Geschwindigkeitsmessung mit dem RCWL-0516

(Radarfalle)

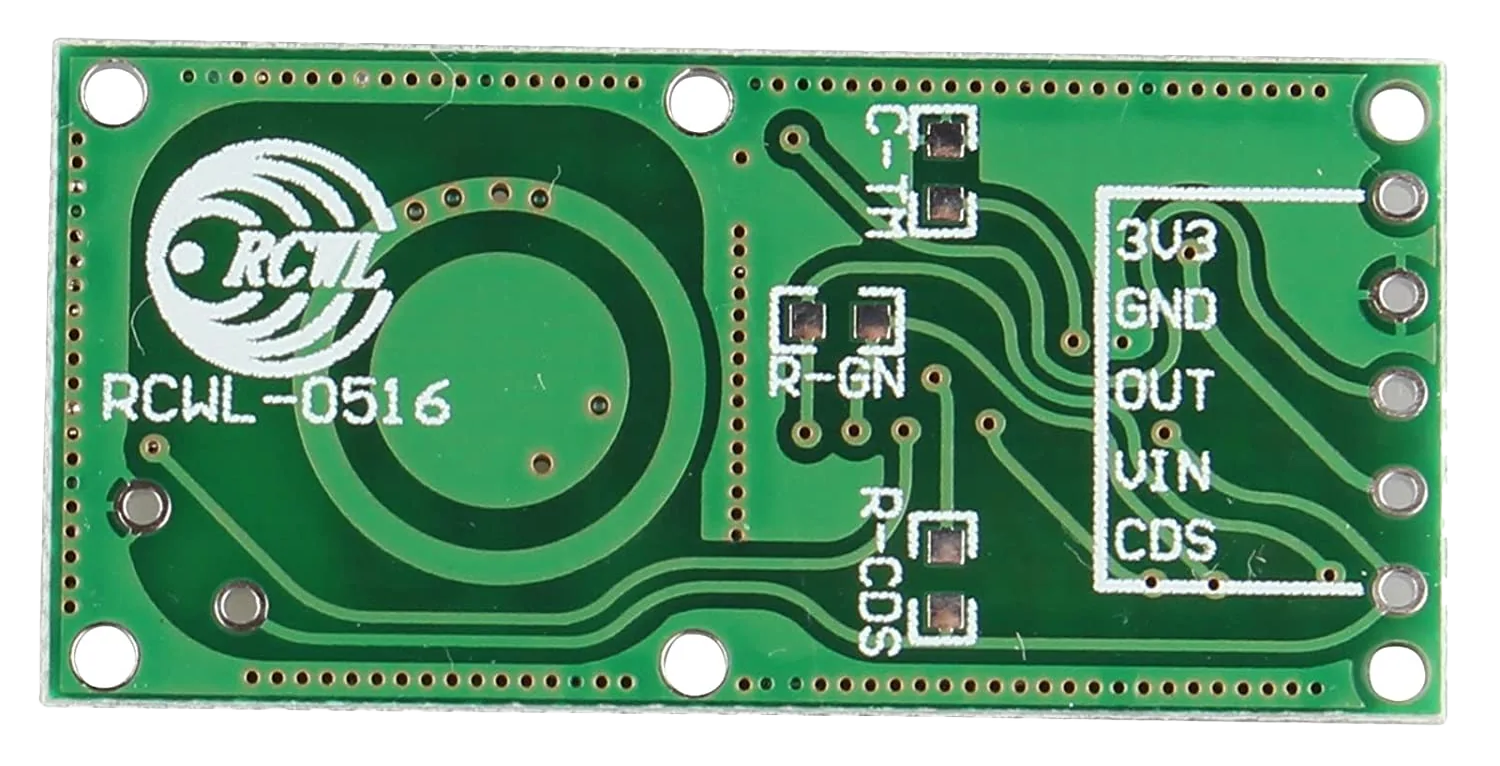
Projekt:

Es wurden 2 RCWL-0516 Mikrowellen-Radar Bauteile auf zwei Hölzerne Pfeiler platziert. Beide Module sind nach oben orientiert und deren Mitte ist 50cm weit voneinander entfernt. Durch das Auslösen des ersten Moduls (unabhängig welches der beiden) wird ein interrupt ausgelöst, in welchen ein temporärer Counter erhöht wird und das Interrupt für dieses Radarmodul wird deaktiviert. Falls der Counter 1 beträgt, wird in einer weiteren Funktion der Timer gestartet. Beim Auslösen des zweiten Moduls wird der Counter ein zweites Mal erhöht ( ctr = 2 ), der Timer wird gestoppt und die Geschwindigkeit wird berechnet. ( 50 / Zeit = Geschw. ). Diese variable wird noch mit 10 multipliziert, um die Zahl in Meter pro Sekunde zu ermitteln und wird auf dem LCD ausgegeben. Nach 3 Sekunden werden alle Werte zurückgesetzt und die Radarmodule wieder Aktiviert.

Nach der Aktivierung eines Moduls, wird nach maximal 3 Sekunden, wenn das zweite Modul nicht aktiviert wird, wird ebenfalls ein Reset durchgeführt.

Dieser ganze Prozess lässt sich über Bluetooth mit den Signalen „1“ aktivieren bzw. jedem anderen deaktivieren.

RCWL-0516:



3V3 – 3.3V output.

GND – Masse.

OUT – Ausgangspin (Digital).

VIN – 4-28V Spannung (Eingang).

CDS –

R-CDS – Mithilfe eines LDRs kann das Modul nur Nachtsüber funktionieren.

R-GN – Reichweite mit Widerstand variabel ( < 100kOhm = 0M ).

C-TM – Zeitspanne für ein HIGH Signal nachdem eine Bewegung entdeckt wurde.

Standard = 2 Sekunden

Code:

Deklarationen:

#include <SoftwareSerial.h>

#include <LiquidCrystal\_PCF8574.h>

#define R1 D6

#define R2 D7

#define BTRX D2 //BLE Recieve

#define BTTX D3 //BLE Transmit

SoftwareSerial SerialBT(BTRX, BTTX);

static HardwareTimer radarTimer = HardwareTimer(TIM2);

LiquidCrystal\_PCF8574 lcd (0x27);

Setup();

SerialBT.begin(9600);

Serial.begin(115200);

pinMode(R1, INPUT); pinMode(R2, INPUT);

radarTimer.setOverflow(100); //1ms

radarTimer.setPrescaleFactor(320); //1ms

radarTimer.attachInterrupt(ISR\_Timer);

attachInterrupt(digitalPinToInterrupt(R1), ISR\_R1\_IR, RISING); //rising = react if sensor becomes HIGH / pullup

attachInterrupt(digitalPinToInterrupt(R2), ISR\_R2\_IR, RISING);

I2C\_LCD\_init();

Interrupt Radar 1 und 2:

void ISR\_R1\_IR(void)

{

detachInterrupt(R1);

if (lock == 1){

ctr++;

}

Serial.println("R1 HIGH");

calcTime(); //geschwindigkeitsberechnung durch zeit

}

Berechnung, sowie Error Ausgabe bei zu schneller Reaktion:

if(timePassed < 100){

I2C\_LCD(cM);

delay(500);

I2C\_LCD("ERROR");

Serial.println("ERROR -> Accidental Trigger?");

}

else {

Serial.print("speed: ");

Serial.println(speed); //speed is calculated in meter / s

Serial.print("time: ");

Serial.println(timePassed);

I2C\_LCD(cM);

delay(500);

I2C\_LCD(String(speed) + " m/s");

delay(2000); //temp value to wait until reset

}

Links:  
<https://github.com/Hannah-Ga/DUAL-RCWL-0516-SPEED-MEASUREMENT/tree/main>

<https://www.radartutorial.eu/11.coherent/pic/phasenlage.print.png>

<https://www.radartutorial.eu/11.coherent/pic/phasenlage.print.png>