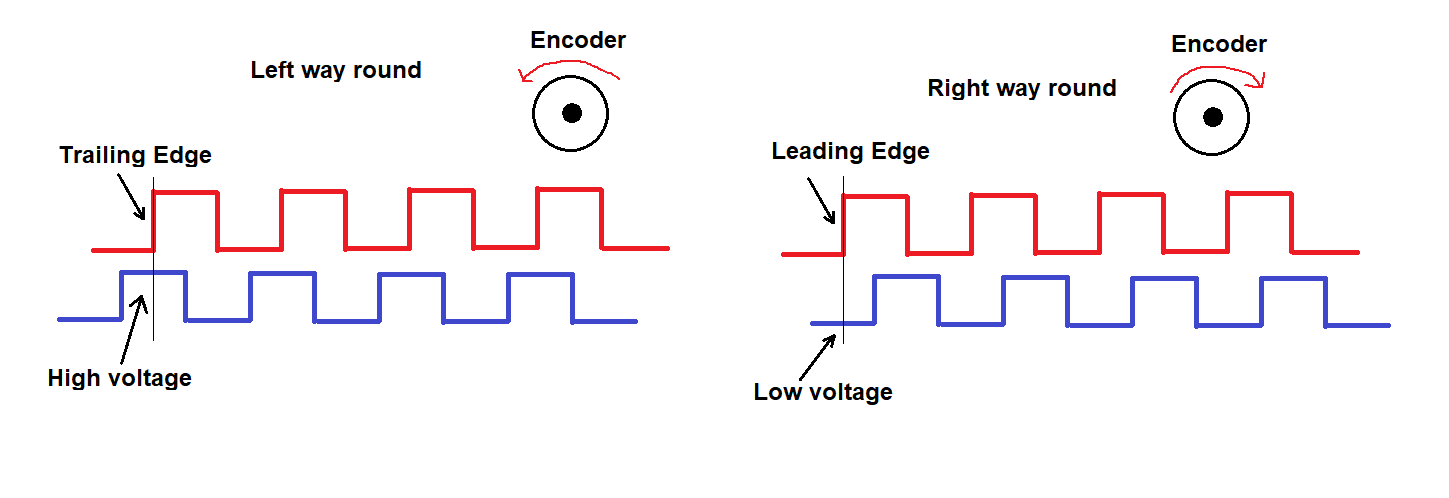
[2: Sensorer og aktuatorer](https://www.moodle.aau.dk/course/view.php?id=46579" \l "section-3)

**Formål**: Formålet her er at læse en encoders output. En encoder er lavet sådan at der kommer to pulser ud som er faseforskudt, således at når den ene er høj, er den anden altid lav. Ved at kigge på Rising edge af det ene signal, og derefter så på om det andet signal er højt eller lavt, kan man aflæse om encoderen køre den ene vej rundt eller den anden.



Som det kan ses på ovenstående figur, vil den blå altid være af lav spænding, i tilfældet af en rising edge på den røde kurve, i tilfælde af at encoderen køre venstre om. Kører den højre om, vil den på altid være lav når en rising edge på den røde registreres.

**Materialer:**

1. 2x Funktionsgenerator
2. Arduino UNO
3. Kabler
4. Oscilloskop

**Forsøgsopstilling:**

Diagram

Description automatically generated

Opstillingen som kan ses på ovenstående billede, “simulere” en encoder. Her er to generatorer låst sammen til samme frekvens, hvor den ene så er sat til at have en fase på 90 grader. Derved får man signalet vist under formål.

**Fremgangsmåde:**

1. Determine through measurement of voltage, the Steinhart-Hart parameters of a thermistor.
2. Create an Arduino program counting pulses (with sign) from the quadrature encoder on the presented setup.
3. Test the program by rotating the wheel 30 revolutions CW followed by 30 revolutions CCW.

**Resultater:**

Arduinoen kunne følge med, og tælle rigtigt antal pulser. Det begyndte dog at være Det et problem ved omkring100 Hz pulser. Dette skyldes at Serial.print taget lidt tid, og derfor kan det fungere bedre uden det.

**Konklusion:**

Det virkede, og vi så hvordan det er muligt at se retning og hastighed af en motor ved at bruge en encoder.

**Kode:**

**const int interPin = 2;**

**const int dirPen = 3;**

**float count = 0;**

**void setup() {**

**pinMode(interPin, INPUT);**

**pinMode(dirPen, INPUT);**

**attachInterrupt(digitalPinToInterrupt(interPin), pin\_ISR, RISING);**

**Serial.begin(115200);**

**}**

**void loop() {**

**}**

**void pin\_ISR() {**

**if(digitalRead(dirPen) == HIGH) {**

**Serial.println("Going Left");**

**count++;**

**}**

**else {**

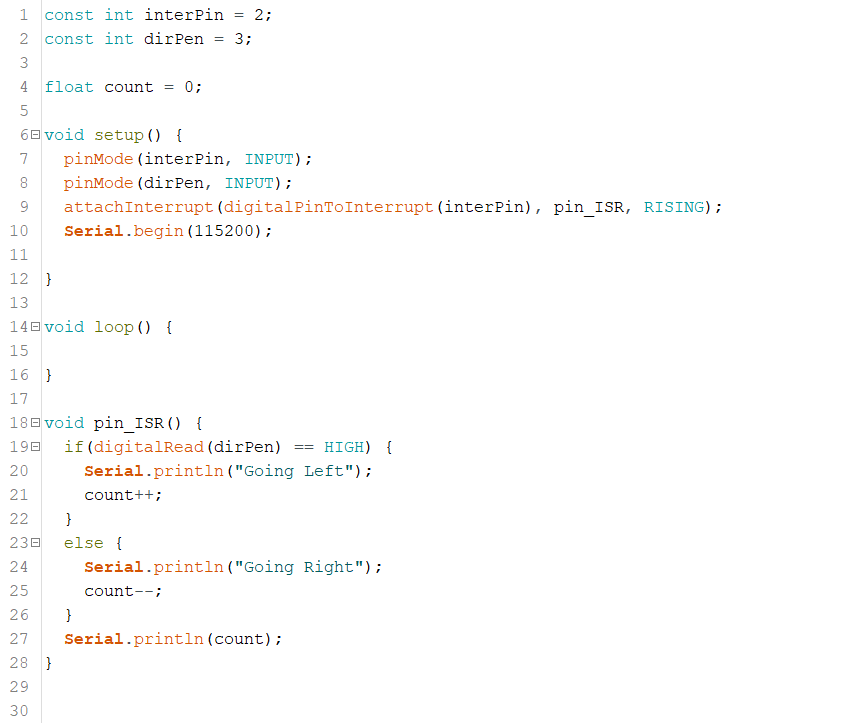
**Serial.println("Going Right");**

**count--;**

**}**

**Serial.println(count);**

**}**



Dette virker op til ca. 100 Hz, altså pulser der kommer med en frekvens på 100 Hz. Det at den er sat til at print hver gang gør den gevaldigt langsomere.