Table of Contents

1. What is MicroPython?

- MicroPython is a compact version of Python 3 optimized to run on microcontrollers.
- Boards Supported:
  - ESP8266 / ESP32
  - Raspberry Pi Pico
  - PyBoard
  - STM32, RP2040, etc.

---

2. Installation (Microcontroller Setup)

Flash MicroPython Firmware (ESP32/ESP8266):

1. Download Firmware:
   https://micropython.org/download/
2. Install esptool:

   ```
   pip install esptool
   ```

3. Erase Flash:

   ```
   esptool.py --port /dev/ttyUSB0 erase_flash
   ```

4. Flash Firmware:

   ```
   esptool.py --port /dev/ttyUSB0 write_flash -z 0x1000 firmware.bin
   ```

---

3. Connecting to MicroPython

REPL (Read Evaluate Print Loop):

```
screen /dev/ttyUSB0 115200  # Linux/Mac
putty (COM port)            # Windows
```

- Exit REPL: Ctrl + A + K (or unplug/reset)

---

4. Basic MicroPython Workflow

1. Connect to Board via USB.
2. Flash Firmware.
3. Use REPL for Testing.
4. Upload Scripts (via rshell/ampy).

---

5. Writing and Uploading Code

1. Create Python File (main.py):

```
print("Hello, MicroPython!")
```

2. Upload to Board:

```
ampy --port /dev/ttyUSB0 put main.py
```

3. Run Code on Boot (Place in main.py or boot.py).

---

6. GPIO (Pin Control)

```
from machine import Pin
led = Pin(2, Pin.OUT)  # GPIO2 (ESP32)
led.value(1)           # Turn LED On
led.value(0)           # Turn LED Off
```

Button Input:

```
button = Pin(12, Pin.IN, Pin.PULL_UP)  # Pull-up resistor
if button.value() == 0:
    print("Button Pressed")
```

---

7. Blinking LED (Basic Project)

```
from machine import Pin
import time

led = Pin(2, Pin.OUT)  # GPIO2 (Built-in LED on ESP32)

while True:
```

```
    led.value(1)
    time.sleep(1)
    led.value(0)
    time.sleep(1)
```

---

8. PWM (Pulse Width Modulation)

```
from machine import Pin, PWM
led = PWM(Pin(2), freq=500, duty=512)  # 50% duty cycle
led.duty(1023)  # Full brightness
led.duty(0)     # Off
```

---

9. Analog to Digital Conversion (ADC)

```
from machine import ADC, Pin
adc = ADC(Pin(36))  # GPIO36 (ADC1)
adc.atten(ADC.ATTN_11DB)  # Max voltage 3.6V
value = adc.read()
print(value)
```

---

10. I2C (Sensor Communication)

```
from machine import Pin, I2C

i2c = I2C(0, scl=Pin(22), sda=Pin(21))  # ESP32 Pins
devices = i2c.scan()
print(devices)  # List of detected devices
```

---

11. SPI (Serial Peripheral Interface)

```
from machine import SPI, Pin

spi = SPI(1, baudrate=1000000, polarity=0, phase=0, sck=Pin(18),
mosi=Pin(23), miso=Pin(19))
```

---

12. UART (Serial Communication)

```
from machine import UART
uart = UART(1, baudrate=9600, tx=17, rx=16)
uart.write("Hello")
response = uart.read()
print(response)
```

---

13. Reading Sensors (DHT11 Example)

```
import dht
from machine import Pin

sensor = dht.DHT11(Pin(4))  # GPIO4
sensor.measure()
print("Temp:", sensor.temperature())
print("Humidity:", sensor.humidity())
```

---

14. Network (WiFi Connection ESP32/ESP8266)

```
import network

wifi = network.WLAN(network.STA_IF)
wifi.active(True)
```

```
wifi.connect("YourSSID", "YourPassword")

while not wifi.isconnected():
    pass

print("Connected, IP:", wifi.ifconfig())
```

---

15. Web Server (ESP32/ESP8266)

```
import socket
import network

wifi = network.WLAN(network.STA_IF)
wifi.active(True)
wifi.connect("YourSSID", "YourPassword")

while not wifi.isconnected():
    pass

s = socket.socket()
s.bind(('', 80))
s.listen(5)

while True:
    conn, addr = s.accept()
    conn.send("HTTP/1.1 200 OK\r\nContent-Type:
text/html\r\n\r\nHello, World!")
    conn.close()
```

---

16. File System Commands (REPL)

```
import os
os.listdir()               # List files
os.remove('main.py')       # Delete file
```

```
f = open('test.txt', 'w')  # Create file
f.write('Hello')
f.close()
```

---

17. Power Saving (Deep Sleep)

```
from machine import deepsleep
deepsleep(10000)  # Sleep for 10 seconds
```

---

18. Firmware Management

- Erase Flash:

  ```
  esptool.py erase_flash
  ```

- Reflash Firmware:

  ```
  esptool.py write_flash -z 0x1000 firmware.bin
  ```

---

19. Upload Files to MicroPython

1. Install ampy:

   ```
   pip install adafruit-ampy
   ```

2. Upload Python File:

   ```
   ampy --port /dev/ttyUSB0 put main.py
   ```

3. List Files:

   ```
   ampy --port /dev/ttyUSB0 ls
   ```

---

20. MicroPython Common Modules

- machine – Access GPIO, ADC, I2C, SPI, etc.
- network – Manage WiFi and networking.
- time – Handle delays and real-time clock.
- os – File system operations.
- dht – DHT11/DHT22 sensors.

Example: Control LED via Web (ESP32/ESP8266)

```
import network
import socket
from machine import Pin

led = Pin(2, Pin.OUT)
wifi = network.WLAN(network.STA_IF)
wifi.active(True)
wifi.connect("YourSSID", "YourPassword")

while not wifi.isconnected():
    pass

s = socket.socket()
s.bind(('', 80))
s.listen(5)

while True:
    conn, addr = s.accept()
    request = conn.recv(1024)
    led.value(not led.value())  # Toggle LED
    conn.send("HTTP/1.1 200 OK\r\nContent-Type: text/html\r\n\r\nLED
Toggled!")
    conn.close()
```