



DesignNews

Raspberry Pi Pico/Pico2 Development Using Visual Studio Code

Day 2:

Day of the Snake

Sponsored by

DigiKey



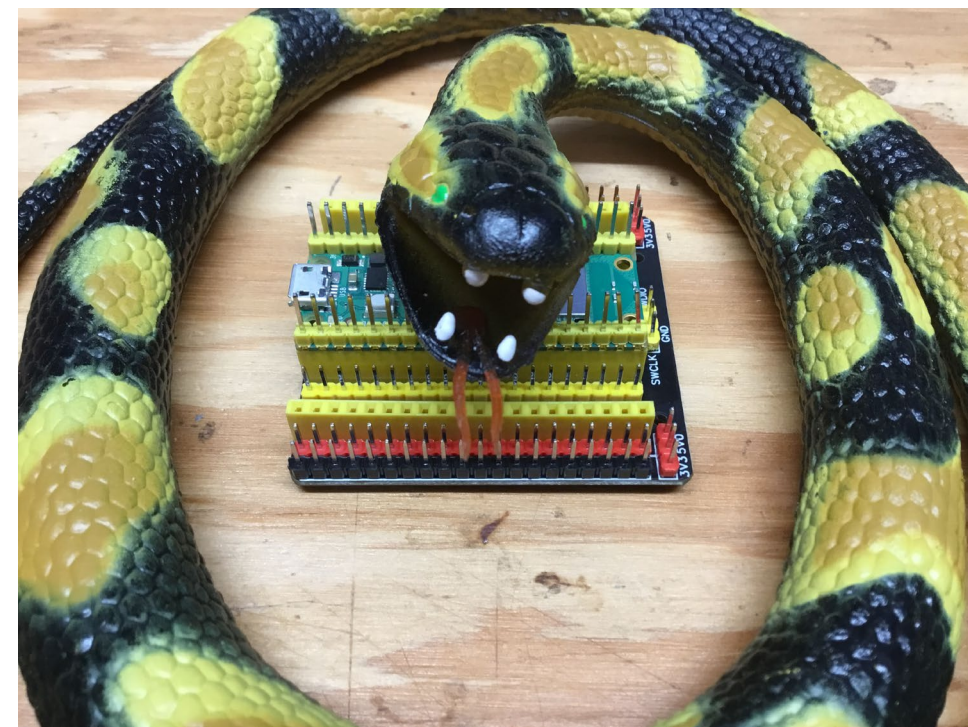
Webinar Logistics

- Turn on your system sound to hear the streaming presentation.
- If you have technical problems, click “Help” or submit a question asking for assistance.
- Participate in ‘Attendee Chat’ by maximizing the chat widget in your dock.



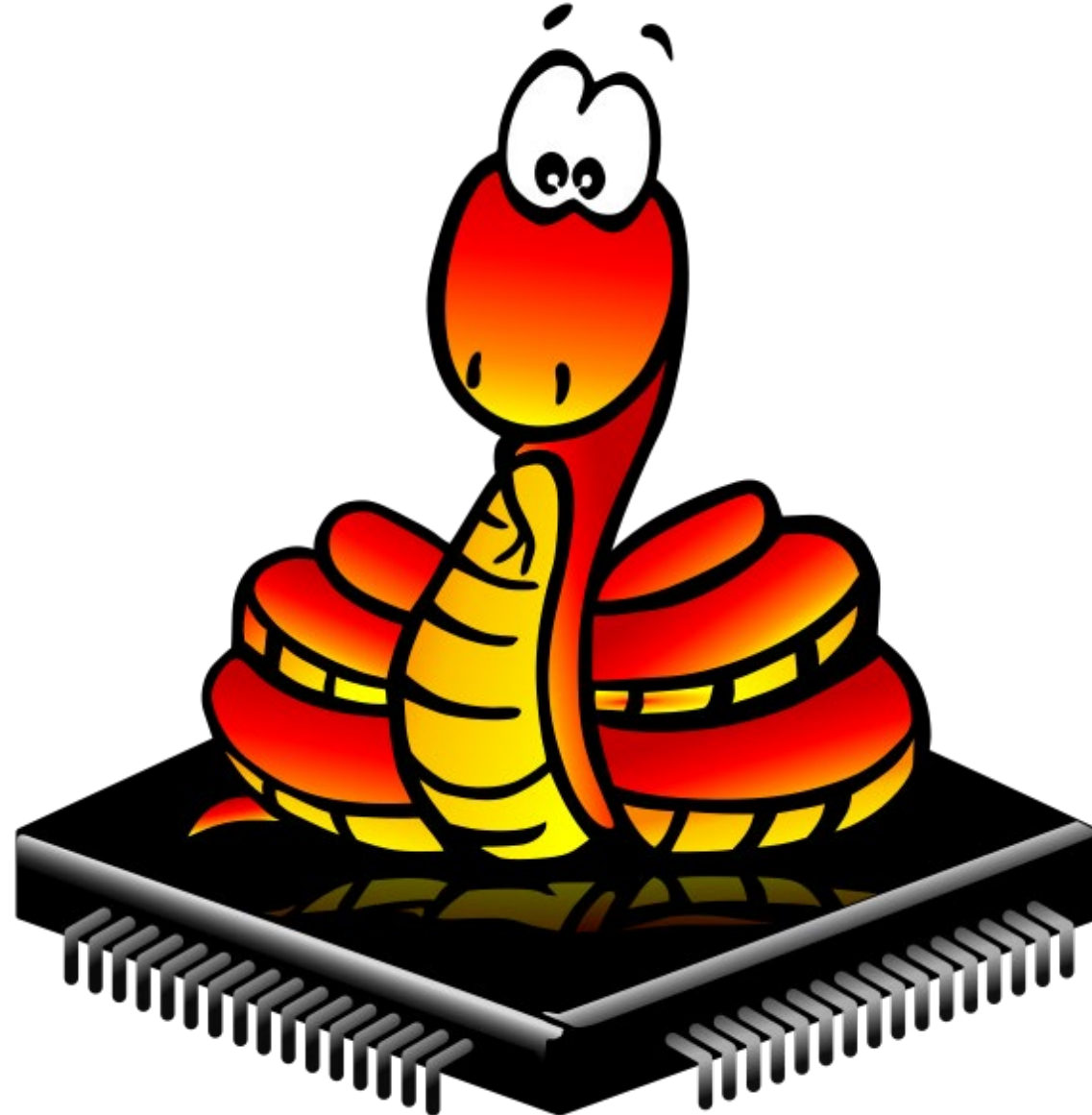
Fred Eady

Visit 'Lecturer Profile' in your console for more details.

