

Micropython Timer

- A timer allows a function to be called at a specified rate or at some later time.
- To use a timer, you need to import the Timer.
- You create a timer and configure it.
- The period is in milliseconds.
- Mode is set to periodic so it repeats.
- Callback is set to the name of the function to execute when time occurs.
- Enter this program. Call it **rover10.py**
- Run the program. The callback function should execute five times for every time the while loop prints its message.
- You can try changing the period of the timer.

```
from machine import Timer
import time

def callback(e):
    print("This is a timer")

timer = Timer(period=200,mode=Timer.PERIODIC,
              callback=callback)

while 1:
    print("Loop")
    time.sleep(1)
```

- Let's use the timer to collect and print sensor data. Create a new program on the robot and call it **rover10a.py**.
- Let's use the ultrasonic sensor, light sensor, and temperature sensor.
- The ADC is imported to read the light and temperature sensor.
- ultra is imported to measure distance with the ultrasonic sensor.
- The callback function named **get_sensors()** is declared. This function will contain the code to get the distance measurement and the light and temperature measurements.
- A variable **msg** will contain the values from the sensors.

```
from machine import Pin,Timer,ADC
import time
import ultra

ts = ADC(Pin(28))
light = ADC(Pin(27))

def get_sensors(e):
    global ts,light
    distance = ultra.ultra()
    adc = ts.read_u16()
    v = adc/65536.0 * 3.3
    temp = -29.78 * v + 73.97
    adc = light.read_u16()
    v = adc/65536.0 * 3.3
    msg = "Data %f %f %f" % (distance,temp,v)
    print(msg)

timer = Timer(period=200,mode=Timer.PERIODIC,
              callback=get_sensors)

while 1:
    time.sleep(1)
```

- the string variable msg is created with formatted text. The formatting contains two parts.
- In the set of quotes is the formatting of the string. You can include text anywhere in there. To include the value of the variables, you use the % followed by a letter that indicates the type of data.
 - %f is for floating point values.
 - %d is for integer values.
 - %s is for string values.
- After the quotes, a % is added and then the list of variables within ().
- You have to list the variables in the order you want them put into the string.

"text here %d %f" % (val1,val2)

- Run the program.
- The sensor data should be displayed five times a second.

```
from machine import Pin,Timer,ADC
import time
import ultra

ts = ADC(Pin(28))
light = ADC(Pin(27))

def get_sensors(e):
    global ts,light
    distance = ultra.ultra()
    adc = ts.read_u16()
    v = adc/65536.0 * 3.3
    temp = -29.78 * v + 73.97
    adc = light.read_u16()
    v = adc/65536.0 * 3.3
    msg = "Data %f %f %f" % (distance,temp,v)
    print(msg)

timer = Timer(period=200,mode=Timer.PERIODIC,
              callback=get_sensors)

while 1:
    time.sleep(1)
```

