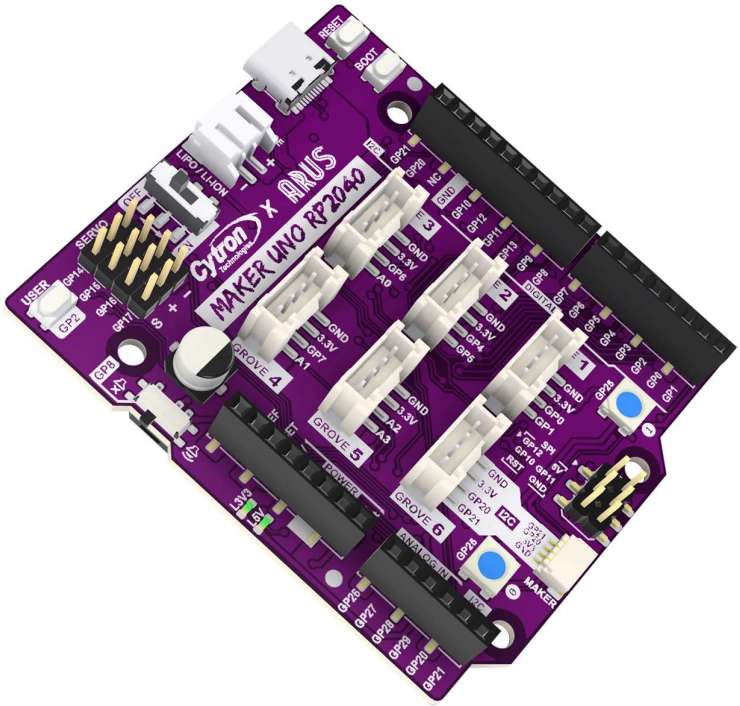
MAKER UNO RP2040



Datasheet

Rev 1.0
February 2024

** Raspberry Pi is a trademark of the Raspberry Pi Foundation.*

1. BOARD LAYOUT AND FUNCTION

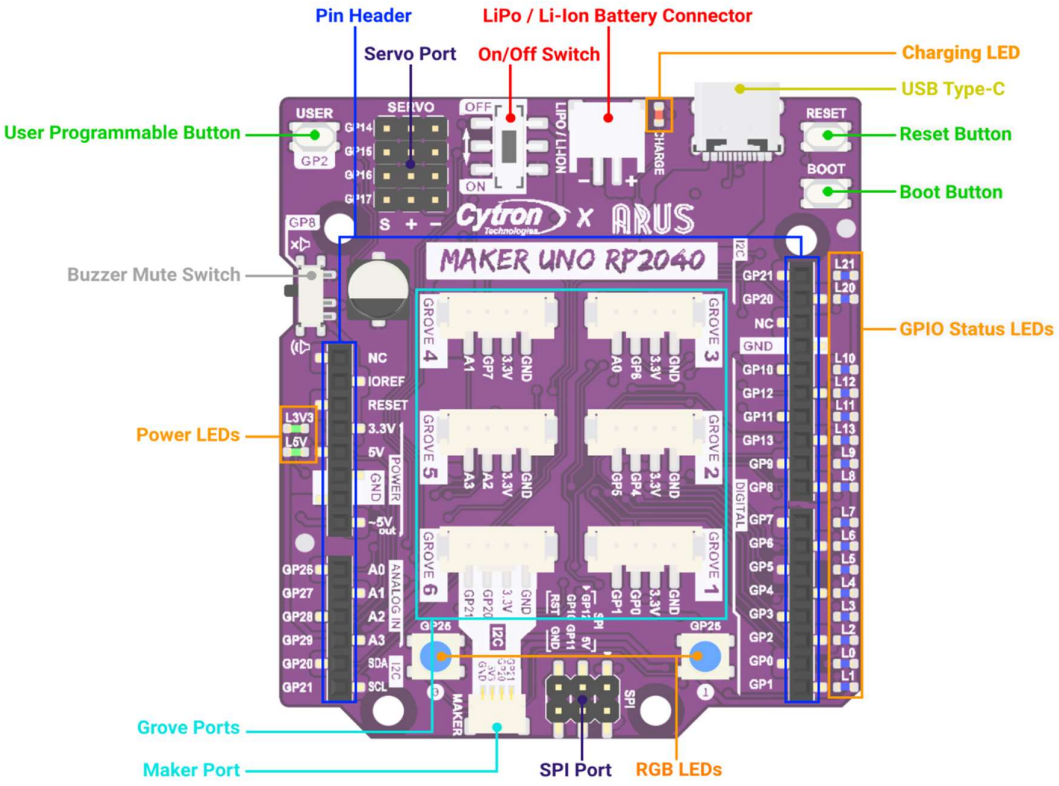


Figure 1: MAKER-UNO-RP2040 Board Functions (TOP)

Function	Description
LiPo / Li-Ion Battery Connector	● JST-PH 2-pin connector for LiPo / Li-Ion battery. ● Connect to single cell LiPo / Li-Ion battery. The battery is rechargeable via USB. ∗ *The battery is protected from overcharged and over discharged. If the board cannot be turned on when the battery is connected, please charge the battery to activate the battery protection circuit.*
On/Off Switch	● Slide switch used to turn on or off the power.
Charging LED	● LED Indicator for the LiPo/Li-Ion battery charging status. ● Turns on when the battery is charging and off when the battery is full
USB Type-C	● Used for both powering and programming the board from a PC.
Reset Button	● Button used to reset the RP2040. ● Press to restart the board without unplugging the USB cable or battery.
Boot Button	● Button used to enter bootloader mode. ● Press and hold this button while resetting the Maker Uno RP2040 to enter bootloader mode. This mode is used to load either the Micropython/Circuitpython firmware or custom C/C++ firmware onto

	the device."