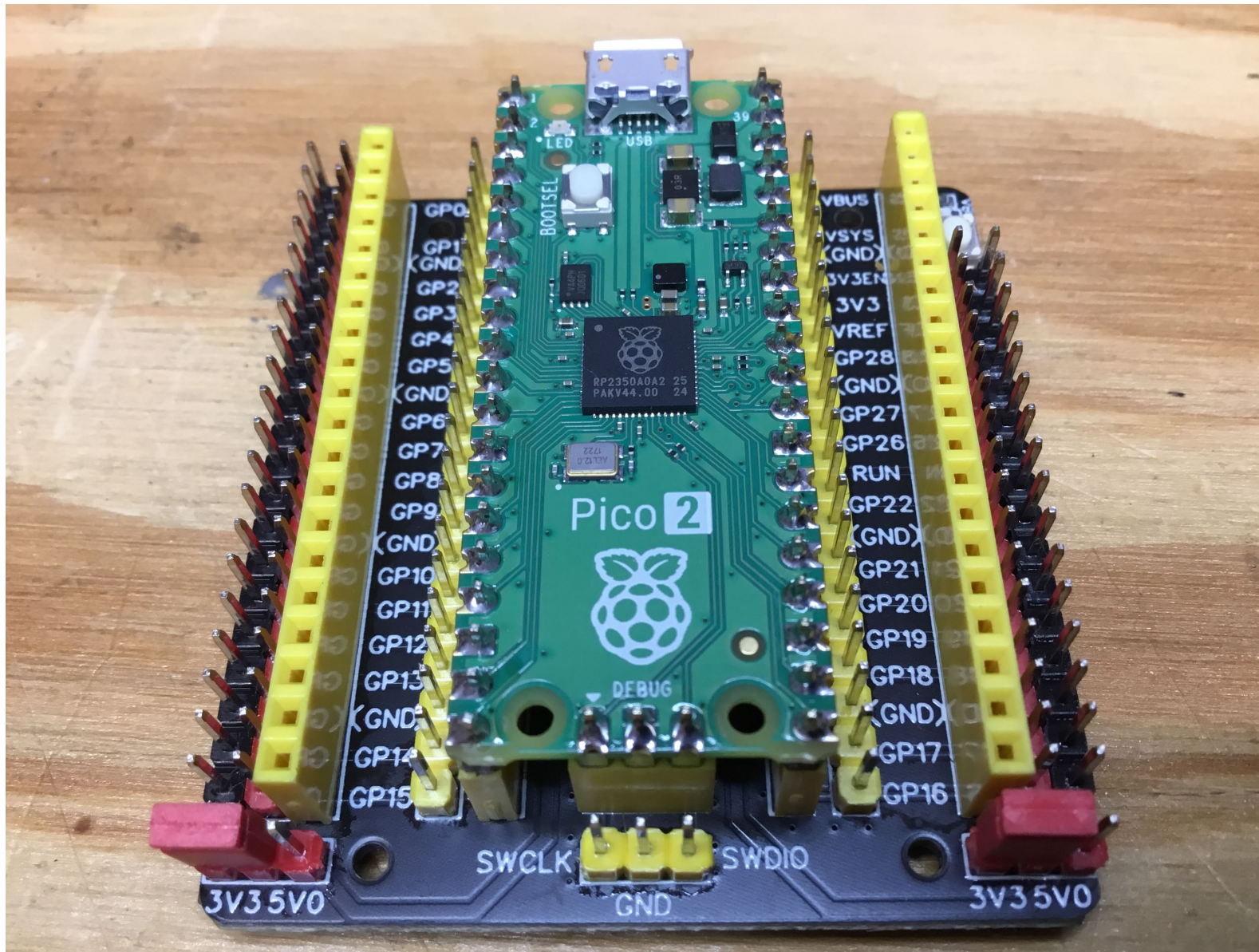


AGENDA

- **Snake Handling 101**
 - **Catch the Snake**
 - **Feed the Snake**
 - **Charm the Snake**



Catch the Snake



Train the Snake - Install MicroPython



MicroPython

DOWNLOAD

DOCS

DISCORD

DISCUSSIONS

WIKI

STORE

MicroPython downloads

MicroPython is developed using git for source code management, and the master repository can be found on GitHub at github.com/micropython/micropython.

The full source-code distribution of the latest version is available for download here:

- [micropython-1.24.1.tar.xz](#) (83MiB)

Daily snapshots of the GitHub repository (not including submodules) are available from this server:

- [micropython-master.zip](#)
- [pyboard-master.zip](#)

Firmware for various microcontroller ports and boards are built automatically on a daily basis and can be found below.

Filter by:

Port: [cc3200](#), [esp32](#), [esp8266](#), [mimxrt](#), [nrf](#), [renesas-ra](#), [rp2](#), [samd](#), [stm32](#)

Feature: [Audio Codec](#), [BLE](#), [Battery Charging](#), [CAN](#), [Camera](#), [DAC](#), [Display](#), [Dual-core](#), [Environment Sensor](#), [Ethernet](#), [External Flash](#), [External RAM](#), [Feather](#), [IMU](#), [JST-PH](#), [JST-SH](#), [LoRa](#), [Microphone](#), [PoE](#), [RGB LED](#), [SDCard](#), [Secure Element](#), [USB](#), [USB-C](#), [WiFi](#), [microSD](#), [mikroBUS](#)

Vendor: [Actinius](#), [Adafruit](#), [Arduino](#), [BBC](#), [Espressif](#), [Espruino](#), [Fez](#), [George Robotics](#), [HydraBus](#), [I-SYST](#), [LEGO](#), [LILYGO](#), [Laird Connectivity](#), [LimiFrog](#), [M5 Stack](#), [M5Stack](#), [Makerdiary](#), [McHobby](#), [Microchip](#), [MikroElektronika](#), [MiniFig Boards](#), [NXP](#), [Netduino](#), [Nordic Semiconductor](#), [OLIMEX](#), [PJRC](#), [Particle](#), [Pimoroni](#), [Pololu](#), [Pycom](#), [Raspberry Pi](#), [Renesas Electronics](#), [ST Microelectronics](#), [Seeed Studio](#), [Silicognition](#), [Silicognition LLC](#), [Sparkfun](#), [Unexpected Maker](#), [VCC-GND Studio](#), [Vekatech](#), [WeAct](#), [WeAct Studio](#), [Wemos](#), [Wireless-Tag](#), [Wiznet](#), [nullbits](#), [u-blox](#)

MCU: [RA6M5](#), [cc3200](#), [esp32](#), [esp32c3](#), [esp32c6](#), [esp32s2](#), [esp32s3](#), [esp8266](#), [mimxrt](#), [nrf51](#), [nrf52](#), [nrf91](#), [ra4m1](#), [ra4w1](#), [ra6m1](#), [ra6m2](#), [ra6m3](#), [rp2040](#), [rp2350](#), [samd21](#), [samd51](#), [stm32f0](#), [stm32f4](#), [stm32f411](#), [stm32f7](#), [stm32g0](#), [stm32g4](#), [stm32h5](#), [stm32l1](#), [stm32l0](#), [stm32l1](#), [stm32l4](#), [stm32wb](#), [stm32wl](#)

Train the Snake - Install MicroPython

MicroPython downloads

MicroPython is developed using git for source code management, and the master repository can be found on GitHub at github.com/micropython/micropython.

The full source-code distribution of the latest version is available for download here:

- [micropython-1.24.1.tar.xz](#) (83MiB)

Daily snapshots of the GitHub repository (not including submodules) are available from this server:

- [micropython-master.zip](#)
- [pyboard-master.zip](#)

Firmware for various microcontroller ports and boards are built automatically on a daily basis and can be found below.

Filter by:

Port: cc3200, esp32, esp8266, mimxrt, nrf, renesas-ra, rp2, samd, stm32

Feature: Audio Codec, BLE, Battery Charging, CAN, Camera, DAC, Display, Dual-core, Environment Sensor, Ethernet, External Flash, External RAM, Feather, IMU, JST-PH, JST-SH, LoRa, Microphone, PoE, RGB LED, SDCard, Secure Element, USB, USB-C, WiFi, microSD, mikroBUS

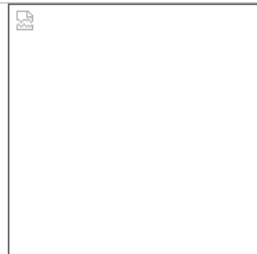
Vendor: Actinius, Adafruit, Arduino, BBC, Espressif, Espruino, Fez, George Robotics, HydraBus, I-SYST, LEGO, LILYGO, Laird Connectivity, LimiFrog, M5 Stack, M5Stack, Makerdiary, McHobby, Microchip, MikroElektronika, MiniFig Boards, NXP, Netduino, Nordic Semiconductor, OLIMEX, PJRC, Particle, Pimoroni, Pololu, Pycom, Raspberry Pi, Renesas Electronics, ST Microelectronics, Seeed Studio, Silicognition, Silicognition LLC, Sparkfun, Unexpected Maker, VCC-GND Studio, Vekatech, WeAct, WeAct Studio, Wemos, Wireless-Tag, Wiznet, nullbits, u-blox

MCU: rp2350 [x]

Click to Download MicroPython



Pico 2
[Raspberry Pi](#)



Pico 2 W
Raspberry Pi



Pro Micro RP2350
Sparkfun

Train the Snake - Install MicroPython

Vendor: Raspberry Pi

Features: Dual-core, External Flash, USB

Source on GitHub: [rp2/RPI_PICO2](#)

More info: [Website](#)

Installation instructions

Flashing via UF2 bootloader

To get the board in bootloader mode ready for the firmware update, execute `machine.bootloader()` at the MicroPython REPL. Alternatively, hold down the BOOTSEL button while plugging the board into USB. The uf2 file below should then be copied to the USB mass storage device that appears. Once programming of the new firmware is complete the device will automatically reset and be ready for use.

