setting up the physical maggy

Contents map

developed content units	taxonomy levels
maglev systems	u1, e1

prerequisite content units	taxonomy levels
arduino IDE	u1, e1
PID control	u1, e1

Main ILOs of lab "setting up the physical maggy"

Run a pre-given Arduino image on the physical maggy

Test how changing the PID controller parameters
changes the closed-loop behavior of the physical maggy

Overview of the structure of this lab

- what is what, in maggy
- how to connect maggy to a PC
- how to launch the Arduino IDE
- how the Arduino image we give you is structured
- how to change the parameters of the PID controller on the Arduino image
- how to flash an Arduino image on maggy
- how to get sensors readings from maggy on the PC, and visualize them

Flashing Maggy's Arduino Code In The Hardware

Steps

- remove the levitating magnet before uploading any new code. The controller needs to self-calibrate sensors, and failure to remove the magnet will disrupt calibration
- open a Linux terminal (win button, then type cmd, then press Enter)
- run the setup script `run_arduino_ide_setup.sh` and confirm the warning with Y (uppercase)
- wait up to ~3 minutes for the script to complete. Messages about "removing existing directories" and "untar archives" will appear. Let it work, and open another terminal
- in this new terminal, launch the Arduino IDE with: `arduino-ide`. If prompted for updates, press "Skip"

Steps

- open yet anoter terminal, and clone the repository with
  - `mkdir Maggy`
  - `cd Maggy`
  - `git clone https://github.com/Hansolini/Take-home-Maglev-lab.git`
- open the .ino file in the Arduino IDE `Maggy/physical_system/software/V4x_PD_example/V4x_PD_example.ino`
- install the TCA9548 and Tlv493.d libraries in the Arduino IDE using the Library Manager
- connect the USB-C port to the power transformer. Wait for the 12V LED to turn on. If it doesnt, notify the instructor
- connect the USB-2 port to the lab PC
- in the Arduino IDE, select the board "teensy4.1" from the top ribbon and upload the .ino file (Sketch → Upload)

Steps

- wait for the upload to complete (check the output terminal)
- open the Serial Monitor (magnifying glass icon in the IDE). You should see a stream of numbers
- open the Serial Plotter (heartbeat icon)
- place the levitating magnet and observe stabilization. If successful, Maggy will levitate
- to turn off Maggy:
  - remove the levitating magnet
  - disconnect the USB-2 connector
  - disconnect the USB-C connector
  - place the magnet back on the board for storage

Understanding Maggy's Arduino Code

Overview

- goal = control the current through the solenoids by processing sensors readings
- implemented algorithm = all the necessary steps to implement a real-time feedback control for stabilization
- macro modules = sensor calibration, real-time monitoring, and control action

System Initialization

- Initializes serial communication for debugging
- Sets up I2C communication for sensors.
- Calibrates sensors and solenoids before operation.

Sensor Data Acquisition

- reads magnetic field values from the sensor
- detrends sensor values by subtracting mean values
- applies a low-pass filter to smooth readings

Control Loop Execution

- runs at a fixed frequency
- has the goal of maintaining stability
- does so through computing error signals used then by a PD controller
- applies control action to solenoids via PWM

Sensor Freeze Detection

- detects sensor malfunction by comparing consecutive readings
- if a freeze is detected, resets the sensor and I2C bus

Logging System State

- logs key parameters for debugging and performance analysis
- helps in tuning control parameters

Assigmnents

Question 1

What is the perceived effect of increasing the gain of the proportional part in the PID controller with respect to the originally given parameters, on the closed loop performance of the system? And what is the perceived effect of decreasing such a gain? Moreover, how did you arrive at such conclusions?

Question 2

What is the perceived effect of increasing the gain of the derivative part in the PID controller with respect to the originally given parameters, on the closed loop performance of the system? And what is the perceived effect of decreasing such a gain? Moreover, how did you arrive at such conclusions?

Question 3

Try to add an integral action to the controller. What is the perceived effect of introducing the integral part in the original controller on the closed loop performance of the system? And what is the perceived effect of increasing or decreasing its gain? Moreover, how did you arrive at such conclusions?

summarizing

Most important software for this lab

- Arduino IDE: `https://www.arduino.cc/en/software`

Suggestions

- you will have to perform some tests; best to know at least the basics on how to design experiments:
  `https://en.wikipedia.org/wiki/Factorial_experiment`, or
  `https://youtu.be/cHoYAyYHzB0`

Recap of lab "setting up the physical maggy"

- to be able to do development and tests on the hardware there is the need for knowing how to set it up
- trial and testing may be cumbersome, it may be good to make a plan of which experiments one may want to do before starting doing them