<table>
<tr><th colspan="2">Function</th><th>Description</th></tr>
<tr><td></td><td>User Programmable Button</td><td>● Accessible from the user program.
● Internally connected to GP2.</td></tr>
<tr><td></td><td>Grove Ports</td><td>● 4-pin, 2.0mm pitch, Grove connector for external grove modules.
● Pins connection;
<table>
<tr><th>Grove Port</th><th>GPIO</th><th>PWM</th><th>SPI</th><th>I2C</th><th>UART</th><th>Analog</th></tr>
<tr><td rowspan="2">1</td><td>0</td><td>PWM0-A</td><td>SDI0</td><td>SDA0</td><td>TX0</td><td>-</td></tr>
<tr><td>1</td><td>PWM0-B</td><td>CSn0</td><td>SCL0</td><td>RX0</td><td>-</td></tr>
<tr><td rowspan="2">2</td><td>4</td><td>PWM2-A</td><td>SDI0</td><td>SDA0</td><td>TX1</td><td>-</td></tr>
<tr><td>5</td><td>PWM2-B</td><td>CSn0</td><td>SCL0</td><td>RX1</td><td>-</td></tr>
<tr><td rowspan="2">3</td><td>6</td><td>PWM3-A</td><td>SCK0</td><td>SDA1</td><td>-</td><td>-</td></tr>
<tr><td>26</td><td>PWM5-A</td><td>SCK1</td><td>SDA1</td><td>-</td><td>ADC0</td></tr>
<tr><td rowspan="2">4</td><td>7</td><td>PWM3-B</td><td>SDO0</td><td>SCL1</td><td>-</td><td>-</td></tr>
<tr><td>27</td><td>PWM5-B</td><td>SDO1</td><td>SCL1</td><td>-</td><td>ADC1</td></tr>
<tr><td rowspan="2">5</td><td>28</td><td>PWM6-A</td><td>SDI1</td><td>SDA0</td><td>TX0</td><td>ADC2</td></tr>
<tr><td>29</td><td>PWM6-B</td><td>CSn1</td><td>SCL0</td><td>RX0</td><td>ADC3</td></tr>
<tr><td rowspan="2">6</td><td>20</td><td>PWM2-A</td><td>SDI0</td><td>SDA0</td><td>TX1</td><td></td></tr>
<tr><td>21</td><td>PWM2-B</td><td>CSn0</td><td>SCL0</td><td>RX1</td><td></td></tr>
</table></td></tr>
<tr><td></td><td>Maker Port</td><td>● JST-SH 4-pin Connector for external modules or sensors.
● Compatible with QWIIC / Stemma QT / Grove (with conversion cable).
* The pins are shared with the Grove 6 port (GPIO 20, 21).</td></tr>
<tr><td></td><td>Pin Header</td><td>● Female connector for external connection.</td></tr>
<tr><td></td><td>Servo Ports</td><td>● Connectors for 4 x RC servo motors.
● Signal (S) is internally connected to GP14, GP15, GP16 and GP17 respectively.
● Servo Voltage is equal to the power source voltage.</td></tr>
<tr><td></td><td>SPI Port</td><td>● Male header for SPI communication.</td></tr>
<tr><td></td><td>RGB LEDs (WS2812)</td><td>● User programmable WS2812B RGB LED.
● Internally connected to GP25.</td></tr>
<tr><td></td><td>GPIO Status LEDs</td><td>● LED indicators for RP2040 GPIOs that are connected to the socket header.
● Turns on when the GPIO state is high.</td></tr>
<tr><td></td><td>Power LEDs</td><td>● LED indicator for 3.3V and 5V.
● Turn on when powered up.</td></tr>
<tr><td></td><td>Buzzer Mute Switch</td><td>● Slide switch used to mute the piezo buzzer (placed at the bottom layer of the PCB).
● Suggested to use if GP8 is used for other purposes.</td></tr>
</table>

Table 1: MAKER-UNO-RP2040 Board Functions

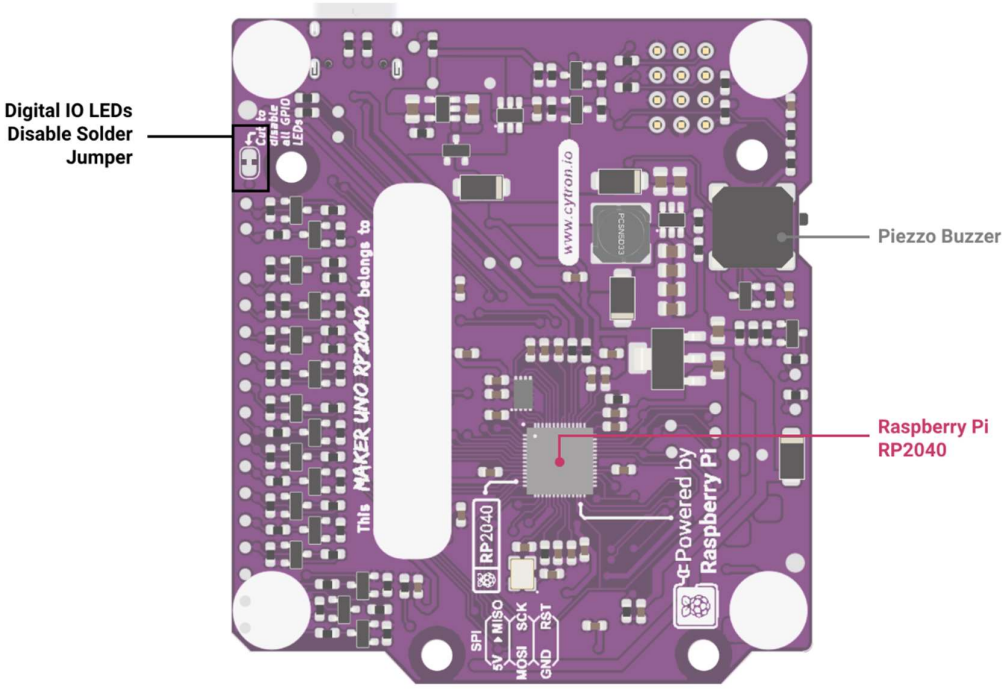


Figure 2: MAKER-UNO-RP2040 Board Functions (Bottom)

Function	Description
Raspberry Pi RP2040	● Microcontroller for the Maker Uno RP2040 board. ● Low-cost, high-performance microcontroller equipped with a dual-core ARM Cortex-M0+ processor, ample memory, and versatile I/O options.
Piezo Buzzer	● Programmable piezo buzzer, can be used to play tone or melody. ● Internally connected to GP8.
Digital IO LEDs Disable Solder Jumper	● Default closed solder jumper that connects 3.3V with IOs LED power trace. ● Cut (open) the connecting trace to disable the digital IO LEDs. ● This action can help to conserve power or reduce distraction from the IO LEDs.

Table 2: MAKER-UNO-RP2040 Board Functions (Bottom)

2. PWM ON RP2040

The RP2040 PWM block is composed of 8 slices (PWM0, PWM1, PWM2...), each having the capability to drive two PWM output signals, A and B resulting in a total of up to 16 controllable PWM outputs. Each GPIO pin is assigned with a PWM slice and output, which is also referred to as a channel. Therefore, all RP2040 GPIO pins support PWM. However, if the GPIO pins share the same PWM channel, simultaneous PWM usage is not possible.

Note that for the Maker Uno RP2040 not all GPIO pins of the RP2040 are broken out and connected, the table below highlights potential channel conflicts that may occur. Please refer to the pinout diagram in the next section for detailed information about each pin's PWM channel.

PWM Channel	GPIO
PWM0-A	0
	16 (Servo 3)
PWM0-B	1
	17 (Servo 4)
PWM2-A	4
	20
PWM2-A	5
	21

Table 3: Shared PWM channel for connected GPIO pins.

3. SPECIFICATION

No	Parameters		Min	Max	Unit
1	Power Input Voltage (USB or LiPo/Li-Ion)		3.6	6	V
2	Digital Input Voltage	Low Level	-0.3	0.8	V
		High Level	2.0	3.6	V
3	Digital Output Voltage	Low Level	0	0.5	V
		High Level	2.62	3.3	V
4	Analog Input Voltage		0	3.3	V
5	Total +3V3 Output Current		-	500	mA
6	Total +5V Output Current		-	600	mA
7	Vservo (Only USB is connected)		VUSB - 0.4		V
8	Vservo (Only LiPo/Li-Ion is connected)		*VBAT		V
9	Vservo (USB and LiPo/Li-Ion are connected)		VUSB - 0.4		V
10	Operating Temperature		-20	85	℃
11	USB VID & PID (CircuitPython & Arduino Core)	VID	0x2E8A		
		PID	0x1071		c

Table 4: MAKER-UNO-RP2040 Absolute Maximum Ratings

*VBAT is the voltage of the LiPo/Li-Ion battery.