Firmware

Releases

v1.24.1 (2024-11-29) .uf2 / [\[Release notes\]](#) (latest)

[v1.24.0 \(2024-10-25\) .uf2](#) / [\[Release notes\]](#)

Preview builds

[v1.25.0-preview.270.gd76733d05 \(2025-02-07\) .uf2](#)

[v1.25.0-preview.267.ge44a2c692 \(2025-02-07\) .uf2](#)

[v1.25.0-preview.253.g71e8b27b2 \(2025-02-07\) .uf2](#)

[v1.25.0-preview.248.g81ab49a60 \(2025-02-07\) .uf2](#)

(These are automatic builds of the development branch for the next release)

Train the Snake - Install MicroPython

The screenshot shows the Visual Studio Code file explorer interface. The main window displays the file system for the RP2350 device. A sidebar on the left contains navigation options: Recent, Starred, Home, Documents, Downloads, Music, and Pictures. The main pane shows a list of files under the heading 'Name':

Name
INDEX.HTM
INFO_UF2.TXT
RPI_PICO2-20241129-v1.24.1.uf2

An inset window shows the 'Downloads' folder, with the breadcrumb 'Home / Downloads'. It contains a list of files:

Name
RPI_PICO2-20241129-v1.24.1.uf2
code_1.96.4-1736991114_amd64.deb
99-picotool.rules

Feed the Snake – Create a MicroPython Project

Raspberry Pi Pico Basic Settings

Basic Settings

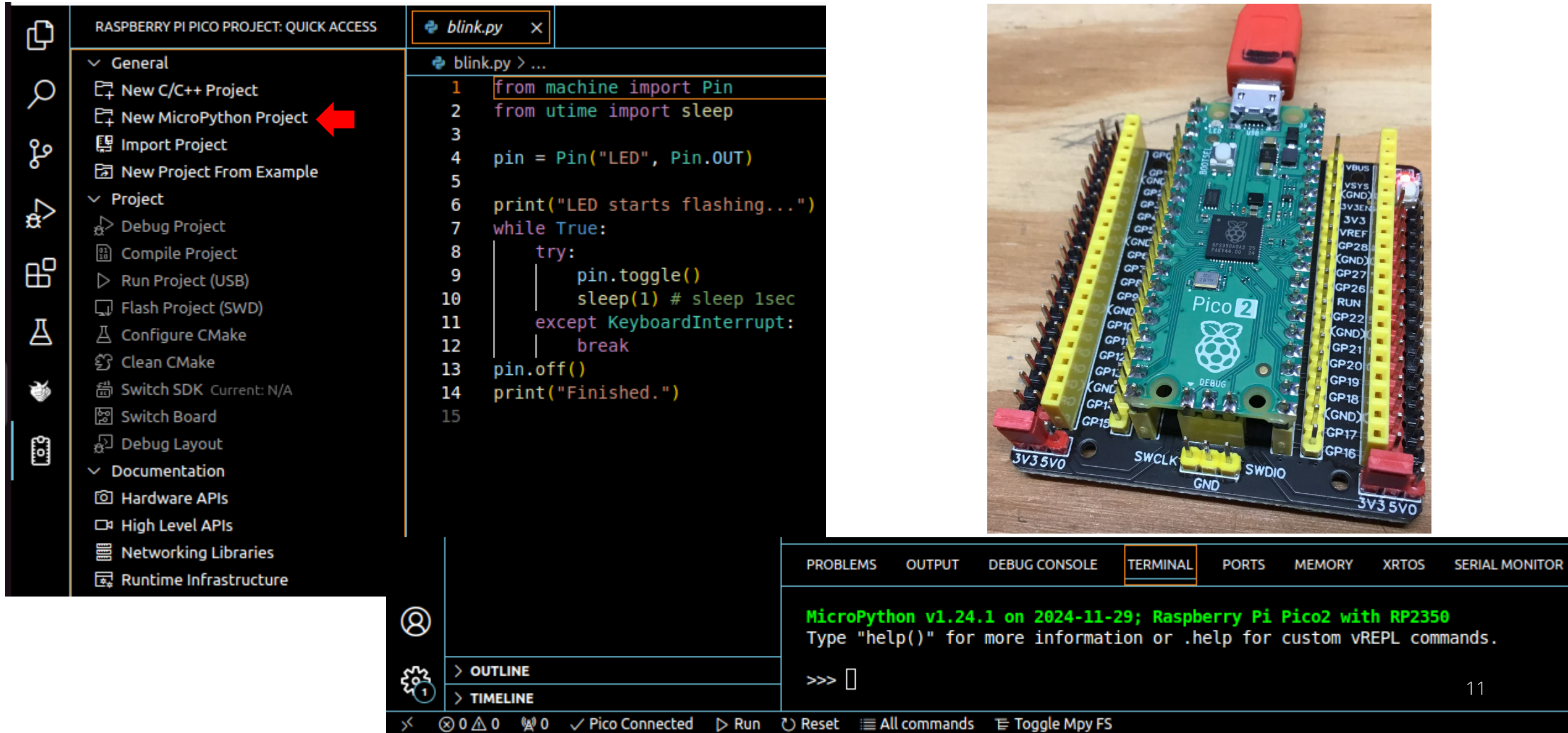
Name: charm_the_snake

Python Version: From Python extension /bin/python3

Location: /home/fred/pico2_python_projects Change

Cancel Create

Feed the Snake – Create a MicroPython Project



RASPBERRY PI PICO PROJECT: QUICK ACCESS

- General
 - New C/C++ Project
 - New MicroPython Project
 - Import Project
 - New Project From Example
- Project
 - Debug Project
 - Compile Project
 - Run Project (USB)
 - Flash Project (SWD)
 - Configure CMake
 - Clean CMake
 - Switch SDK Current: N/A
 - Switch Board
 - Debug Layout
- Documentation
 - Hardware APIs
 - High Level APIs
 - Networking Libraries
 - Runtime Infrastructure

```

1 from machine import Pin
2 from utime import sleep
3
4 pin = Pin("LED", Pin.OUT)
5
6 print("LED starts flashing...")
7 while True:
8     try:
9         pin.toggle()
10        sleep(1) # sleep 1sec
11    except KeyboardInterrupt:
12        break
13 pin.off()
14 print("Finished.")
15

```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS MEMORY XRTOS SERIAL MONITOR

```

MicroPython v1.24.1 on 2024-11-29; Raspberry Pi Pico2 with RP2350
Type "help()" for more information or .help for custom vREPL commands.

>>>

```

< 0 0 0 ✓ Pico Connected ▶ Run ↺ Reset ≡ All commands ⌵ Toggle Mpy FS

Charm the Snake – Run the MicroPython Project

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS MEMORY XRTOS SERIAL MONITOR

```
MicroPython v1.24.1 on 2024-11-29; Raspberry Pi Pico2 with RP2350
Type "help()" for more information or .help for custom vREPL commands.

>>> |
```

