Adapted from: https://internetofthingsagenda.techtarget.com/definition/microcontroller
https://www.dummies.com/article/technology/computers/hardware/arduino/what-is-arduino-258576

Module 4 – Blinking an LED with software

For this module you will need:

- ESP32
- Breadboard
- Male-to-Male jumper wires
- Resistor
- LED

Be sure the ESP32 is unplugged.

We will now interface our LED circuit with the ESP32 (a microcontroller). Doing so will allow us to write code that will turn the LED on and off, using the GPIO pins.

Assemble the circuit below:

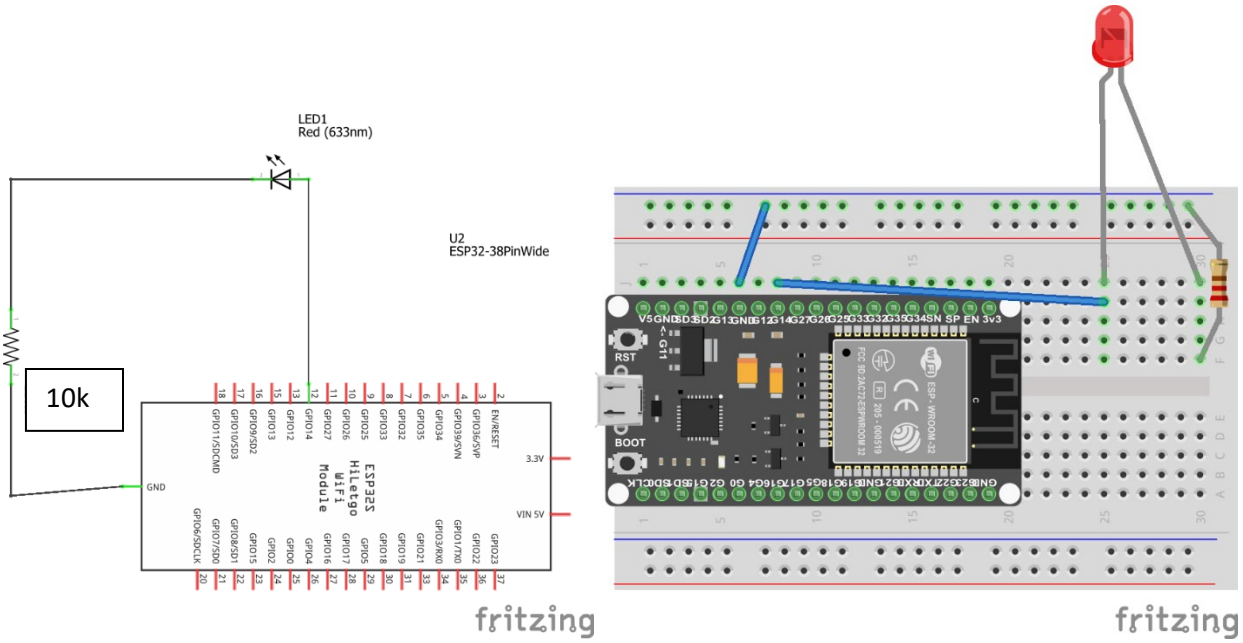
ESP32 GND to blue rail

GPIO14 to row 25

LED anode to row 25

LED cathode to row 30

10K resistor to row 30, the other end of resistor to blue rail



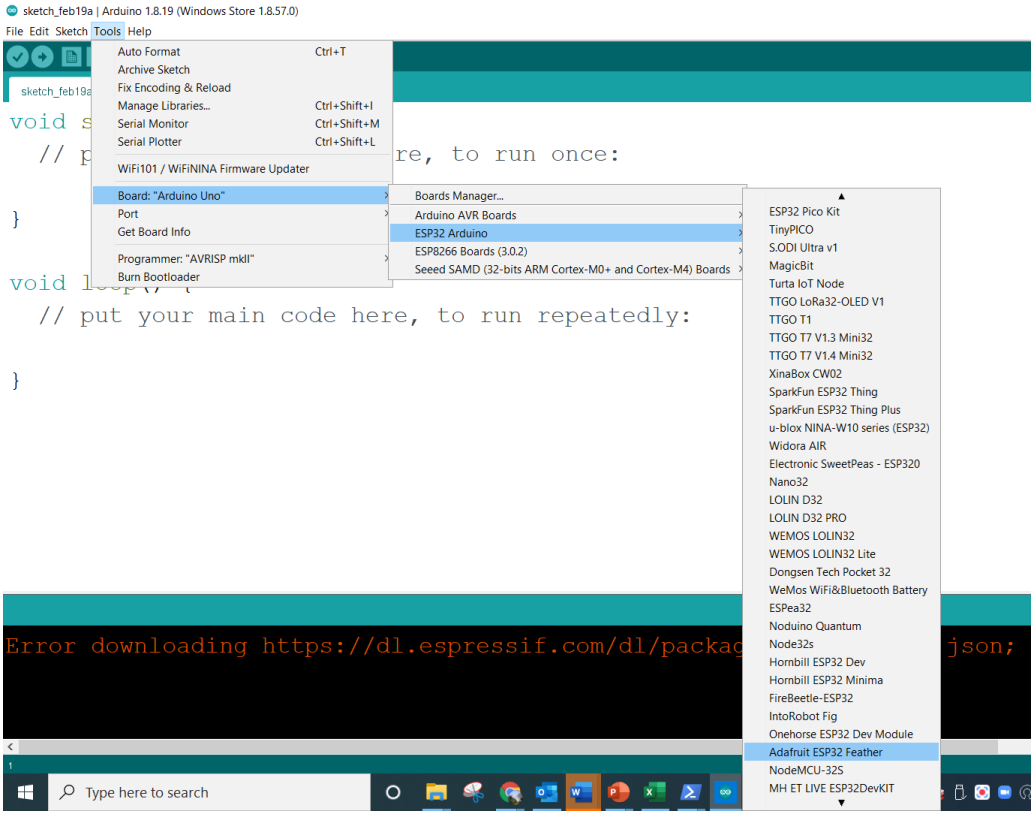
Adapted from: https://internetofthingsagenda.techtarget.com/definition/microcontroller