4. DIMENSION

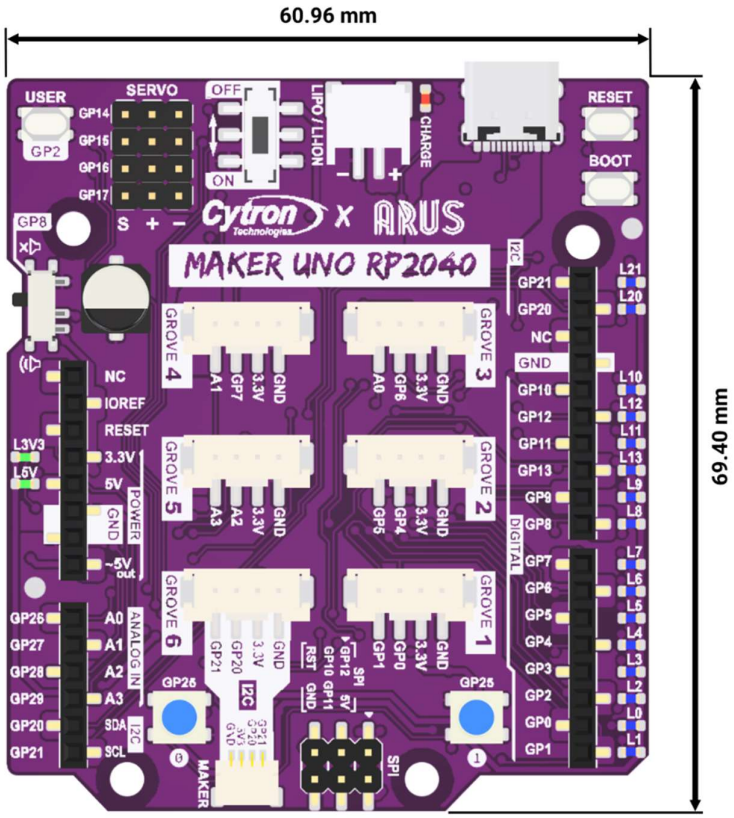


Figure 4: MAKER-UNO-RP2040 Dimension

5. SETTING UP FOR ARDUINO IDE, CIRCUITPYTHON OR MICROPYTHON

In this guide, we will walk you through setting up your Maker Uno RP2040 for three different programming platform; CircuitPython, Arduino IDE and MicroPython. We'll then test each setup with a simple blink code to ensure everything is working correctly.

Prerequisites:

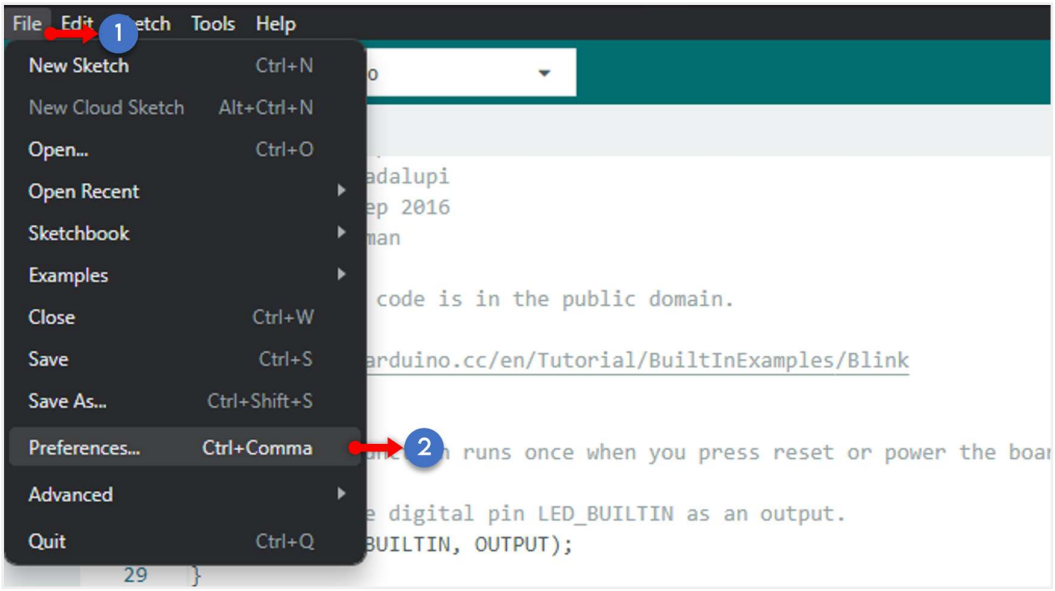
- Maker Uno RP2040
- USB-C cable
- Computer with a USB port
- Corresponding IDEs/coding softwares for each coding platform.

ARDUINO IDE

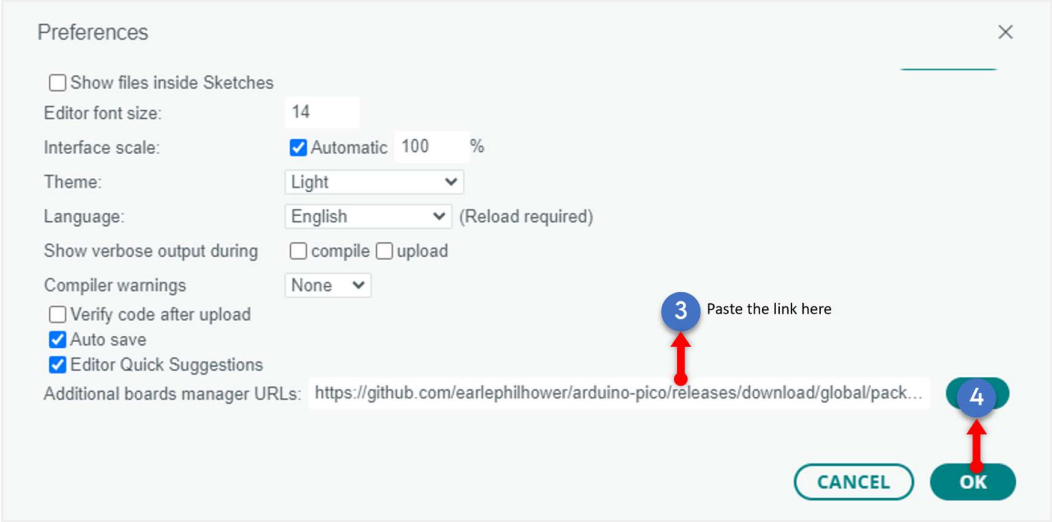
For this guide, it is written based on Arduino IDE version 2.

Before starting programming the board you will need to add the Maker Uno RP2040 to the Arduino Boards Manager.