< 0 0 0 0 ✓ Pico Connected **▶ Run** ⌂ Reset ≡ All commands ⌘ Toggle Mpy FS

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS MEMORY XRTOS SERIAL MONITOR

```
MicroPython v1.24.1 on 2024-11-29; Raspberry Pi Pico2 with RP2350
Type "help()" for more information or .help for custom vREPL commands.

>>>
LED starts flashing...
|
```

< 0 0 0 0 ✓ Pico Connected **◻ Stop** ⌂ Reset ≡ All commands ⌘ Toggle Mpy FS

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS MEMORY XRTOS SERIAL MONITOR

```
MicroPython v1.24.1 on 2024-11-29; Raspberry Pi Pico2 with RP2350
Type "help()" for more information or .help for custom vREPL commands.

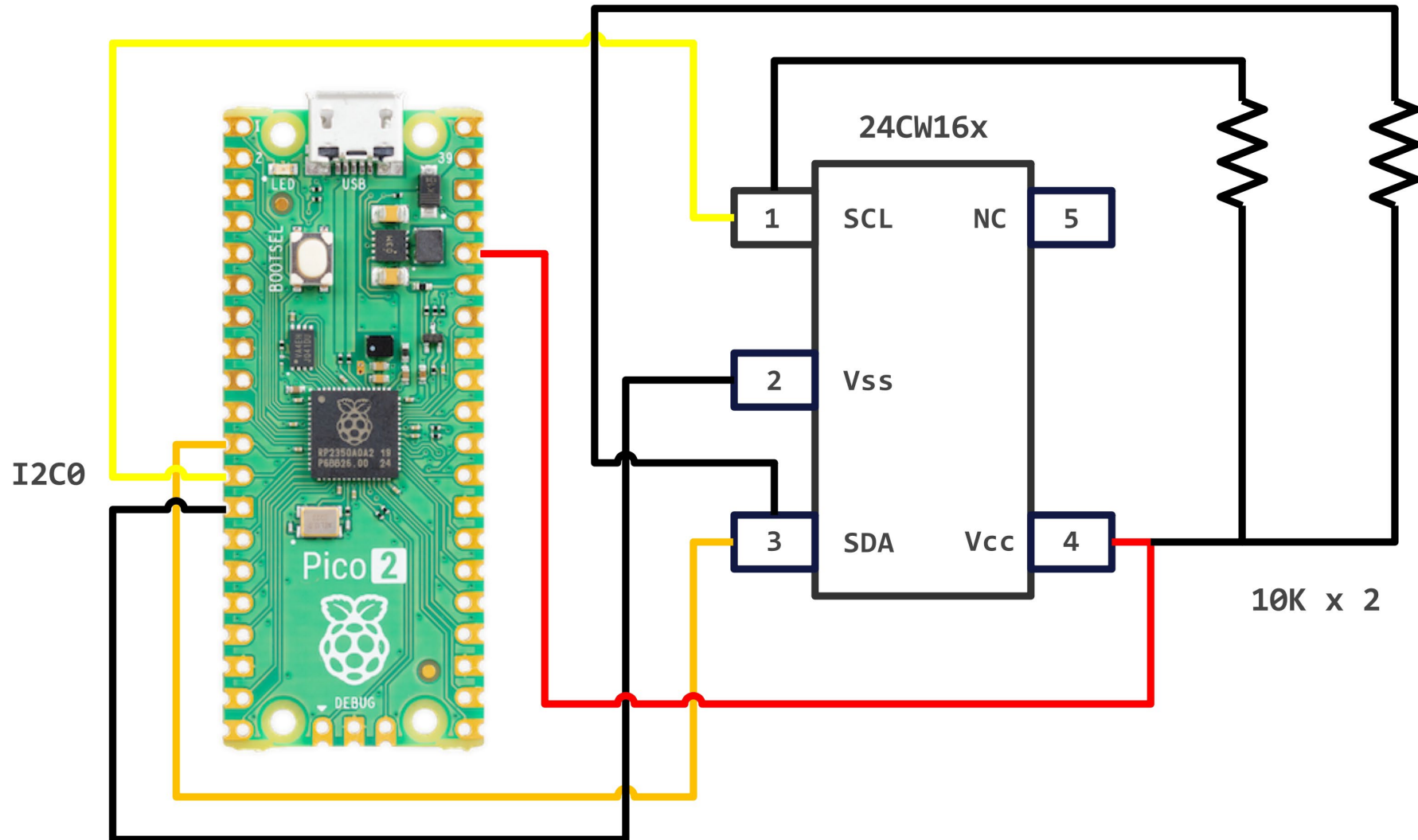
>>>
LED starts flashing...
Finished.

>>> |
```

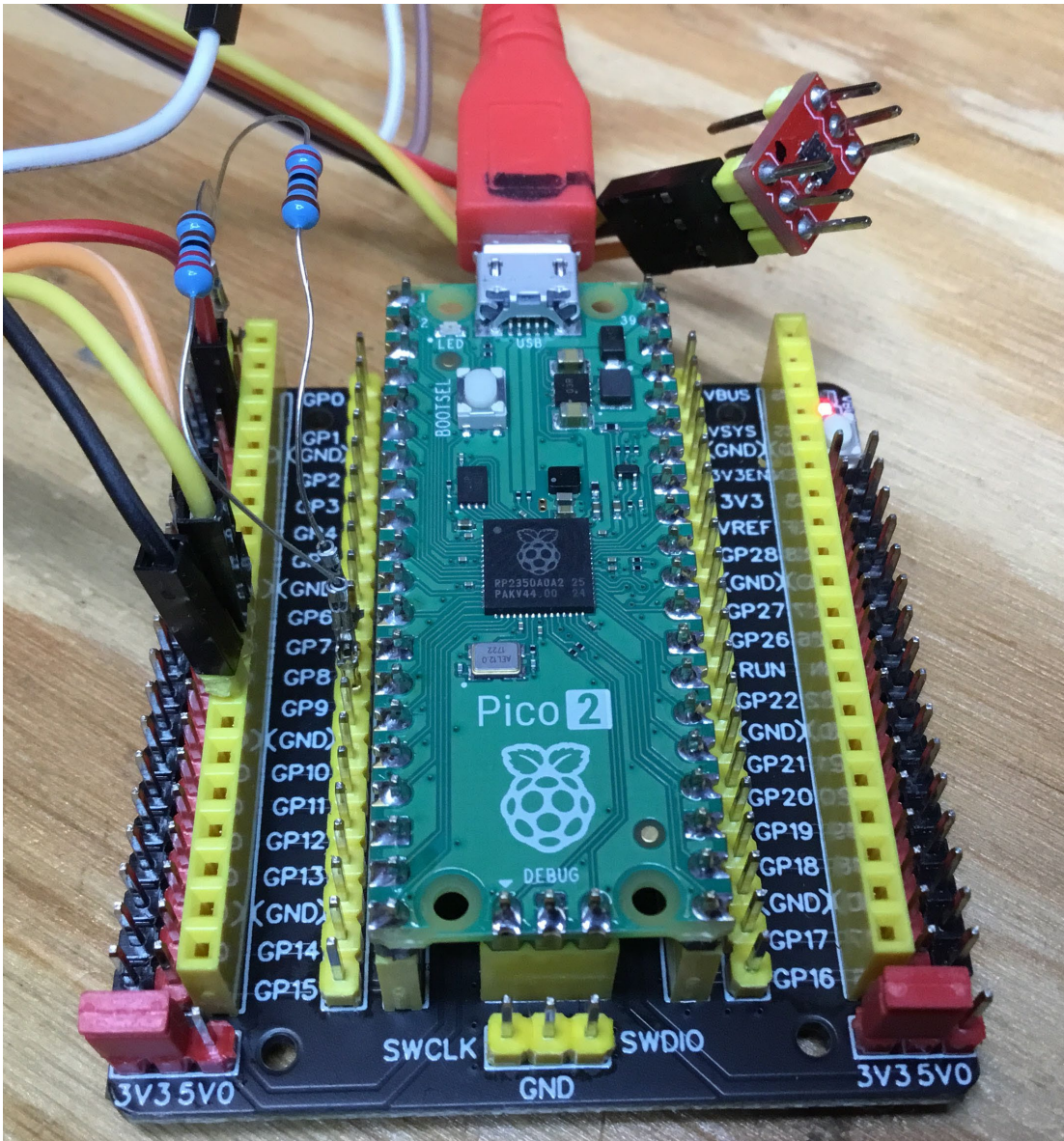
< 0 0 0 0 ✓ Pico Connected **▶ Run** ⌂ Reset ≡ All commands ⌘ Toggle Mpy FS

```
blink.py > ...
1 from machine import Pin
2 from utime import sleep
3
4 pin = Pin("LED", Pin.OUT)
5
6 print("LED starts flashing...")
7 while True:
8     try:
9         pin.toggle()
10        sleep(1) # sleep 1sec
11    except KeyboardInterrupt:
12        break
13 pin.off()
14 print("Finished.")
```

