Sheffield

October 11, 2023

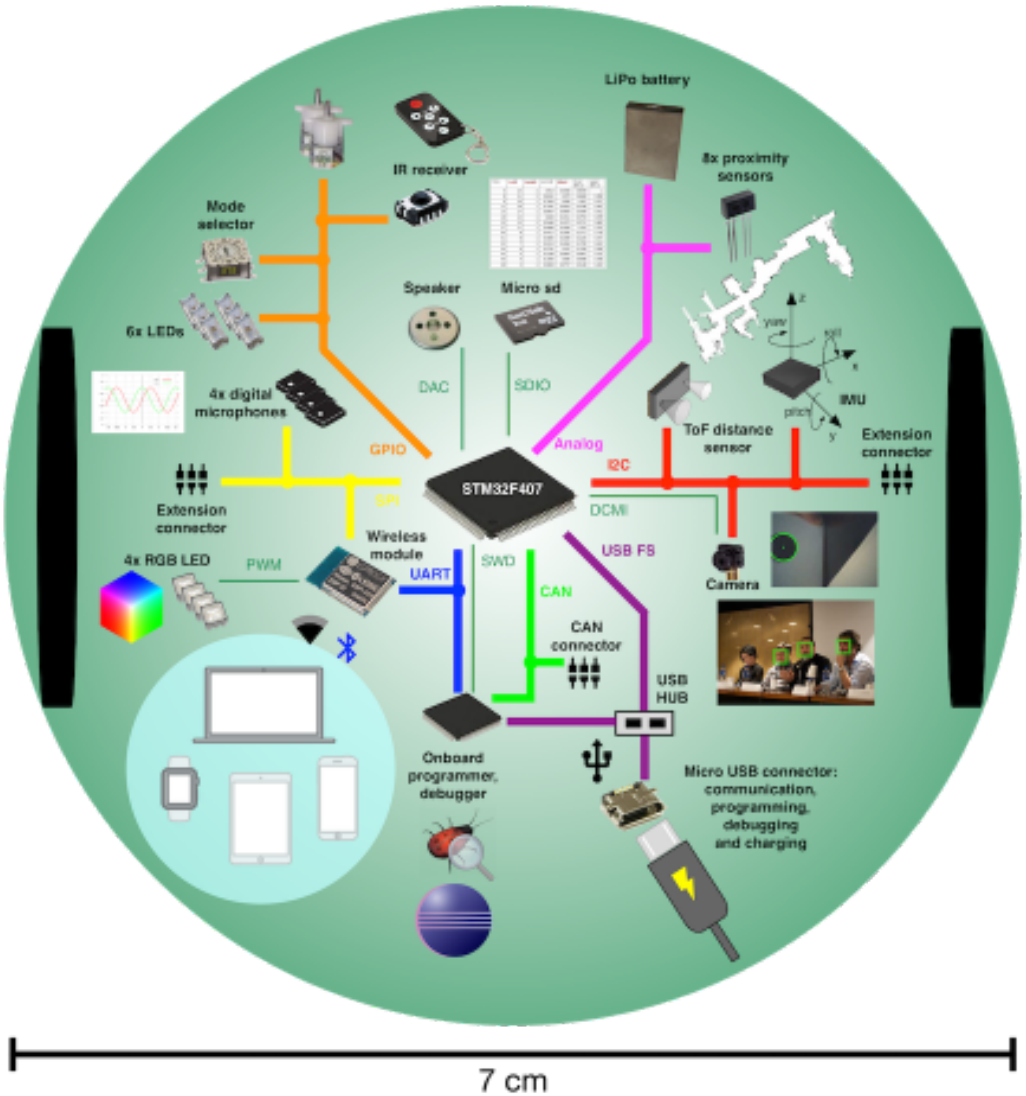
The e-puck2



- Developed by EPFL and manufactured by GCtronic.

Reprinted from `https://www.gctronic.com/doc/index.php/e-puck2`

The e-puck2



Reprinted from `https://www.gctronic.com/doc/index.php/e-puck2`

Objectives

- To gain hands-on experience with a *real*, autonomous robot
- To design and analyse a control system for an embedded system
- To become familiar with software development process
  - The Unix Command Line
  - Integrated development environment (IDE): Eclipse
  - Programming languages: C and/or C++
  - Distributed version-control system (`git`)
- To gain experience of working in a team
- To gain experience of writing an academic report (and using LaTeX to do so)

Assessment (30% of module mark)

- "Teams" of 2–3 students
- Before Week 4, each student
  - completes the Blackboard pre-lab quiz
- In Week 6, each team
  - performs two lab **demonstrations** (50% of assessment)
- By Week 8, Mon, noon, each team
  - submits their **team report** via turn-it-in (50% of assessment)
  - submits their **source code** via turn-it-in