1. Open the Arduino IDE and navigate to the following directory: File > Preferences > Settings.



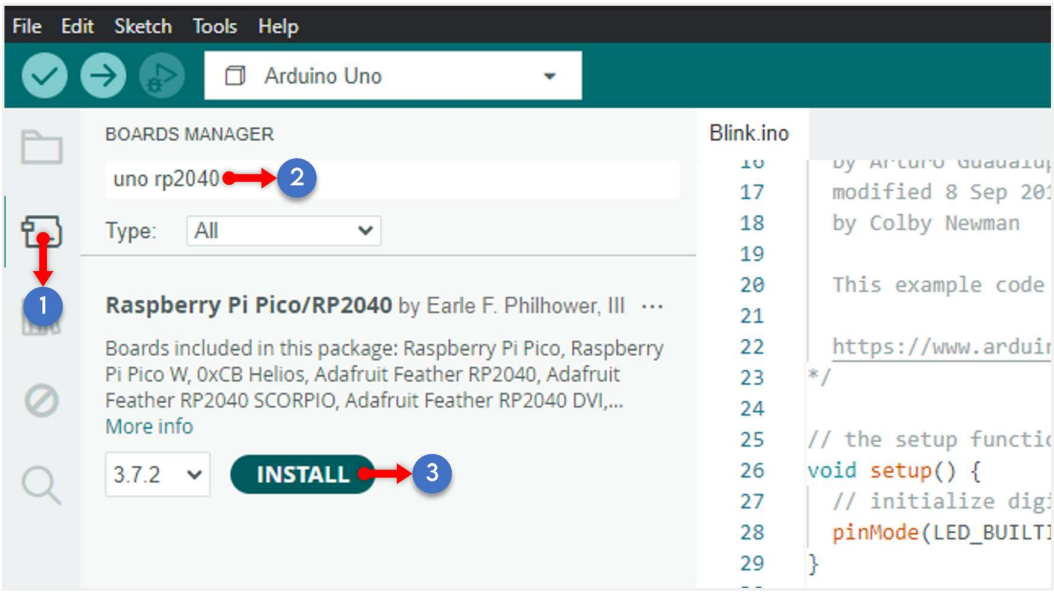
2. Scroll to the bottom of the window pane and locate the "Additional Boards Manager URLs" box and paste this following URL: https://github.com/earlephilhower/arduino-pico/releases/download/global/package_rp2040_index.json. Click “OK” to close the window.



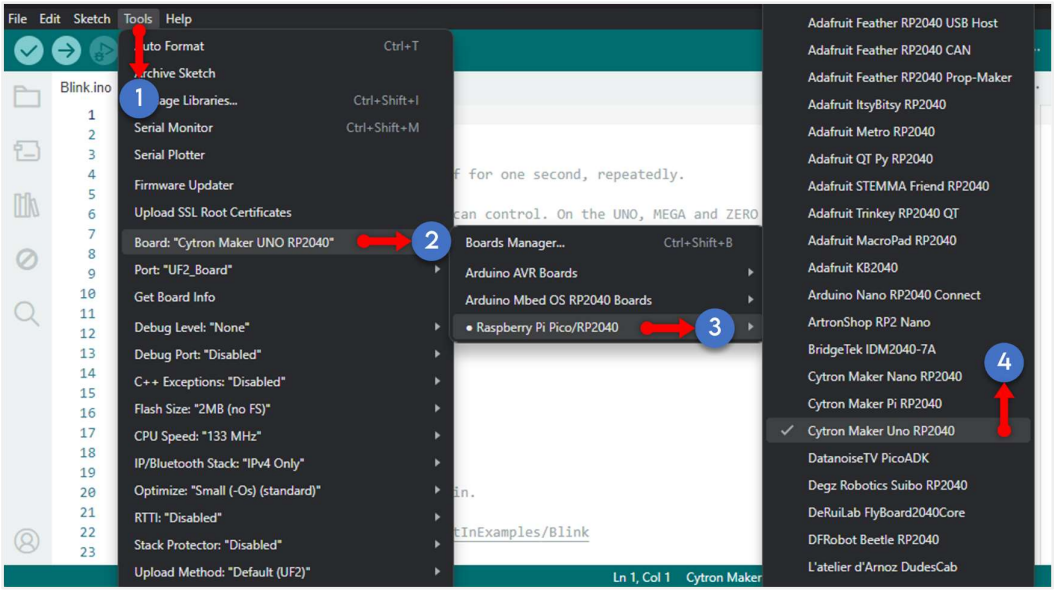
Note: If you already have URLs for other boards, you can separate them with commas like this:

```
https://github.com/earlephilhower/arduino-pico/releases/download/global/package_rp2040_index.json,
https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json
```

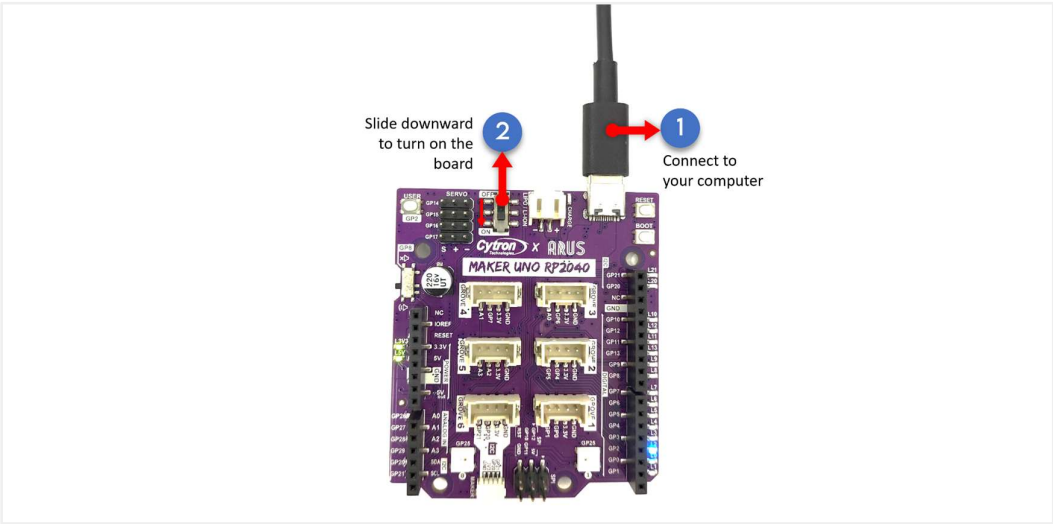
3. Open the Boards Manager by clicking the board manager icon or navigating to Tools > Board > Boards Manager. Search for the word "uno rp2040," and the **Raspberry Pi Pico/RP2040 by Earle F. Philhower** board package should appear, as shown in the picture below. Click "Install" to install the board package.



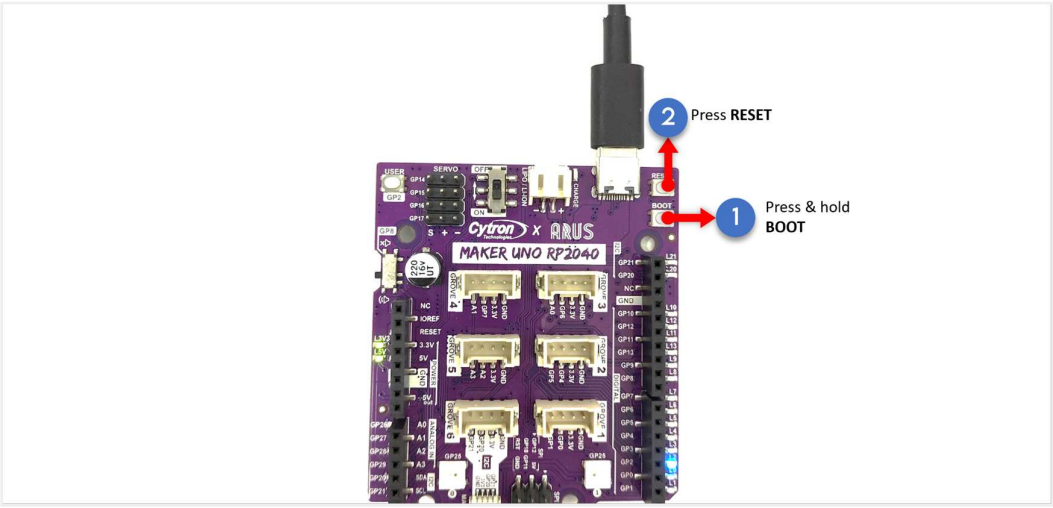
4. After installation completed. Select Maker Uno RP2040 board from Tools > Board>Raspberry Pi Pico/RP2040> Cytron Maker Uno RP2040.