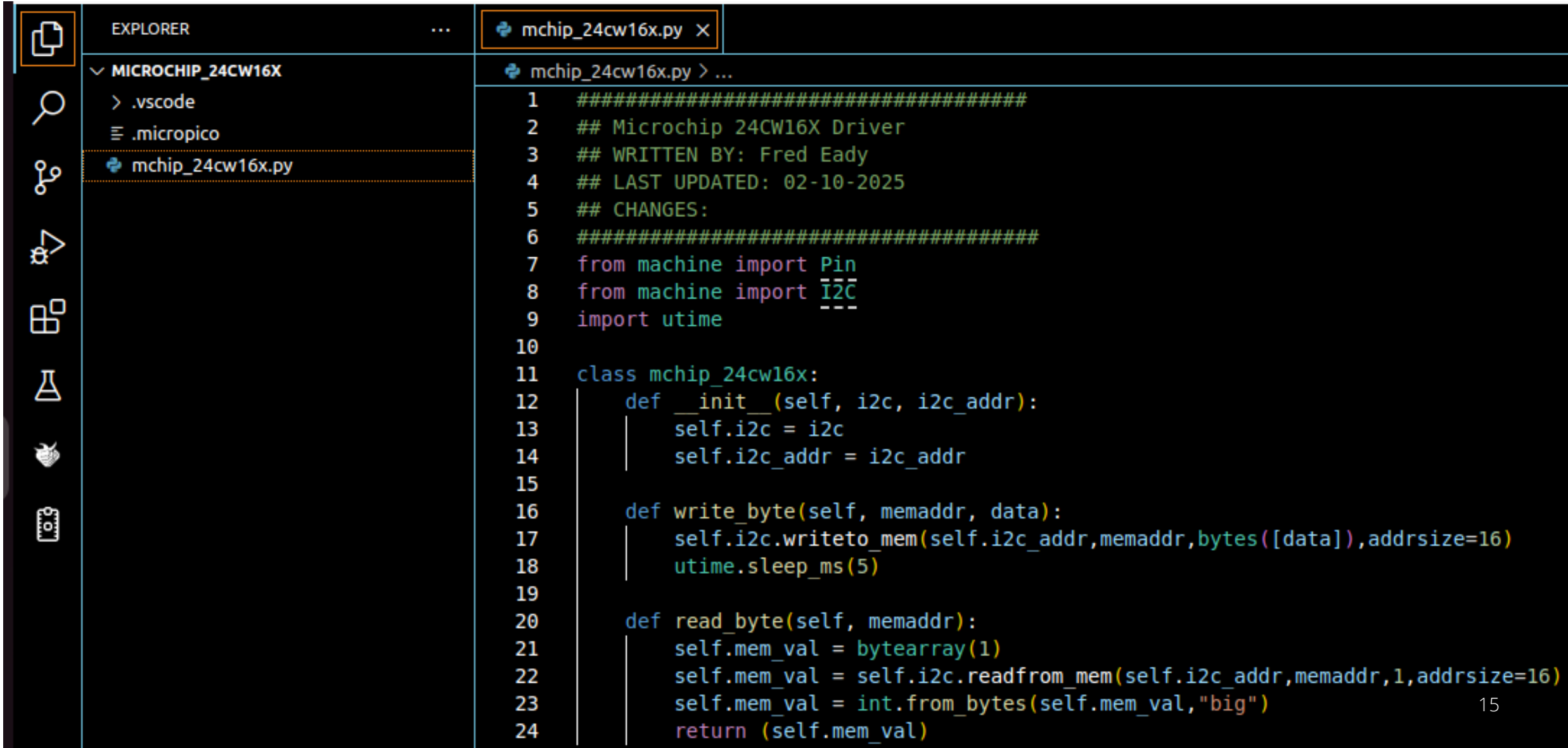
Charm the Snake – 24CW16x EEPROM Driver



Charm the Snake – 24CW16x EEPROM Driver

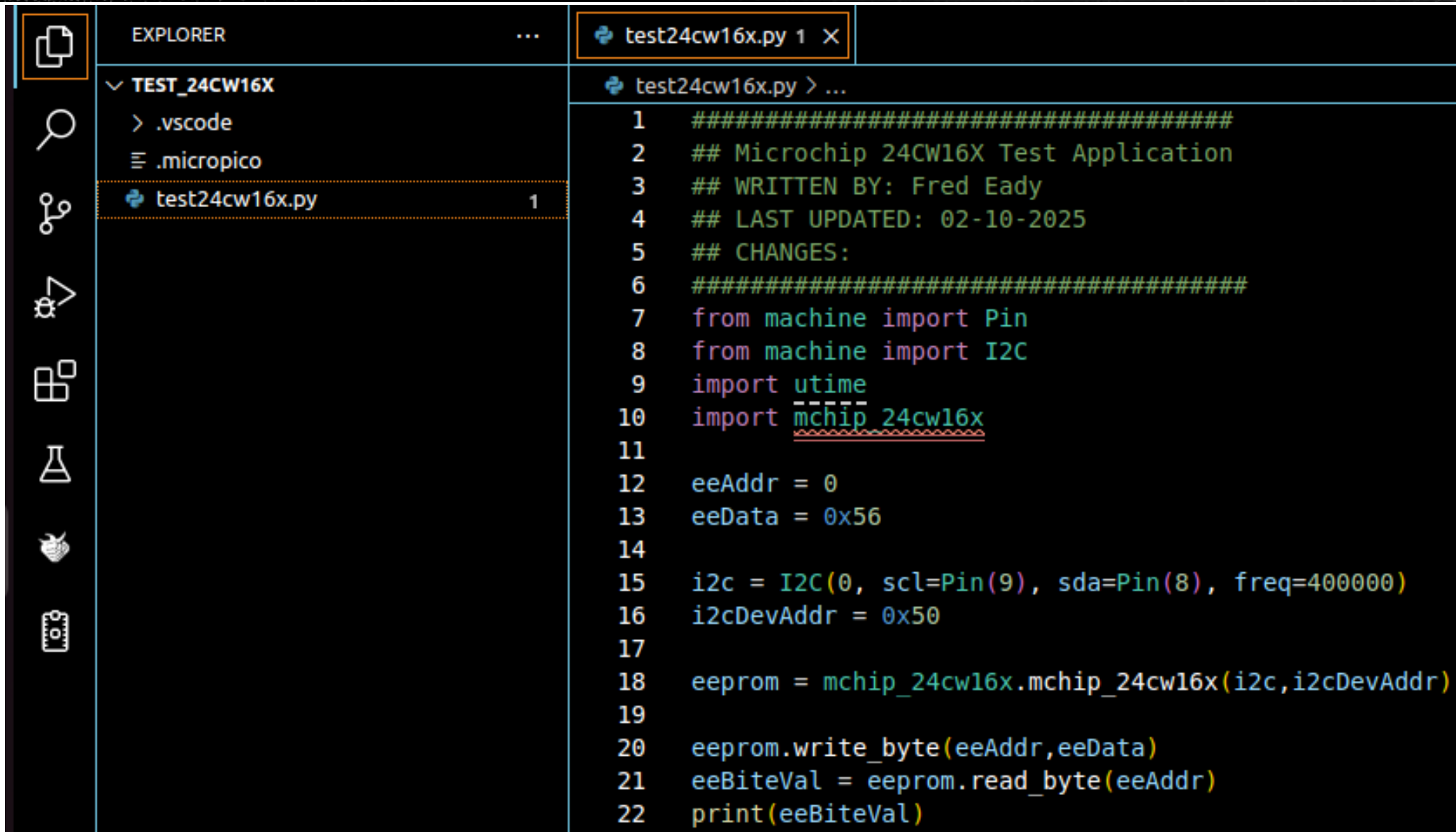


Charm the Snake – 24CW16x EEPROM Driver



```
1 #####
2 ## Microchip 24CW16X Driver
3 ## WRITTEN BY: Fred Eady
4 ## LAST UPDATED: 02-10-2025
5 ## CHANGES:
6 #####
7 from machine import Pin
8 from machine import I2C
9 import utime
10
11 class mchip_24cw16x:
12     def __init__(self, i2c, i2c_addr):
13         self.i2c = i2c
14         self.i2c_addr = i2c_addr
15
16     def write_byte(self, memaddr, data):
17         self.i2c.writeto_mem(self.i2c_addr, memaddr, bytes([data]), addrsize=16)
18         utime.sleep_ms(5)
19
20     def read_byte(self, memaddr):
21         self.mem_val = bytearray(1)
22         self.mem_val = self.i2c.readfrom_mem(self.i2c_addr, memaddr, 1, addrsize=16)
23         self.mem_val = int.from_bytes(self.mem_val, "big")
24         return (self.mem_val)
```

Charm the Snake – 24CW16x EEPROM Driver

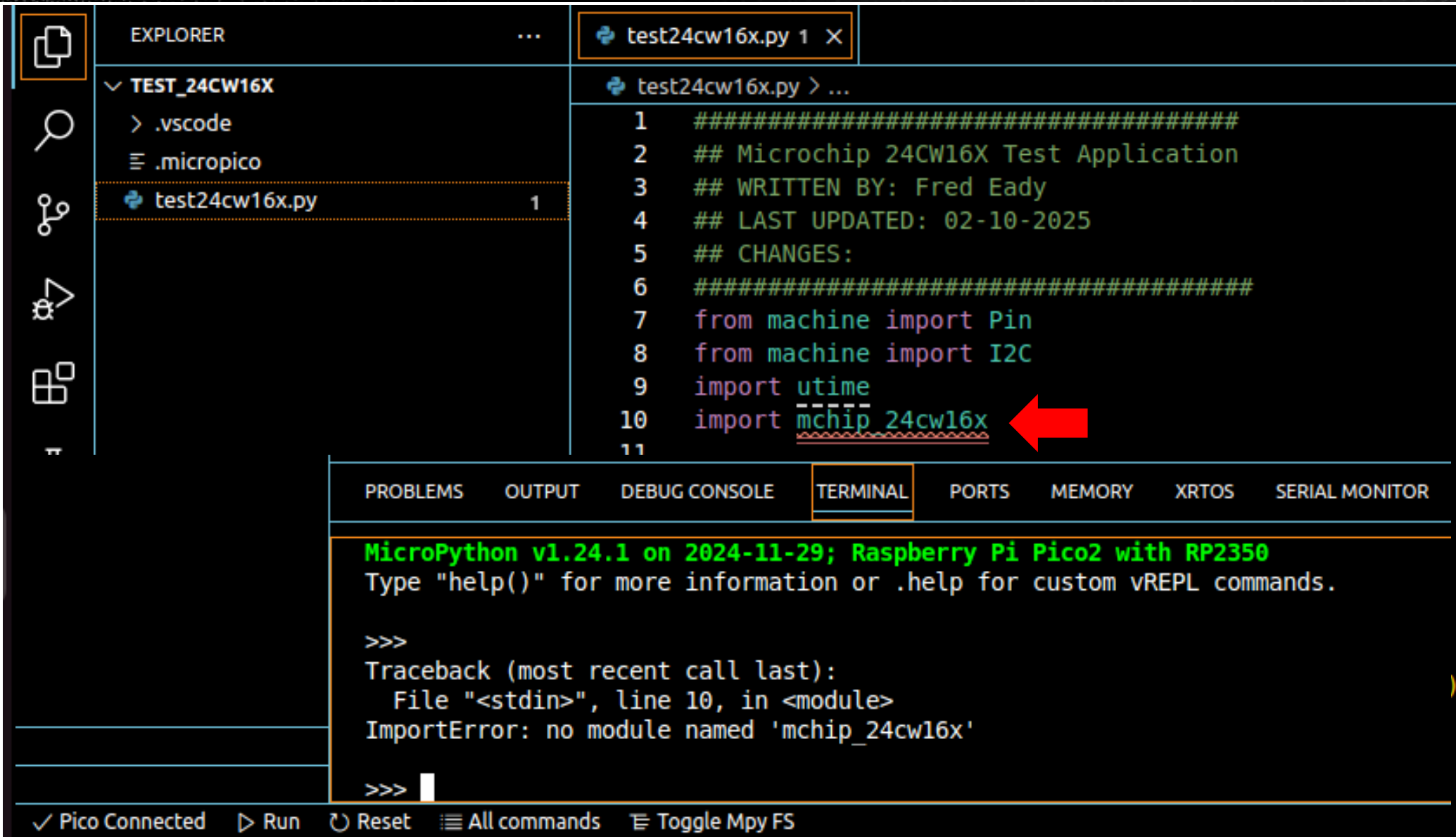


```
EXPLORER
...
test24cw16x.py 1 X

TEST_24CW16X
> .vscode
≡ .micropico
test24cw16x.py 1

test24cw16x.py > ...
1 #####
2 ## Microchip 24CW16X Test Application
3 ## WRITTEN BY: Fred Eady
4 ## LAST UPDATED: 02-10-2025
5 ## CHANGES:
6 #####
7 from machine import Pin
8 from machine import I2C
9 import utime
10 import mchip_24cw16x
11
12 eeAddr = 0
13 eeData = 0x56
14
15 i2c = I2C(0, scl=Pin(9), sda=Pin(8), freq=400000)
16 i2cDevAddr = 0x50
17
18 eeprom = mchip_24cw16x.mchip_24cw16x(i2c,i2cDevAddr)
19
