michael.engel [☺] hs-coburg.de

Autonomous Embedded Systems
Summer 2018

Assignment 5

Assignment issued on Friday, May 11, 2018

Honor code: Work on the assignments is to be carried out in the assigned groups of two or three students. You're welcome to talk to classmates about assignments conceptually, but the implementation must be your own unless otherwise noted. For example, speaking with a classmate about why some bug might be present in your program is okay as long as your classmate doesn't share with you a program that fixes your bug. You should not use any existing code that we don't supply, whether it is online or otherwise. Further, copying code from places like Stack Exchange is prohibited: you will learn much more if you write your own programs.

5.1 Motors

Our robot is now finally prepared to drive. The motors are controlled using the following GPIO pins:

- Left motor: P1.25 – forwards, P1.26 – backwards
- Right motor: P1.23 – forwards, P1.24 – backwards

In addition, the motors can be controlled using PWM:

- Left motor: P1.21 (PWM1.3)
- Right motor: P1.20 (PWM1.2)

Implement the following functionality:

- Write a function that lets the robot drive. The following modes are to be supported:
  - Drive forward
  - Drive backward
  - Turn centered clockwise (around the middle point of the robot)
  - Turn centered counterclockwise
  - Turn off-centered clockwise (around one of the wheels)
  - Turn off-centered counterclockwise
  - Stop the robot
- Write functions to drive the robot for a given amount of time or distance (in a straight line) and turn a given angle. Calibrate distance and angle to the duration of motor activation.

5.2 Encoder-Controlled Driving

A more precise acquisition of distance and speed is possible by using the encoder disks mounted to the robot's wheels. These can be read out using phototransistors at GPIO pins P2.12 = EINT2 (left) and P2.11 = EINT1 (right). The phototransistors are able to generate interrupts and you are expected to use these.

- Write interrupt functions that count the number of markings seen on the left resp. right encoder disk. The C signature of an interrupt handler is `void EINT1_IRQHandler (void);` and it can be activated using `NVIC_EnableIRQ(EINT1_IRQn);`.

  The external interrupts have to be triggered on the rising edge of the signal, see the LPC1778 documentation on the registers `LPC_SC->EXTMODE` and `LPC_SC->EXTPOLAR`.

  Remember to configure the mode of the interrupt pins to the alternative function "external interrupt" and activate the pull-up resistor and hysteresis.

- Implement functions to drive and turn the robot controlled by the encoder disk counters (instead of controlling the motor activation time). Calibrate the functions accordingly.