5. Next, to upload code to the Maker Uno RP2040 for the first time, you need to ensure it is in bootloader mode. For that, firstly connect the board to your computer using a USB-C cable and turn on the board.

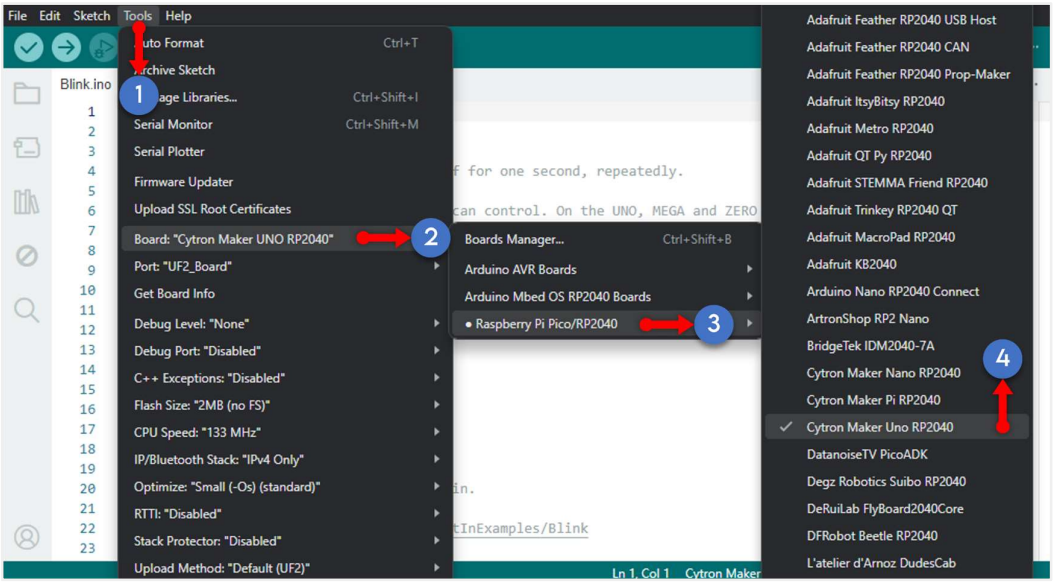


6. Press and hold the “BOOT” button and then press the “RESET” button. Make sure you release the “RESET” button first before releasing the “BOOT” button.



7. Select the corresponding COM port by selecting Tools>Port. Now we are all set to program and upload the code to the Maker Uno RP2040 board!

   Usually, the COM port will initially appear as "UF2_Board". After the upload is successful, the board will reset, and the COM port will then reappear as its corresponding port.



Note: You might need to recheck and reselect the COM port for the next sketch/code upload.

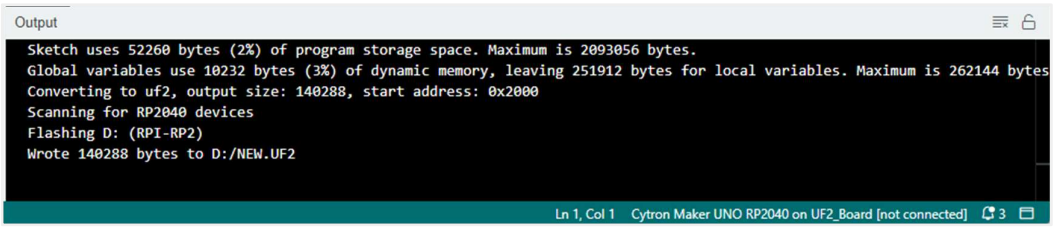
8. Now let's try uploading a simple sketch. Copy the code below into the Arduino IDE. Verify and upload the sketch by clicking by clicking the "Upload" icon in the Arduino IDE

```
//Turn an LED on for one second, then off for one second, repeatedly.
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(1, OUTPUT);
}

// the loop function runs over and over again forever
```

```
void loop() {
  digitalWrite(1, HIGH);  // turn the LED on (HIGH is the voltage level)
  delay(1000);                // wait for a second
  digitalWrite(1, LOW);   // turn the LED off by making the voltage LOW
  delay(1000);                // wait for a second
}
```

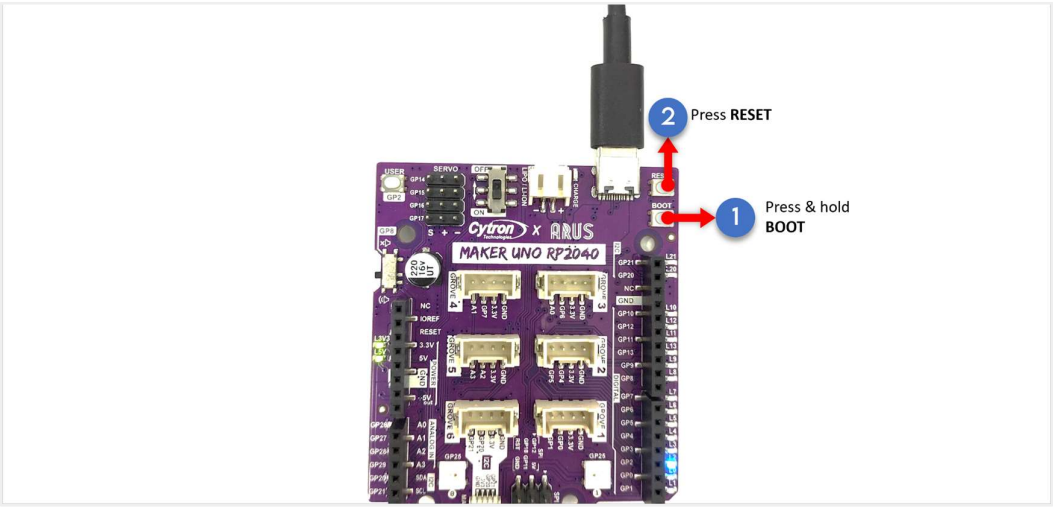
9. If the code is successfully uploaded to the board, an output message as shown in the figure below should appear, and the onboard LED1 should be blinking every one second.