20 eeprom.write_byte(eeAddr,eeData)
21 eeBiteVal = eeprom.read_byte(eeAddr)
22 print(eeBiteVal)
```

Charm the Snake – 24CW16x EEPROM Driver



The screenshot displays the Visual Studio Code interface. The Explorer pane on the left shows a project named 'TEST_24CW16X' with files '.vscode', '.micropico', and 'test24cw16x.py'. The main editor shows the content of 'test24cw16x.py', which includes a header with author and date information, and Python code imports. A red arrow points to the import statement for 'mchip_24cw16x' on line 10. The bottom panel shows the 'TERMINAL' output, which displays the MicroPython version and environment, followed by a traceback error: 'ImportError: no module named 'mchip_24cw16x''. The status bar at the bottom indicates 'Pico Connected' and provides buttons for 'Run', 'Reset', 'All commands', and 'Toggle Mpy FS'.

```
EXPLORER
...
test24cw16x.py 1 X

TEST_24CW16X
  > .vscode
  ≡ .micropico
  test24cw16x.py 1

test24cw16x.py > ...
1 #####
2 ## Microchip 24CW16X Test Application
3 ## WRITTEN BY: Fred Eady
4 ## LAST UPDATED: 02-10-2025
5 ## CHANGES:
6 #####
7 from machine import Pin
8 from machine import I2C
9 import utime
10 import mchip_24cw16x
11

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS MEMORY XRTOS SERIAL MONITOR

MicroPython v1.24.1 on 2024-11-29; Raspberry Pi Pico2 with RP2350
Type "help()" for more information or .help for custom vREPL commands.

>>>
Traceback (most recent call last):
  File "<stdin>", line 10, in <module>
ImportError: no module named 'mchip_24cw16x'

>>>
```

