

# Infrarotsteuerung

Ein Infrarotsensor kann zum Erkennen von Hindernissen oder der Verfolgung einer schwarzen Linie genutzt werden. Die folgende Klasse steuert den Sensor.

```
from machine import Pin,Timer

import micropython

micropython.alloc_emergency_exception_buf(100)

class IR:
    """This class manages the IR-sensor. Write your code in Robot.ir_detected()"""
    def __init__(self, pinNo,callback_func):
        self.out = pinNo
        self.ir_detected = callback_func
        self.ir = Pin(pinNo, Pin.IN, Pin.PULL_UP)
        self.ir.irq(trigger=Pin.IRQ_FALLING | Pin.IRQ_RISING, handler=self.obstacle)
        self.detected=False
        self.timer = Timer()

    def reset_detected(self,t):
        self.detected = False

    def obstacle(self, pin):
        """This is called on any change in the IR-sensor. """
        if not self.detected:
            self.ir_detected(pin,self.out)
            self.detected = True
            self.timer.init(mode=Timer.ONE_SHOT, period=100, callback=self.reset_detected)
```

Der Code muss unter dem Dateinamen „infrared.py“ auf dem Pico gespeichert werden.

Der Infrarotsensor wird folgendermaßen im eigenen Programm eingebunden. Das Beispiel ist für zwei Infrarotsensoren.

```
from infrared import IR

IR_PIN_LEFT=0
```

```
IR_PIN_RIGHT=1
```

```
def ir_detected(pin, pinno):  
    print(f"Pin: {pin}, pin number: {pinno}")  
    if pinno == IR_PIN_LEFT:  
        print("links")  
    elif pinno == IR_PIN_RIGHT:  
        print("right")  
  
ir_left = IR(0, ir_detected)  
ir_right = IR(1, ir_detected)
```

---

Revision #9

Created 28 May 2024 16:30:47 by Marcus Jacobs

Updated 17 December 2024 11:50:05 by Marcus Jacobs