CIRCUITPYTHON

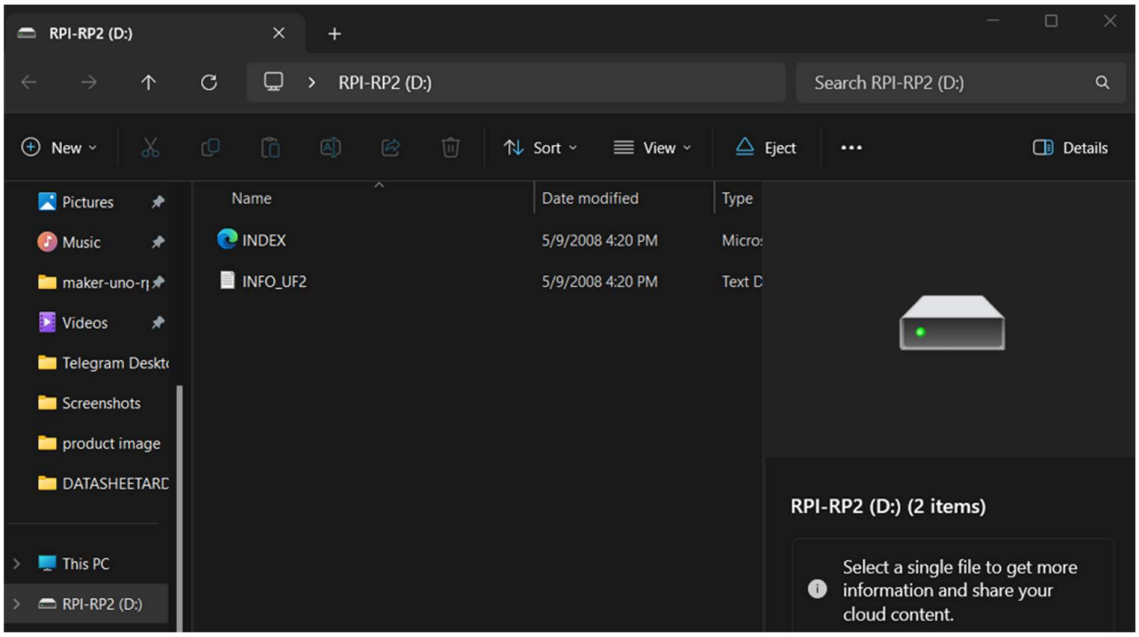
There are many software options you can use to program CircuitPython and MicroPython, even simple text editors like Notepad will do the work. For this guide, Thonny IDE is chosen as the coding software. Please download the IDE from the following link: https://thonny.org/

1. Connect the board to your laptop, power up the board and then enter bootloader mode by pressing and holding the “BOOT” button, followed by pressing the “RESET” button.



Make sure you switch on the board first before start flashing your code

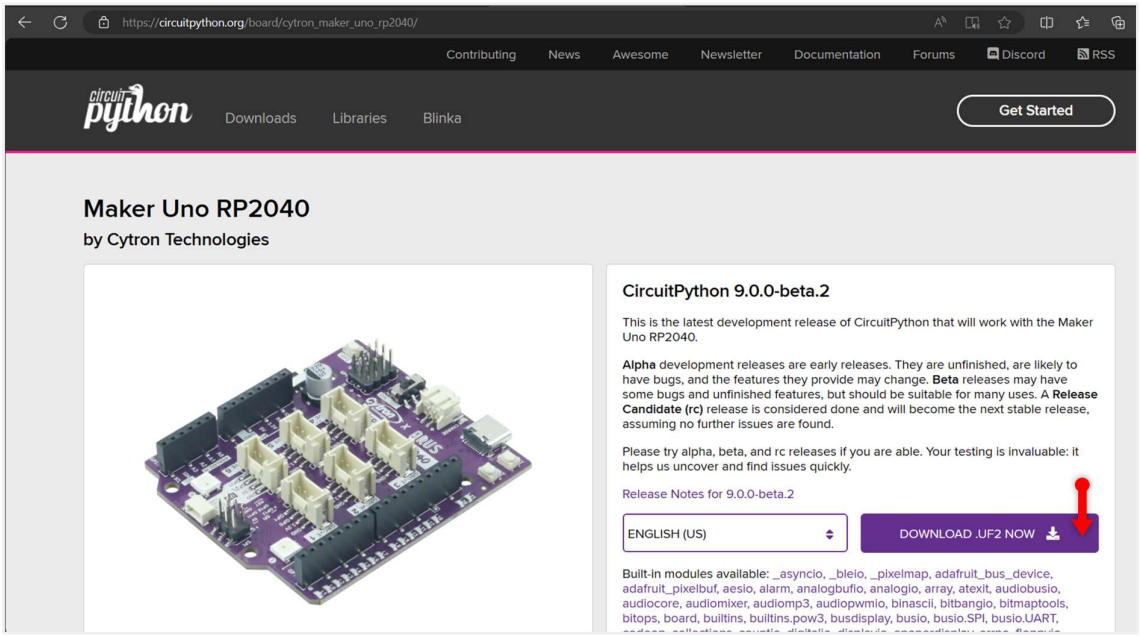
Continue to hold the “BOOT” button until the RPI-RP2 drive as shown in the figure below appeared



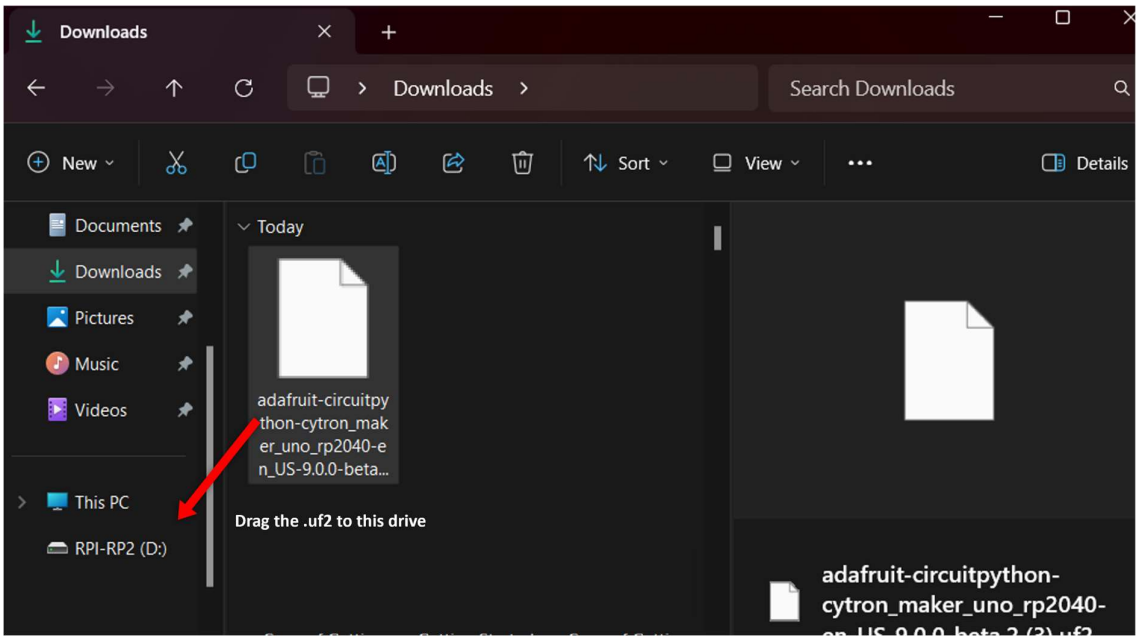
2. The next step is to flash the CircuitPython firmware onto the board. There are two ways it can be done;