✓ Pico Connected ▶ Run ↺ Reset ≡ All commands ≡ Toggle Mpy FS

Charm the Snake – 24CW16x EEPROM Driver

EXPLORER

test24cw16x.py

UNTITLED (WORKSPACE)

- test_24cw16x
 - .vscode
 - .micropico
 - mchip_24cw16x.py
 - test24cw16x.py
- Mpy Remote Workspace

```
test_24cw16x > test24cw16x.py > ...
1 #####
2 ## Microchip 24CW16X Test Application
3 ## WRITTEN BY: Fred Eady
4 ## LAST UPDATED: 02-10-2025
5 ## CHANGES:
6 #####
7 from machine import Pin
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS MEMORY XRTOS SERIAL MONITOR

```
MicroPython v1.24.1 on 2024-11-29; Raspberry Pi Pico2 with RP2350
Type "help()" for more information or .help for custom vREPL commands.

>>>
Traceback (most recent call last):
  File "<stdin>", line 10, in <module>
ImportError: no module named 'mchip_24cw16x'

>>>
```

✓ Pico Connected Run Reset All commands **Toggle Mpy FS**

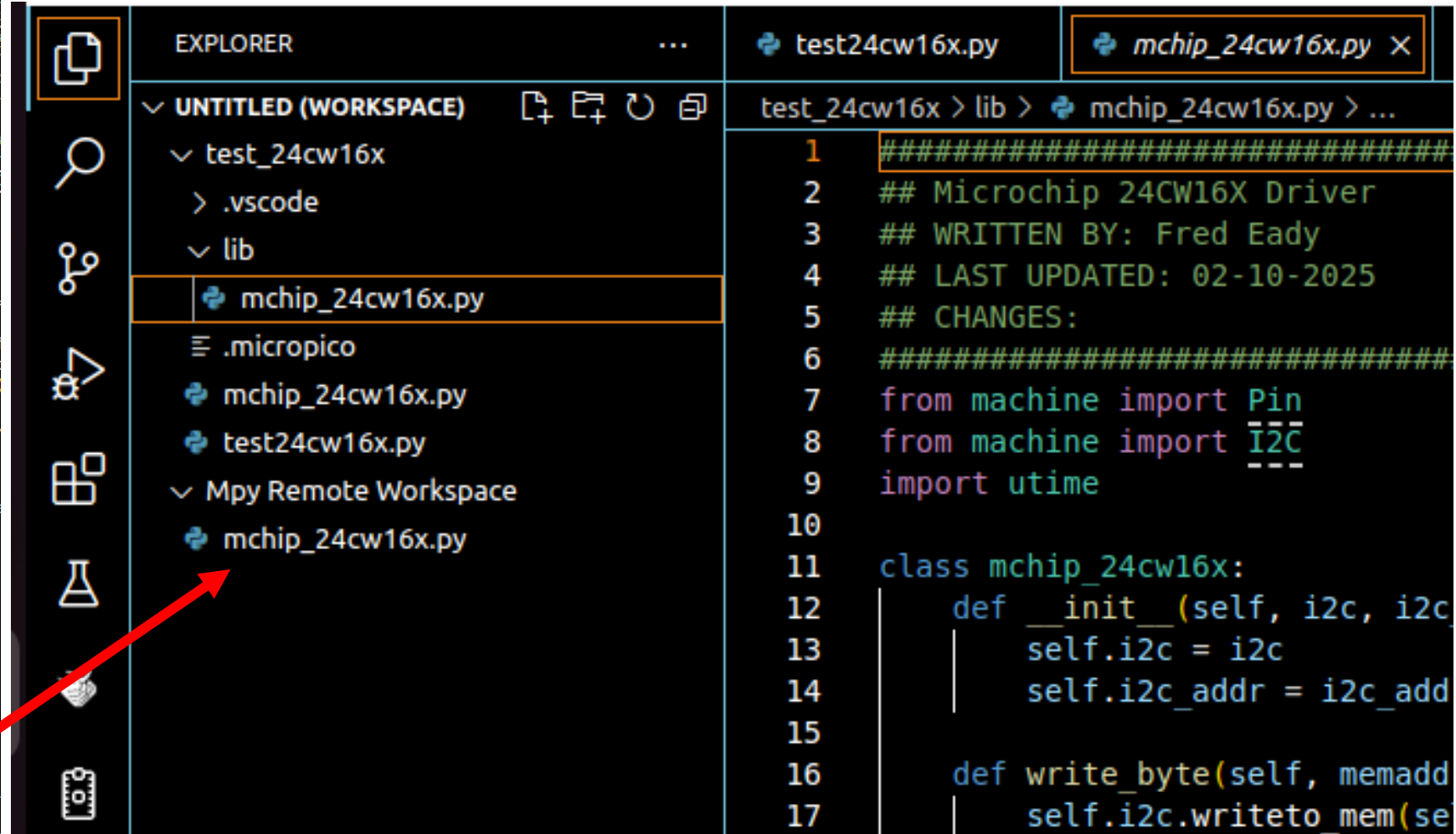
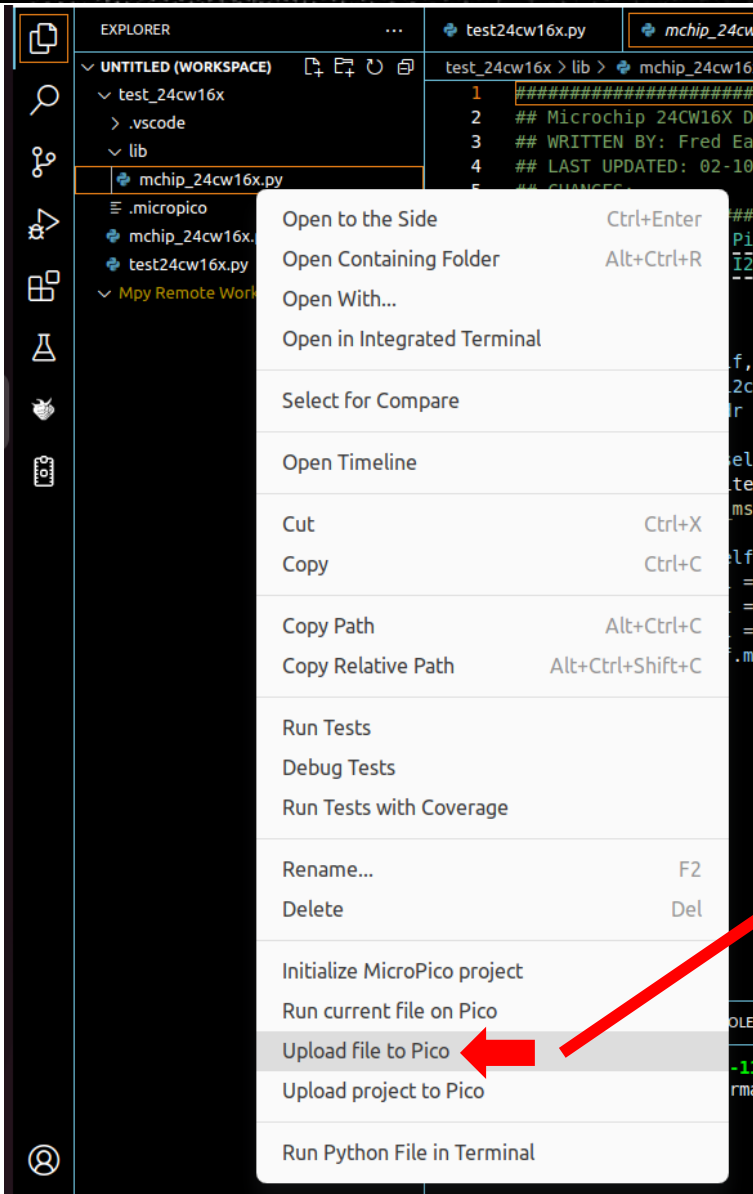
Charm the Snake – 24CW16x EEPROM Driver

The image shows a Visual Studio Code interface with two windows open. The Explorer window on the left shows a project structure under 'UNTITLED (WORKSPACE)'. It contains a folder 'test_24cw16x' with subfolders '.vscode' and 'lib'. The 'lib' folder is expanded, showing a file 'mchip_24cw16x.py'. A red arrow points from this file to the code editor window. The code editor window shows the content of 'mchip_24cw16x.py' with the following text:

```
1 #####  
2 ## Microchip 24CW16X Driver  
3 ## WRITTEN BY: Fred Eady  
4 ## LAST UPDATED: 02-10-2025  
5 ## CHANGES:  
6 #####
```

A file browser window is also open, showing the path 'Home / pico2_python_projects / microchip_24cw16x'. The file 'mchip_24cw16x.py' is selected and highlighted in orange.

Charm the Snake – 24CW16x EEPROM Driver



Charm the Snake – 24CW16x EEPROM Driver

The screenshot displays the Visual Studio Code interface for a Raspberry Pi Pico project. The Explorer view on the left shows a workspace named 'TEST_24CW16X (WORKSPACE)' containing a folder 'test_24cw16x' with files like '.vscode', 'lib', '.micropico', 'mchip_24cw16x.py', and 'test24cw16x.py'. The main editor shows the code for 'test24cw16x.py'.

```
1 #####
2 ## Microchip 24CW16X Test Application
3 ## WRITTEN BY: Fred Eady
4 ## LAST UPDATED: 02-10-2025
5 ## CHANGES:
6 #####
7 from machine import Pin
8 from machine import I2C
9 import utime
10 import mchip_24cw16x
11
12 eeAddr = 0
13 eeData = 0x56
14
15 i2c = I2C(0, scl=Pin(9), sda=Pin(8), freq=400000)
16 i2cDevAddr = 0x50
17
18 eeprom = mchip_24cw16x.mchip_24cw16x(i2c,i2cDevAddr)
19
20 eeprom.write_byte(eeAddr,eeData)
21 eeBiteVal = eeprom.read_byte(eeAddr)
22 print(eeBiteVal)
23
24
```

The Terminal view at the bottom shows the MicroPython prompt:

```
MicroPython v1.24.1 on 2024-11-29; Raspberry Pi Pico2 with RP2350
Type "help()" for more information or .help for custom vREPL commands.

>>>
86

>>> |
```

Charm the Snake – 24CW16x EEPROM Driver

The screenshot displays the Visual Studio Code interface. The Explorer sidebar on the left shows a project structure for 'TEST_24CW16X' with subfolders '.vscode', 'lib', and '.micropico'. The file 'test24cw16x.py' is selected. The main editor window shows the code for 'test24cw16x.py' with the following content:

```
1 #####  
2 ## Microchip 24CW16X Test Application  
3 ## WRITTEN BY: Fred Eady  
4 ## LAST UPDATED: 02-10-2025  
5 ## CHANGES:  
6 #####
```

A file explorer overlay is visible in the foreground, showing the file system structure for the project. The path is 'Home / pico2_python_projects / test_24cw16x'. The file explorer lists the following files and folders:

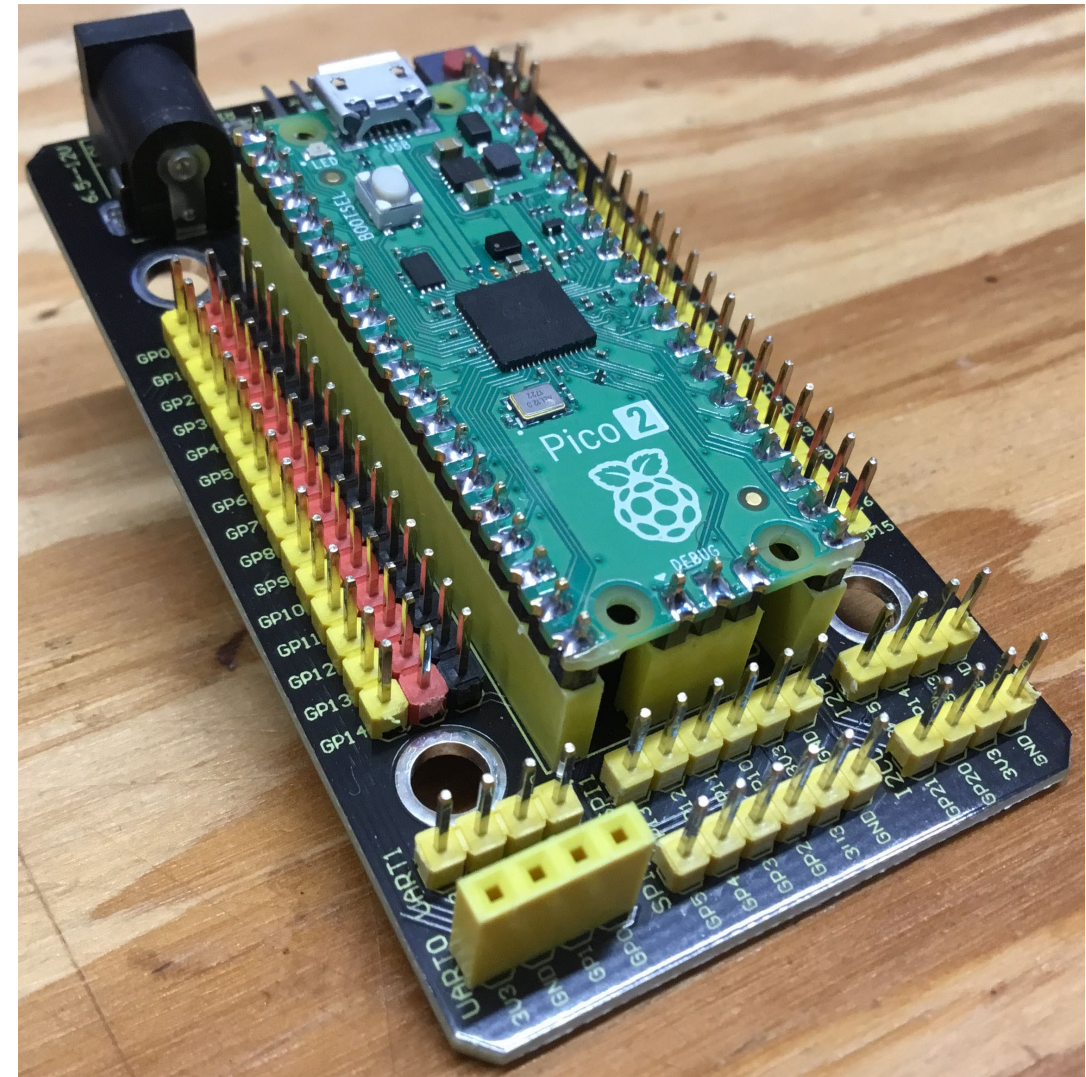
- lib
- mchip_24cw16x.py
- test24cw16x.py
- test_24cw16x.code-workspace (highlighted)

Next Time...**MORE TO COME..**

Thank you for attending!!!

Please consider the resources below:

- [Today's Download Package](#)
- [Microchip 24CW16x Data Sheet](#)
- micropython.org





DesignNews

Thank You

Sponsored by

DigiKey

