

WOKWI

SAVE SHARE

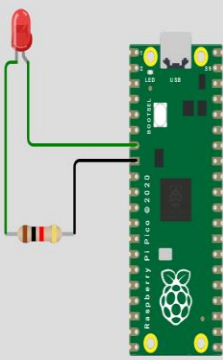
Blink with MicroPython by urish

Docs SIGN IN

main.py diagram.json Library Manager

```
1 from machine import Pin
2 from time import sleep
3
4 led = Pin(5, Pin.OUT)
5 while True:
6     led.toggle()
7     sleep(0.5)
```

Simulation



32°C Mostly sunny

Search

ENG IN

09:22 AM 20-04-2025

WOKWI

SAVE SHARE

Blink with MicroPython by urish

Docs SIGN IN

main.py diagram.json Library Manager Simulation

```
1 from machine import Pin
2 from time import sleep
3
4 led = Pin(5, Pin.OUT)
5 while True:
6     led.toggle()
7     sleep(0.5)
```

Restart the simulation

00:02:199 101%

32°C Mostly sunny

Q Search

ENG IN

09:22 AM 20-04-2025

WOKWI

SAVE SHARE

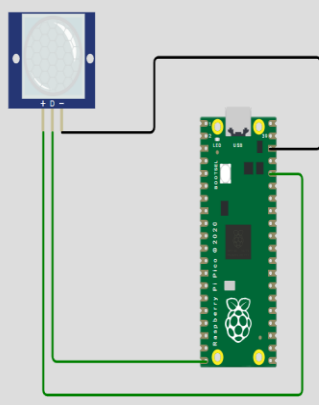
Docs SIGN UP

main.py diagram.json

```
1 from machine import Pin
2 import time
3
4 pir = Pin(15, Pin.IN)
5
6 while True:
7     if pir.value():
8         print("Motion Detected!")
9     else:
10        print("No Motion")
11        time.sleep(1)
12
```

Simulation

00:08.456 99%



No Motion  
No Motion  
No Motion  
No Motion  
No Motion  
No Motion  
No Motion  
No Motion  
No Motion  
No Motion

Q Search

ENG IN 08:22 PM 24-04-2025

WOKWI

SAVE SHARE Docs SIGN UP

main.py diagram.json

```
1 from machine import Pin
2 import time
3
4 pir = Pin(15, Pin.IN)
5
6 while True:
7     if pir.value():
8         print("Motion Detected!")
9     else:
10        print("No Motion")
11    time.sleep(1)
12
```

Simulation

PIR Motion Sensor

Simulate motion

No Motion  
No Motion  
No Motion  
No Motion  
No Motion  
No Motion  
No Motion  
No Motion  
Motion Detected!  
Motion Detected!  
Motion Detected!

00:16.979 100%

08:22 PM 24-04-2025

WOKWI

SAVE SHARE

Docs SIGN UP

main.py diagram.json

```
1 import time
2 import dht
3 from machine import Pin
4
5 sensor = dht.DHT22(Pin(15))
6
7 while True:
8     sensor.measure()
9     temp = sensor.temperature()
10    hum = sensor.humidity()
11    print("Temperature:", temp, "°C")
12    print("Humidity:", hum, "%")
13    time.sleep(2)
14
```

Simulation

00:13.836 100%

Editing DHT22

Temperature: 17.4°C

Humidity: 40.0%

Temperature: 61.5 °C  
Humidity: 40.0 %  
Temperature: 61.5 °C  
Humidity: 40.0 %  
Temperature: 8.0 °C  
Humidity: 40.0 %  
Temperature: 17.4 °C  
Humidity: 40.0 %  
Temperature: 17.4 °C  
Humidity: 40.0 %

08:33 PM  
24-04-2025

WOKWI

SAVE SHARE

Docs SIGN IN

main.py diagram.json

```
1 import time
2 import dht
3 from machine import Pin
4
5 sensor = dht.DHT22(Pin(15))
6
7 while True:
8     sensor.measure()
9     temp = sensor.temperature()
10    hum = sensor.humidity()
11    print("Temperature:", temp, "°C")
12    print("Humidity:", hum, "%")
13    time.sleep(2)
14
```

Simulation

00:22.818 99%

Editing DHT22

Temperature: 42.0°C

Humidity: 78.0%

Temperature: 17.4 °C  
Humidity: 40.0 %  
Temperature: 17.4 °C  
Humidity: 40.0 %  
Temperature: 17.4 °C  
Humidity: 40.0 %  
Temperature: 42.0 °C  
Humidity: 85.0 %  
Temperature: 42.0 °C  
Humidity: 78.0 %