https://www.dummies.com/article/technology/computers/hardware/arduino/what-is-arduino-258576

Plug in the Arduino IDE, and open the Arduino IDE on the computer:

Type **Ctrl+s** to save the code (it is ok if it is blank).

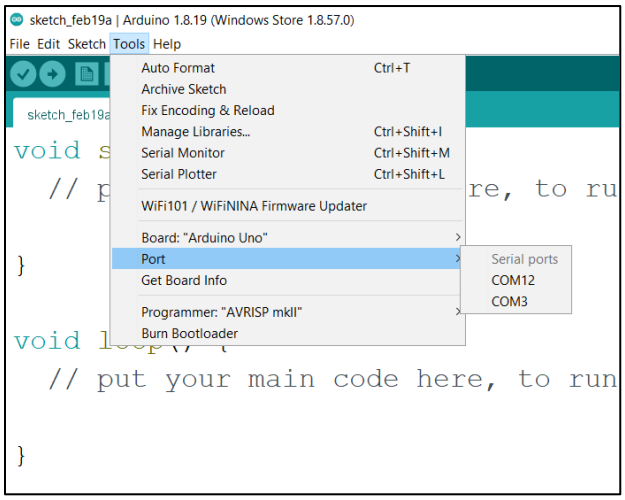
Let's make sure the Arduino IDE can contact the ESP32.

Click Tools->Board->ESP32->Adafruit ESP32 Feather

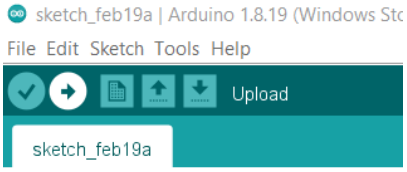


Adapted from: https://internetofthingsagenda.techtarget.com/definition/microcontroller
https://www.dummies.com/article/technology/computers/hardware/arduino/what-is-arduino-258576

With the ESP32 connected to the computer, select the COM port that is associated with the ESP32: ***Tools->Port->COMXX***

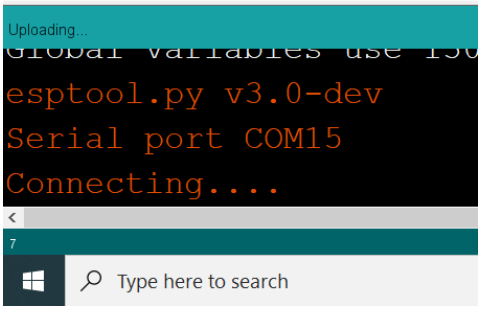


Click the arrow



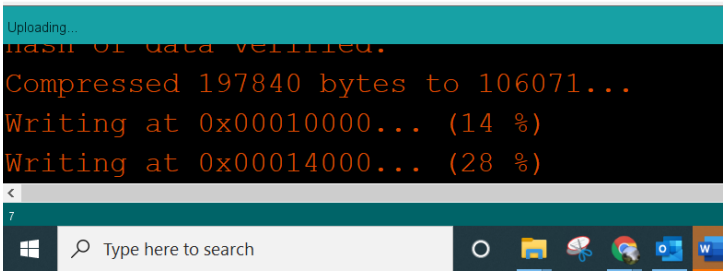
The code will then compile.

