HOCHSCHULE
RHEIN-WAAL
Rhine-Waal University
of Applied Sciences

LABORATORY

Microelectronic Control Systems

EXPERIMENT:

Ball and seesaw

Please read the whole document before starting the experiment. This document contains 5 preparation jobs. It is mandatory to make all preparation jobs in written form! Without a written document that includes all preparation jobs it is not possible to pass this experiment.

1 The experiment set-up

This experiment consists of one ball, that is able to move in only one dimension. This ball is placed in a rail, where an angle can be set (positive and negative), see Figure 1.

So we have:

- Actor: One servo motor to control the angle of the rail
- Sensor: eight light barriers that show if the ball is at fixed positions

The eight light barriers are placed at equal distances from each other. They are able to detect when the ball is at one end of the rail. There are positions between the light barriers, where the ball can be "'hidden"'.

A schematic drawing of the mechanical part is in Figure 2.

1.1 Electronic

The electronic circuit is shown in Figure 3. Please have a look at the schematic diagram. The electronic of the servo is not in this circuit, because this is a pre-build part, explained in 1.2.

Prepare 1: The servo has its own voltage regulator and there is a low pass filter. What could happen if this filter is missing or if the circuit has only one voltage regulator?

Prepare 2: Should the internal pullup resistors inside the microcontroller for the eight light barriers on or off?

1.2 The servo motor

The servo motor is an integrated product. Normally this device is used to steer a remote controlled car. It is produced in bulk, so this is possible to buy it on good terms.

The receiver from the remote control gives the servo three wires:

- 5 volt power supply
- GND
- The signal

The servo will automatically go to the wanted position as fast as possible. It needs 0.2 seconds for 60 degree. If it is necessary to drive slower, the desired position must change with an equal slow speed. There is no feedback to the microcontroller, what the real position is. This is not a problem, since it is impossible for the servo to be hanging.

You can see an example of an equal servo in Figure 4.