   a. Paste the CircuitPython .uf2 file to the Raspberry Pi Drive

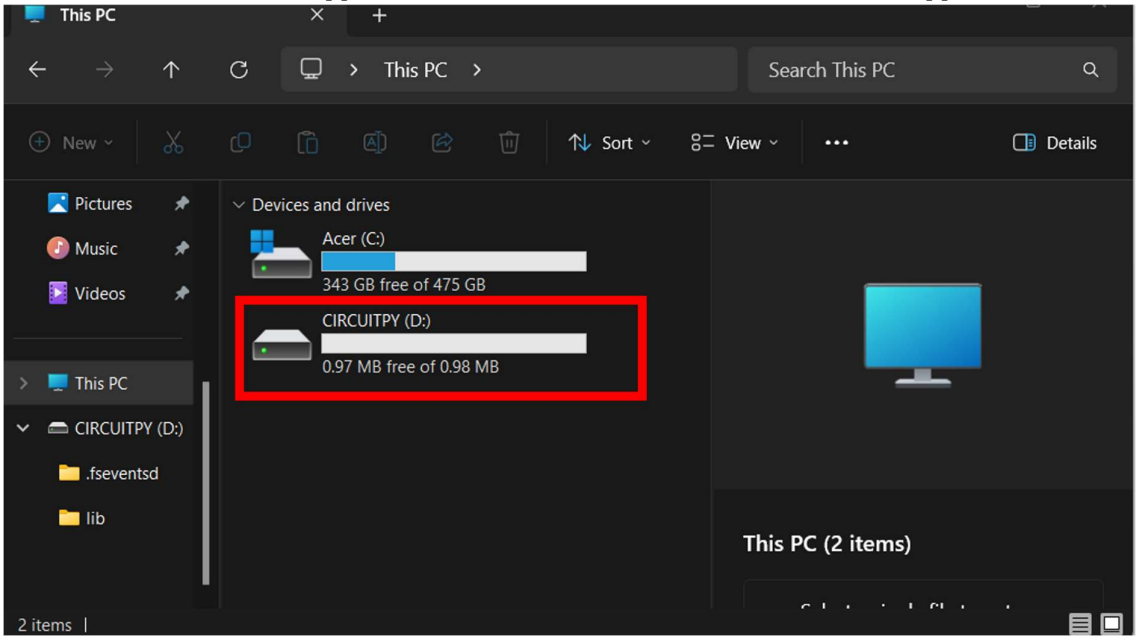
   i. Download the latest CircuitPython firmware for the Maker Uno RP2040 board from the CircuitPython website.



   ii. Drag the adafruit_circuitpython_etc.uf2 file you have downloaded to the RPI-RP2 drive.

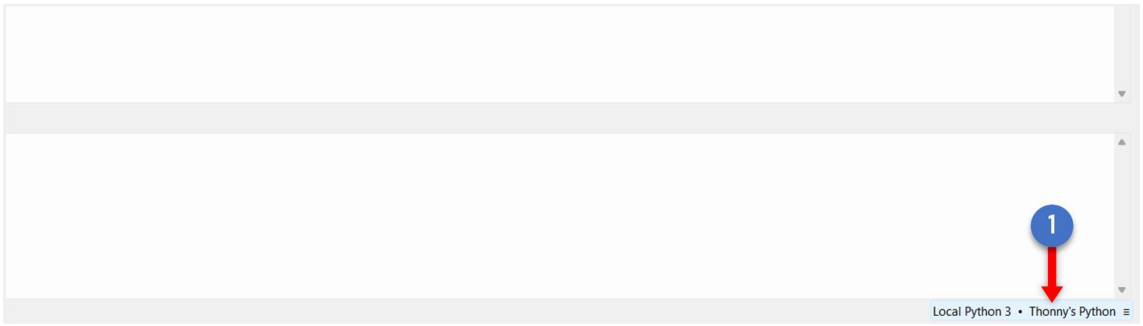


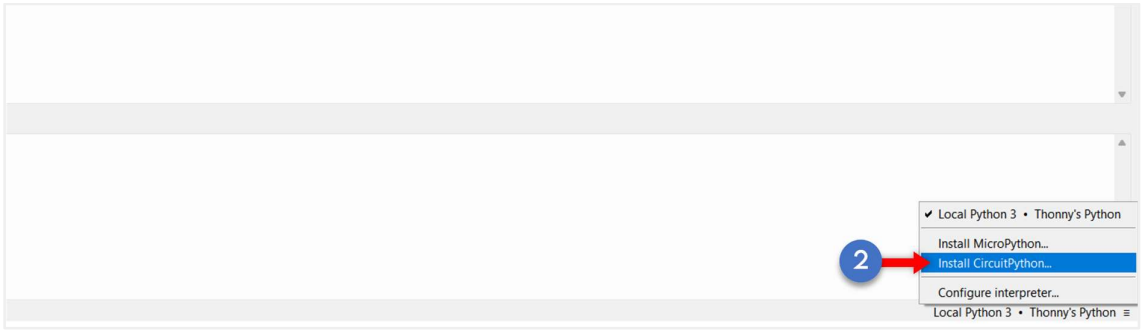
The RPI-RP2 drive will disappear and a new disk drive called CIRCUITPY will appear.



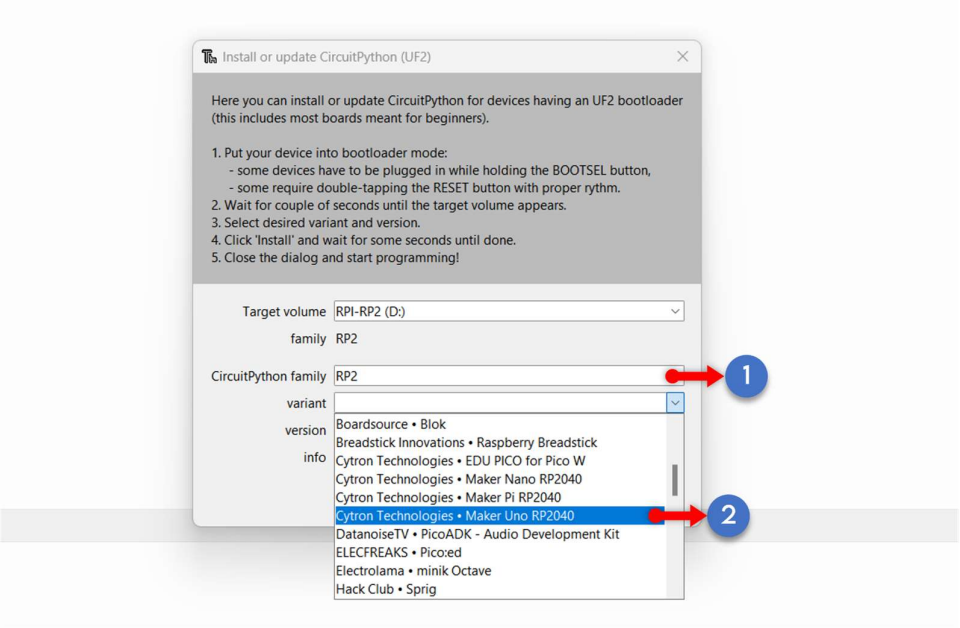
b. Paste the CircuitPython .uf2 file to the Raspberry Pi Drive

i. Open the Thonny IDE, look at the bottom right corner of the window, select 'Install CircuitPython...'.

