Rotary Encoder

Introduction:

in the arduino code we are calculating the number of rotations of a rotary encoder from A and B pins

code:

```
const int encoderPinA = 2;
const int encoderPinB = 3;

int count = 0;
```

here we are starting the code with identifying the variables we are going to use:

- encoder pin A at pin 2 to allow interupt
- encoder pin B at pin 3 also to allow use of interupt
- and identifying the count to track number of revolutions

Setup and Loop:

```
void setup() {
  pinMode(encoderPinA, INPUT_PULLUP);
  pinMode(encoderPinB, INPUT_PULLUP);

  attachInterrupt(digitalPinToInterrupt(encoderPinA), updateEncoderA, CHANGE);
  attachInterrupt(digitalPinToInterrupt(encoderPinB), updateEncoderB, CHANGE);

  Serial.begin(9600); // Initialize the Serial Monitor
```

```
}

void loop() {
 // handled by interrupts
}
```

1. in the setup we are identifing the encoder pins as input pullup.
2. next we are initiating the interupts to use them to count and to start when we recive input from encoder A or B.
3. we start the serial monitor
4. lastly we dont use the void loop because the looping and the main code is handeled by the interrupt

Encoder Functions:

```
void updateEncoderA() {
  if (digitalRead(encoderPinA) != digitalRead(encoderPinB)) {
      count++;
    } else {
      count--;

  }


  // Print the count to the Serial Monitor
  Serial.print("Count: ");
  Serial.println(count);
}

void updateEncoderB() {
  if (digitalRead(encoderPinA) != digitalRead(encoderPinB)) {
    count++;
  } else {
    count--;
  }
  // Print the count to the Serial Monitor
  Serial.print("Count2: ");
  Serial.println(count);
}
```

- in these functionsn we control the count in encoder A function we check if the pin A value isnt equal to pin B we increase the count else we decrease it
- in Encoder B function we check for the same condition and in both we write the count in the serial monitor

Low Pass Filter (LPF):

Step 1: Calculate Maximum Angular Velocity (ωmax)

- The circumference of the wheel (C) is given by:
  $C = \pi * D$
  $C = \pi * 0.4$ meters = 1.26 meters
- The angular velocity (ω) is given by:
  $\omega = v\_max / R$
  $\omega = 0.5$ m/s / 0.2 meters = 2.5 radians/second

Step 2: Calculate Maximum Angular Frequency (ωmax)

- $\omega max = 2\pi * \omega$
- $\omega max = 2\pi * 2.5$ radians/second = 15.71 radians/second

Step 3: Calculate Maximum Frequency of Encoder Pulses (fmax)

- $fmax = (\omega max / (2\pi))$ * pulses per revolution
- $fmax = (15.71 / (2\pi)) * 540$ pulses = 1339.29 Hz

Step 4: Apply the Nyquist Theorem

- According to Nyquist, the sampling frequency (fs) should be at least twice fmax for accurate signal processing.
- $fs \geq 2 * fmax$
- $fs \geq 2 * 1339.29$ Hz
- $fs \geq 2678.58$ Hz

Step 5: Choose the Cutoff Frequency (fc)

- To design your LPF, you can choose a cutoff frequency (fc) that is a fraction of the sampling frequency (fs). A common choice is to use about 80% of fs for filtering.
- fc = 0.8 * fs
- fc = 0.8 * 2678.58 Hz = 2142.87 Hz

The LPF cutoff frequency should be 2142.97 Hz