WANTED: Committed team members :-)

- Assessment corresponds to 4.5 credits = 45 hours of work. E.g. each team member to devote
  - 10 h for labs (including demos)
  - 10 h preparing for the pre-lab quiz and Week 4 lab
  - 5 h preparing for Weeks 5 labs
  - 5 h preparing for Weeks 6 labs
  - 15 h preparing for report and source code submissions
- Where a team member does not engage, follow advice in Section 6 of *Assessment Briefing*.
- Remark: In the event of significant concerns regarding the engagement of a particular team member, penalties could be applied (see Section 6 of *Assessment Briefing*).

Lab schedule

Group	Session	Week	Day	Time
1	1	4	Mon	09:00–10:50
1	2	5	Wed	11:00–13:50
1	3	5	Thu	13:00–14:50
1	4	6	Mon	10:00–11:50
1	5	6	Wed	11:00–12:50
2	1	4	Wed	12:00–13:50
2	2	5	Tue	09:00–10:50
2	3	5	Thu	11:00–12:50
2	4	6	Tue	10:00–11:50
2	5	6	Wed	13:00–14:50

Pre-lab quiz

- The pre-lab quiz is released Week 3, Tuesday (9am).
- Each team member must complete it on their own **prior to Week 4** (technically, Week 3, Sunday, 23:59).
- Not attempting the pre-lab quiz (or obtaining a mark of 0) will result in a 25% penalty on a student's demo mark (other team members are not affected) even where the student is otherwise engaging well with the assignment.

Read supporting materials! (on Blackboard)

- *Team Allocation* (check whom you work with)
- *Assessment Briefing* (read prior to lab)
  - Detailed marking criteria for both demonstrations and report
- *Lab Induction* (read at least Sections 1–4 prior to lab)
  - What to do at the beginning of each lab. By using `git` you will reduce the setup time from around 15–30 mins to 2–4 mins from Sessions 2 onwards (while also having systematic backups and version control)
  - Unassessed tasks to learn core features of the robot (Session 1)
- *Library Cheat Sheet* (read prior to lab)
  - Explains source code for using various features of the robot
- *General Guidelines on Writing Reports* (read in Week 5)

Inside the lab

- Programming the robot
  - Each team is allocated a Linux PC (Ubuntu)
  - It is your responsibility to backup your solutions at the end of each session using `git`
- High-value research equipment: £950 per robot
- **Needs to be handled with care**
  - Your smart phone is 100 times more robust.
  - No food or drink in the lab.
  - Be gentle!
    - Do not *force* USB connector into slot.
    - Do not push buttons too hard.
    - Operate the robot only inside its arena (expect for Task 2).

Tasks

- Unassessed tasks (optional; see Section 4 of *e-puck2 Lab Induction* document)
- Task 1
- Task 2

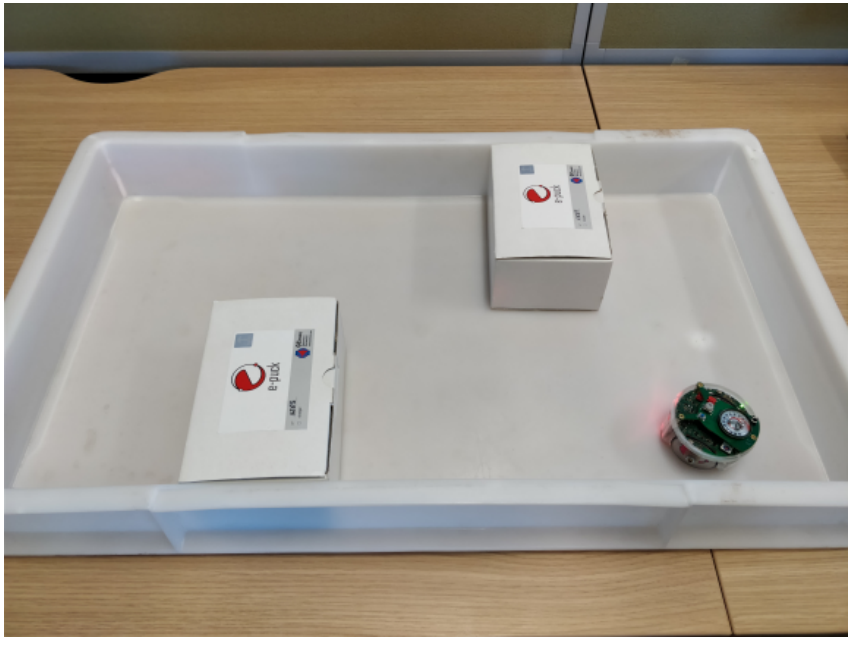
Unassessed tasks

- The unassessed tasks are
  1. Toggle the green body LED
  2. Control the movement with the selector switch
  3. Detect nearby objects and analyse the sensor data
- They help you develop a program, and become familiar with:
  - LEDs
  - Selector switch
  - Motors (wheels)
  - Proximity sensors
  - UART module (Bluetooth)