ENG IN 08:33 PM 24-04-2025

WOKWI

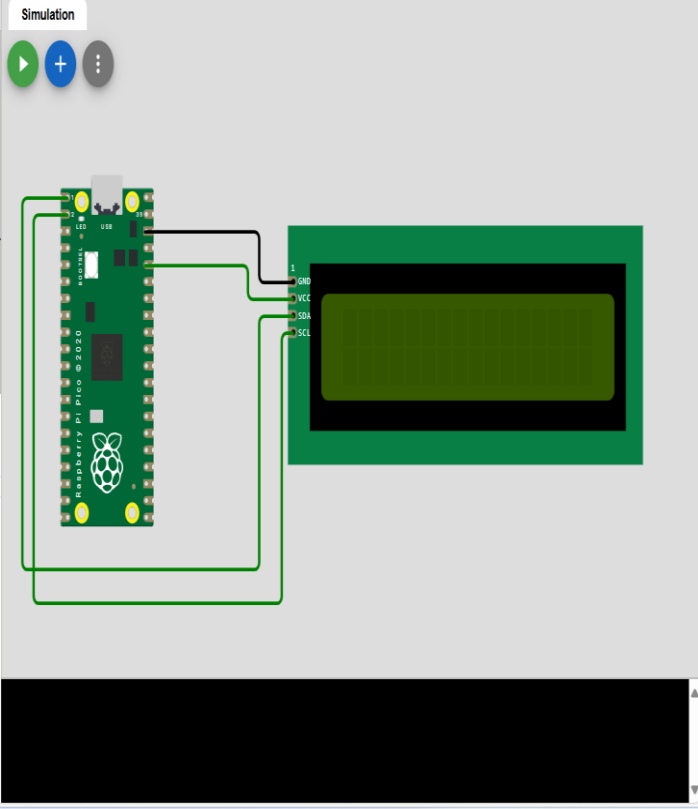
SAVE SHARE

Docs SIGN UP

main.py diagram.json

```
1 from machine import Pin, I2C
2 from time import sleep, sleep_ms, localtime
3 LCD_ADDR = 0x27
4 LCD_WIDTH = 16
5 LCD_LINE_1 = 0x80
6 LCD_LINE_2 = 0xC0
7 LCD_BACKLIGHT = 0x08
8 ENABLE = 0b00000100
9 i2c = I2C(0, scl=Pin(1), sda=Pin(0), freq=400000)
10 def lcd_send_byte(bits, mode):
11     high = mode | (bits & 0xF0) | LCD_BACKLIGHT
12     low = mode | ((bits << 4) & 0xF0) | LCD_BACKLIGHT
13     for val in [high, high | ENABLE, high, low, low | ENABLE, low]:
14         i2c.writeto(LCD_ADDR, bytearray([val]))
15         sleep_ms(1)
16 def lcd_init():
17     sleep_ms(20)
18     lcd_send_byte(0x33, 0)
19     lcd_send_byte(0x32, 0)
20     lcd_send_byte(0x28, 0)
21     lcd_send_byte(0x0C, 0)
22     lcd_send_byte(0x06, 0)
23     lcd_send_byte(0x01, 0)
24     sleep_ms(2)
25 def lcd_clear():
26     lcd_send_byte(0x01, 0)
27     sleep_ms(2)
28 def lcd_write(message, line):
29     lcd_send_byte(line, 0)
30     for char in message:
31         lcd_send_byte(ord(char), 1)
32 lcd_init()
33 while True:
34     t = localtime()
35     time_str = "{:02}:{:02}:{:02}".format(t[3], t[4], t[5])
```

Simulation



GSW - HOU  
Video highlight

Search

ENG IN 09:34 PM 24-04-2025

WOKWI

SAVE SHARE

Docs SIGN UP

main.py diagram.json

```
1 from machine import Pin, I2C
2 from utime import sleep, sleep_ms, localtime
3 LCD_ADDR = 0x27
4 LCD_WIDTH = 16
5 LCD_LINE_1 = 0x80
6 LCD_LINE_2 = 0xC0
7 LCD_BACKLIGHT = 0x08
8 ENABLE = 0b00000100
9 i2c = I2C(0, scl=Pin(1), sda=Pin(0), freq=400000)
10 def lcd_send_byte(bits, mode):
11     high = mode | (bits & 0xF0) | LCD_BACKLIGHT
12     low = mode | ((bits << 4) & 0xF0) | LCD_BACKLIGHT
13     for val in [high, high | ENABLE, high, low, low | ENABLE, low]:
14         i2c.writeto(LCD_ADDR, bytearray([val]))
15         sleep_ms(1)
16 def lcd_init():
17     sleep_ms(20)
18     lcd_send_byte(0x33, 0)
19     lcd_send_byte(0x32, 0)
20     lcd_send_byte(0x28, 0)
21     lcd_send_byte(0x0C, 0)
22     lcd_send_byte(0x06, 0)
23     lcd_send_byte(0x01, 0)
24     sleep_ms(2)
25 def lcd_clear():
26     lcd_send_byte(0x01, 0)
27     sleep_ms(2)
28 def lcd_write(message, line):
29     lcd_send_byte(line, 0)
30     for char in message:
31         lcd_send_byte(ord(char), 1)
32 lcd_init()
33 while True:
34     t = localtime()
35     time_str = "{:02}:{:02}:{:02}".format(t[3], t[4], t[5])
```

Simulation

00:08.650 99%

GSW - HOU  
Video highlight

Search

ENG IN 09:35 PM 24-04-2025