The code will then attempt to upload.



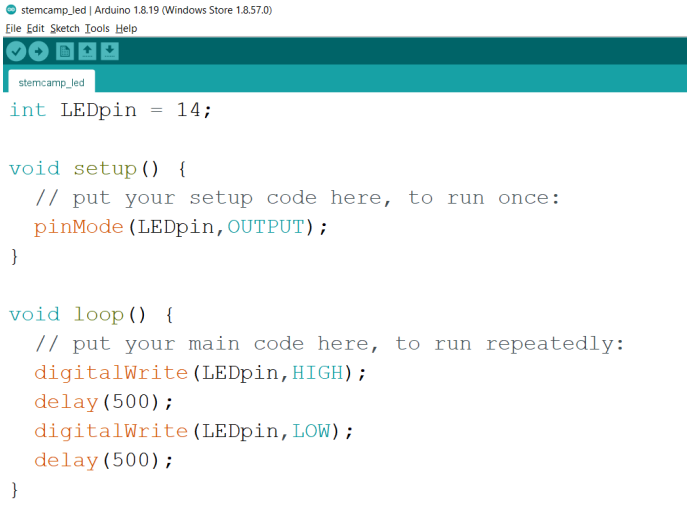
If this takes too long to connect, hold the boot button down, until you see: ***"Writing at…"***

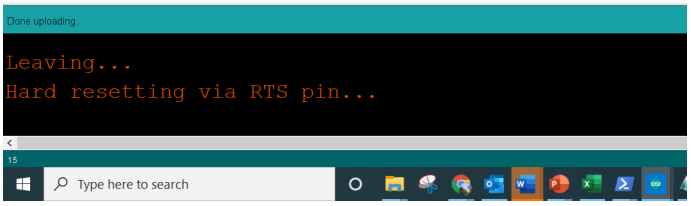
Adapted from: https://internetofthingsagenda.techtarget.com/definition/microcontroller
https://www.dummies.com/article/technology/computers/hardware/arduino/what-is-arduino-258576



If this still doesn't work, restart the IDE or choose another COMXX port.

Let's write the code to blink the LED.





Upload the code to the ESP32.

The LED will now blink!

What is a microcontroller?

The ESP32 is made up of both hardware and software. The ESP32 board is a printed circuit board (PCB) designed to use a microcontroller chip as well as other input and outputs. The board has many other electronic components that are needed for the microcontroller to function or to extend its capabilities.

A microcontroller is a small computer contained in a single, integrated circuit or computer chip. Microcontrollers are an excellent way to program and control electronics. Microcontroller boards have a microcontroller chip and other useful connectors and components that allow a user to attach inputs and outputs.

Adapted from: https://internetofthingsagenda.techtarget.com/definition/microcontroller
https://www.dummies.com/article/technology/computers/hardware/arduino/what-is-arduino-258576

Microcontrollers are used in a wide array of systems and devices. Devices often utilize multiple microcontrollers that work together within the device to handle their respective tasks.

For example, a car might have many microcontrollers that control various individual systems within, such as the anti-lock braking system, traction control, fuel injection or suspension control. All the microcontrollers communicate with each other to inform the correct actions. Some might communicate with a more complex central computer within the car, and others might only communicate with other microcontrollers. They send and receive data using their I/O peripherals and process that data to perform their designated tasks.

You write code in the Arduino software to tell the microcontroller what to to-do. For example, by writing a line of code, you can tell a light-emitting diode (LED) to blink on and off. If you connect a pushbutton and add another line of code, you can tell the LED to turn on only when the button is pressed. Next, you may want to tell the LED to blink only when the pushbutton is held down. In this way, you can quickly build a behavior for a system that would be difficult to achieve without a microcontroller.

Like a conventional computer, an ESP32 can perform a multitude of functions, but it's not much use on its own. It requires inputs or outputs to make it useful. These inputs and outputs allow a computer — and an ESP32 — to sense objects in the world and to affect the world.