Example program

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <string.h>
4  #include <math.h>
5
6  #include "ch.h"
7  #include "hal.h"
8  #include "memory_protection.h"
9  #include <main.h>
10
11
12 int main(void)
13 {
14
15     halInit();
16     chSysInit();
17     mpu_init();
18
19
20     /* Infinite loop. */
21     while (1) {
22         //waits 1 second
23         chThdSleepMilliseconds(1000);
24     }
25 }
26
27 #define STACK_CHK_GUARD 0xe2dee396
28 uintptr_t __stack_chk_guard = STACK_CHK_GUARD;
29
30 void __stack_chk_fail(void)
31 {
32     chSysHalt("Stack smashing detected");
33 }
```

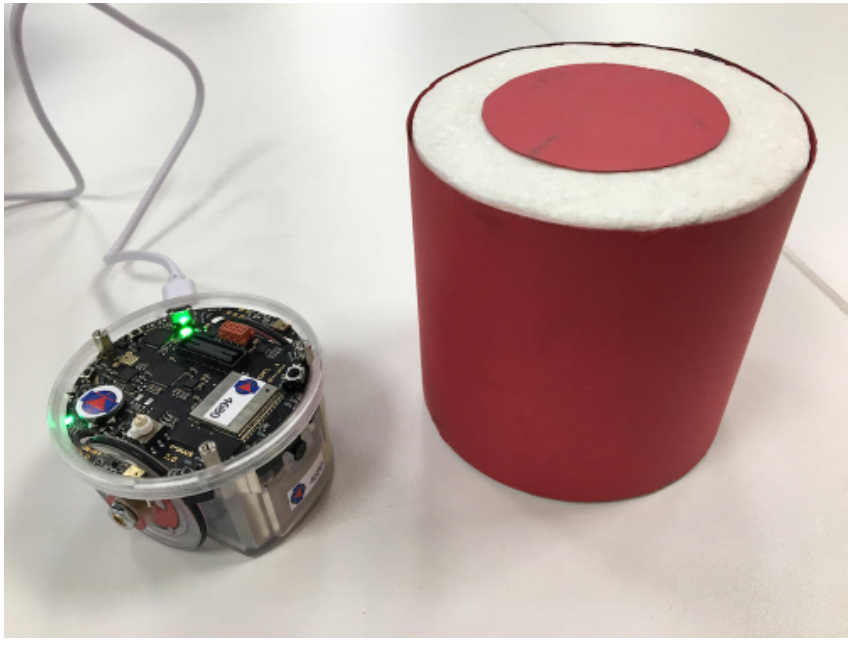
Task 1: Explore the environment



You shall design, implement and test a control strategy for the robot

- to explore a bounded environment with obstacles;
- to avoid any collision with the environment boundary or obstacles.

Task 2: Chase an object



You shall design, implement and test a control strategy for the robot

- to chase an object in an open environment that is free of obstacles;
- to avoid any collision with the object.

Tasks 1 and 2

- Your control strategy must be executed directly on-board the e-puck2 robot.
- **Up to 40 points** can be gained if the robot performs Tasks 1 extremely well.
- **Up to 40 points** can be gained if the robot performs Tasks 2 extremely well.
- **Up to 20 points** are awarded if the robot demonstrates abilities that go **significantly** beyond what is expected. These abilities need to relate directly to the task the robot is performing. Be creative!

Tasks 1 and 2

Regarding the creative part:

- No points will be awarded for trivial extensions;
- 1 to 10 points: Significant advancements of abilities, clearly distinct from what is expected;
- 11 to 20 points: Outstanding demonstration that is highly original and/or could have lasting impact (e.g. open days demonstrations).

Preparing your team report using LaTeX

- Week 3: Teams get invited to their project on `https://www.overleaf.com/`. This project already contains the correct IEEE template.
- Week 4–7: Teams are encouraged to start writing their reports as early as possible (given them more time to experience LaTeX). LaTeX specific questions can be posted on the *Discussions Board*.
- Week 5, Friday: Teams to receive feedback helping them to write better reports.
- Week 7, Friday: Teams to receive feedback helping them to write better reports.
- Week 8, Monday (noon): Teams to submit report and source code via turn-it-in.

The e-puck lab

Questions?