CPE416
Fall 2023 - Seng
Lab 4

Due Date

Friday, 11/10/23

Objectives

- To implement Monte Carlo localization

Part 1

Cut out the wheel encoders and tape them on the inside of your robot wheels. Verify that the sensor is working and can read the encoder disk.

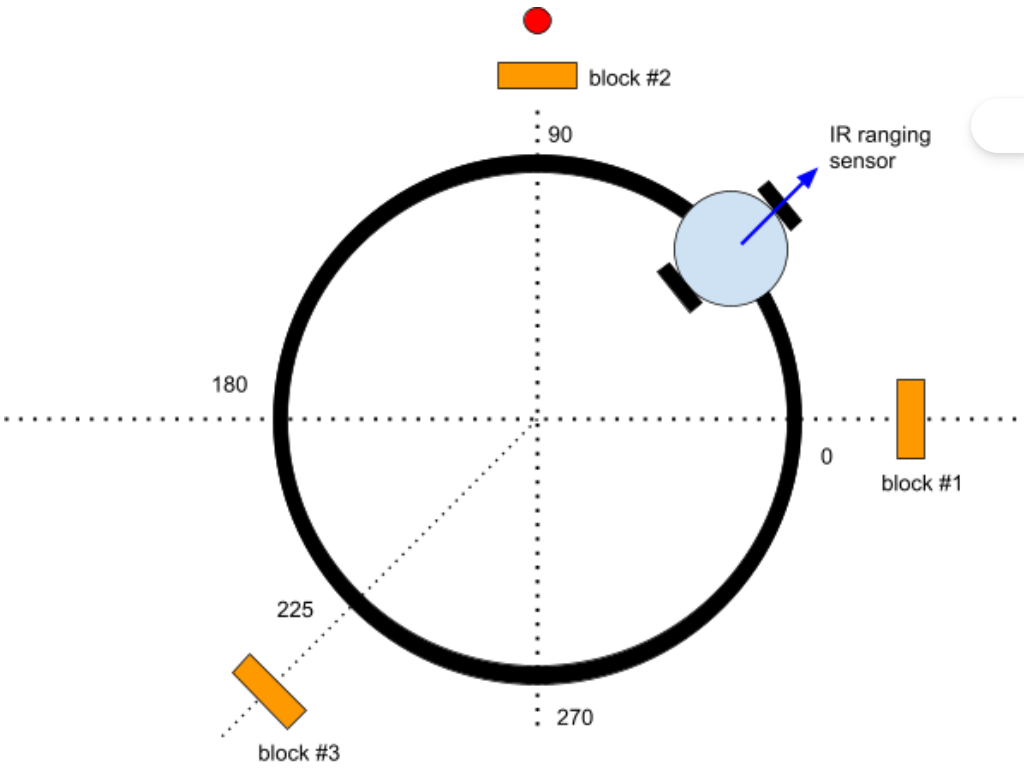
- The code lists having 2 encoders, but you can use just 1 encoder

Once your encoders are functional, add the following code snippet to your program so that any edge transition will count as an encoder tick:

```
void init_encoder() {
    // enable encoder interrupts
    EIMSK = 0;
    EIMSK |= _BV(PCIE1) |
_BV(PCIE0);

    PCMSK1 |= _BV(PCINT13); //PB5
- digital 5
    PCMSK0 |= _BV(PCINT6);  //PE6
- digital 4

    // enable pullups
    PORTE |= _BV(PE6);
    PORTB |= _BV(PB5);
}

volatile uint16_t left_encoder =
0;
volatile uint16_t right_encoder =
0;

ISR(PCINT0_vect) {
   left_encoder++;  //increment
left encoder
}

ISR(PCINT1_vect) {
   right_encoder++;  //increment
right encoder
}
```

CPE416 Fall 2023 - Lab 4

arranged in the following layout (the picture show an example):

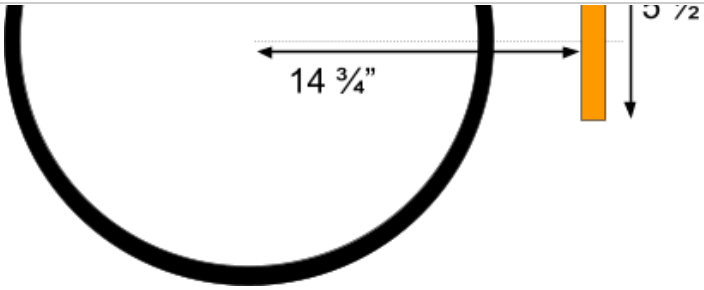


The orange blocks are 2x6 blocks of wood and the black line is electrical tape arranged in a circle. At the start of a run, you will be told the following numbers:

- `number of towers (from 2 to 4)`
- `tower locations (counterclockwise in order)`
- `target tower (tower number to knock over)`

For example, the diagram above represents the configuration:

```
3   (number of towers)
0   (tower #1)
90  (tower #2)
225 (tower #3)
2   (target tower)
```



The task for your robot is to drive around the circle (following the line) and once the robot has localized itself, it should turn and knock over the target block. At the start of a run, your robot will start in a random location on the circle.

- Mount the IR sensor on your robot and point it towards the outside of the circle
- The IR sensor has 3 leads and should be wired:
  - red - 5V
  - black - Ground
  - white or yellow - Analog input
- The datasheet for the Sharp IR sensor is here

Implementation Notes

Hints and Tips:

- The motion model should incorporate normally (Gaussian) distributed motion noise
- The sensor model describes the probability `p(z|x)` – the probability that the robot will get a sensor reading `z` given that the robot is at location `x`
  - use the trapezoidal distribution for each of the 2 sensor models
  - you will need to develop 2 sensor models (1 for block and 1 for free space)
- You may drive the robot a fixed distance and stop (to take a sensor reading and perform computation)
- Write your code so that it is scalable in the number of particles (start with 100 particles)
- During an incorrect localization, the particles may converge to the wrong location. To allow for recovery, introduce random particles during each iteration.

Demo

The demo of this lab will consist of the following:

- Demo of a simulated version of the assignment - a C program you

CPE416 Fall 2023 - Lab 4