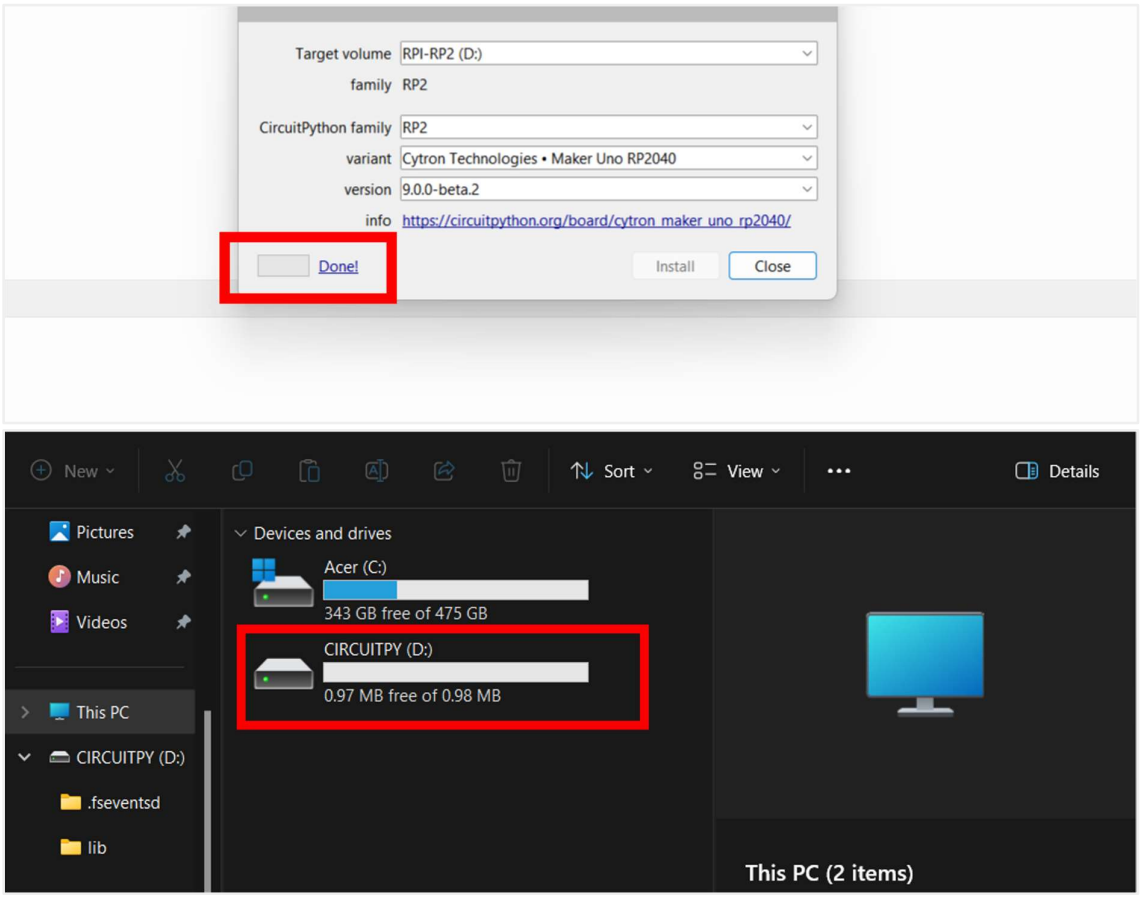




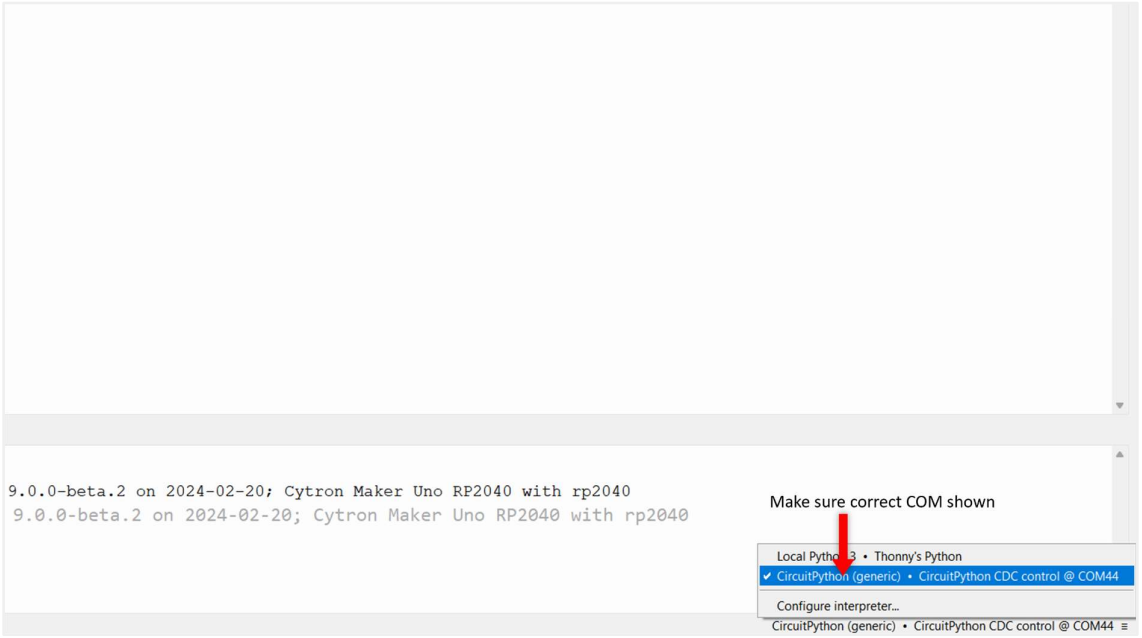
ii. Select the correct target volume and CircuitPython family as shown in the figure below. Then, choose the variant as "Cytron Technologies • Maker Uno RP2040"



iii. Click Install. You should get the output 'Done!' and a new disk drive called CIRCUITPY will appear.



3. Now let's try a simple code to verify that everything is working correctly. At the bottom right corner of the window make sure the correct Configure Interferer, **CircuitPython (generic)** and COM are shown and chosen.

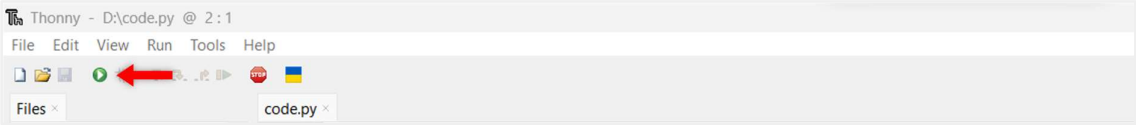


4. Copy this code below and Click the Run icon to run the code.

```
import board
import digitalio
import time

led = digitalio.DigitalInOut(board.GP1)
led.direction = digitalio.Direction.OUTPUT

while True:
    led.value = True
    time.sleep(0.5)
    led.value = False
    time.sleep(0.5)
```



5. The onboard LED 1 should be blinking by 0.5 second. To stop the execution of the program, click the STOP button or simply press CTRL+C.

Prepared by:

Cytron Technologies Sdn Bhd
www.cytron.io
No. 1, Lorong Industri Impian 1,
Taman Industri Impian,
14000 Bukit Mertajam,
Penang, Malaysia.

Tel: *+604 - 548 0668*
Fax: *+604 - 548 0669*

Email:
support@cytron.io
sales@cytron.io