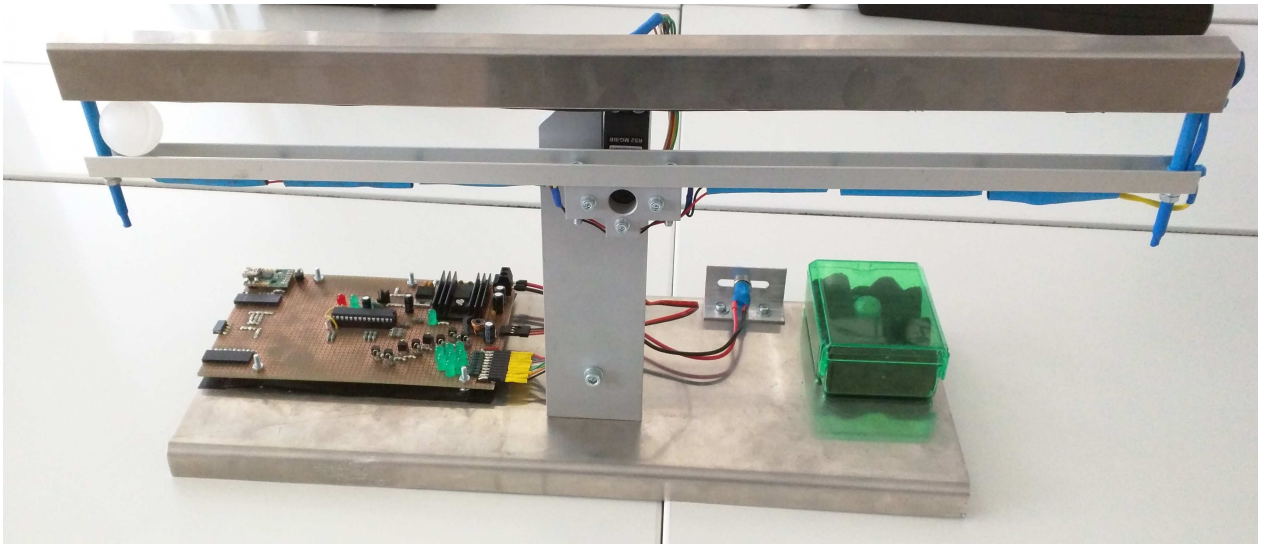

Figure 1: The experiment setup

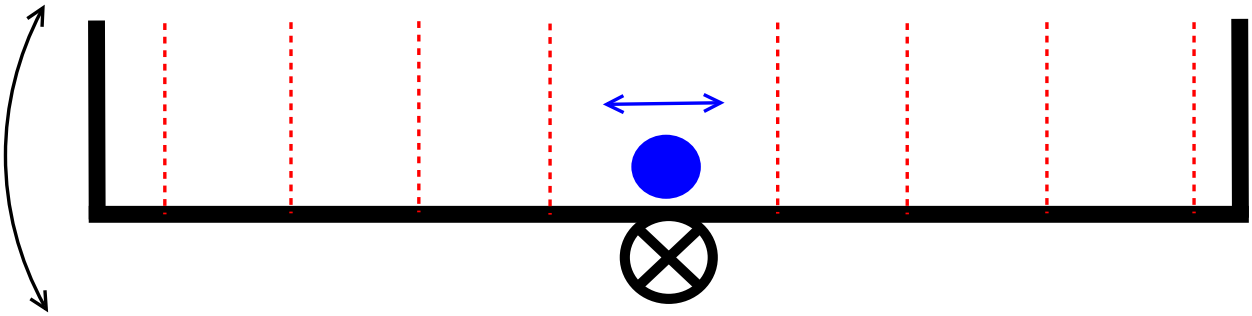

Figure 2: The mechanical part

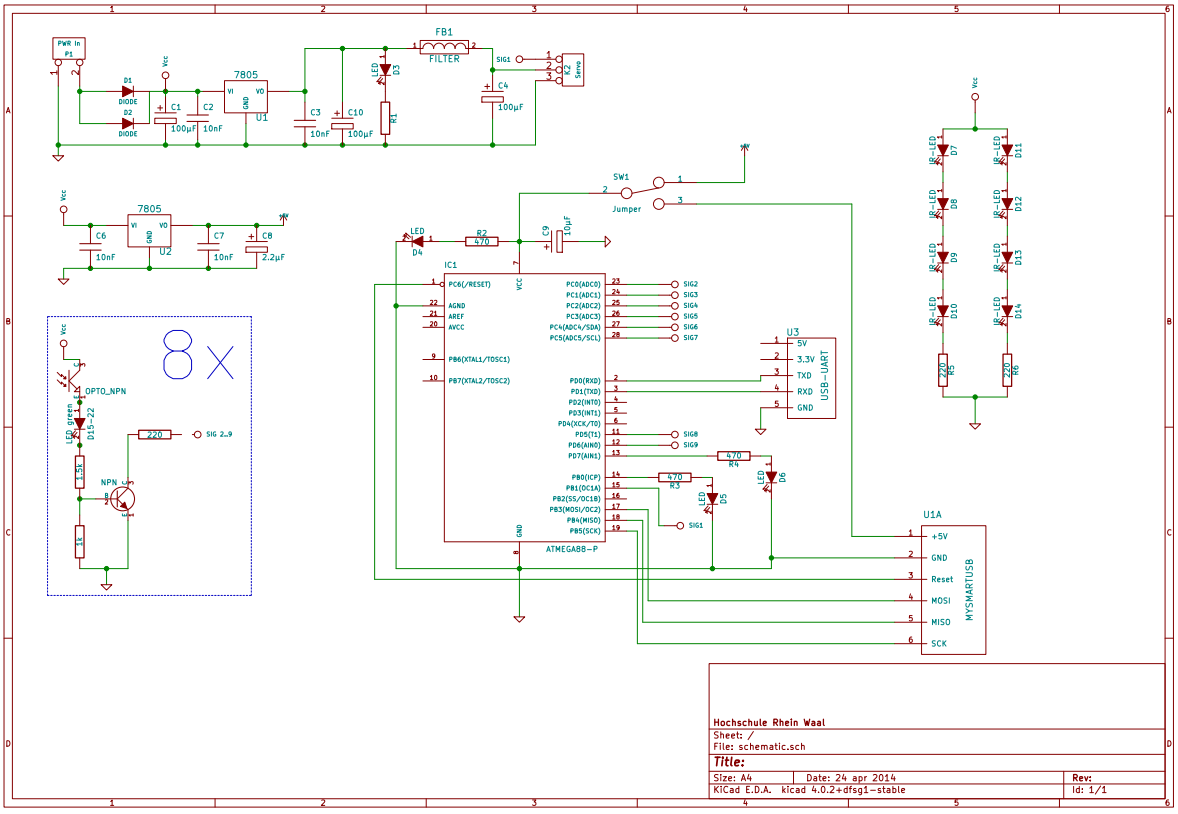

Figure 3: The electronic circuit

Figure 4: Servo

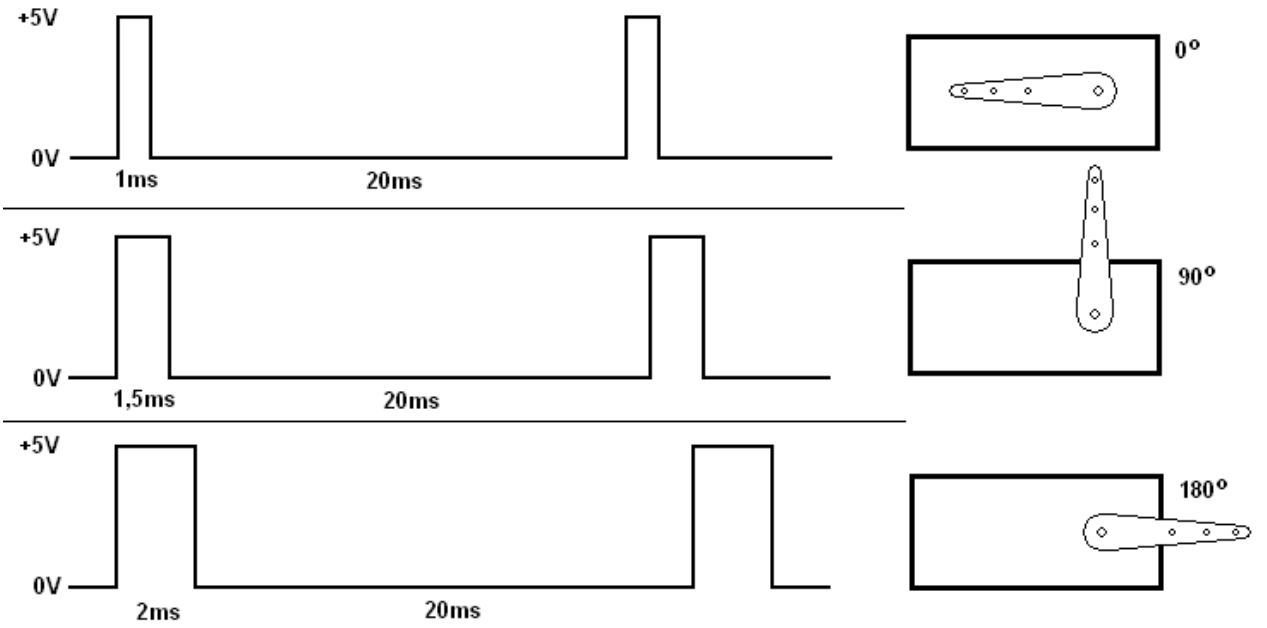

Figure 5: Servo-Signal

The signal is periodic in 20ms. The position is coded with an impulse of 1ms to 2ms. 1ms is the code for the left side, 2ms for the right side. The middle position is codes with 1.5ms, the relation between the impulse time and the position is linear, see Figure 5. The 20ms delay between two pulses are uncritical, minor changes have no effect.

Prepare 3: The servo signal input is connected to the port PB1. Why this port? Look into the datasheet of the Atmel Mega88 and read about OC1A.

2 The challenge

The challenge is to write a program, that places the ball in the middle of the seesaw. When something pushes the ball to one side, the controller should correct this as fast as possible by moving the ball back to the middle.

Your program is good when your program can completely stop or extremly slow down the ball inside the middle area (middle position +/- two sensors) without using the mechanical limit left/right.

There are a few problems:

- The exact position and speed of the ball are unknown and must be calculated
- Too quick position changes or jerky movements throw the ball out of the rail
- A function to generate the signal for the servo position must be created
- The middle position is not exact, there is a small offset, you need to find out the right value for this
- A PD controller is necessary

Prepare 4: Why is a simple P controller not possible?

Prepare 5: What advantage has a PID controller instead of an PD controller?

If all at this point is complete and works fine, inform the staff. You must be able to show and explain all steps you have done. The next steps are optional.

If your controller program works fine, try the other ball. The difference between the two balls is the weight. The other ball is massive. First edit your program, that it works fine with the heavy ball. Then write one program that works regardless of which ball is used. An automatic test routine at the program start to find out what ball is inside the rail is allowed.

3 Appendix

Here are some additional information's.

3.1 Ports of IR sensors

The ports for the IR light barriers from left to right:

1. PD6
2. PD5
3. PC0
4. PC1
5. PC2
6. PC3
7. PC4
8. PC5

3.2 Microcontroller

The used microcontroller in this experiment is from Atmel:

- Type: Atmel Mega88
- Clock rate: 8MHz

To get more informations read: http://www.atmel.com/devices/atmega88.aspx