LROBRUYA

Raspberry Pi Pico Starter Kit

--Piper Make

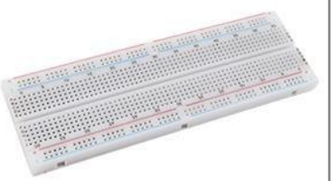
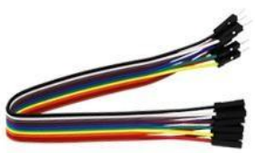
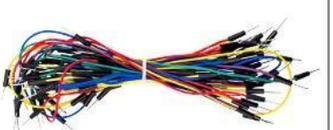
Content

Raspberry Pi Pico Introduction..... 2
Specifications..... 3
Pinout.....4
Getting Started with Piper Make.....7
Burning piper circuitpython Firmware (important).....7
How to use Piper Make(important)..... 16
Project 1 Hello, LED!(important).....22
Wiring..... 23
Code..... 24
Code Explanation..... 28
Project 2 Button Control LED.....29
Component knowledge..... 29
Wiring..... 30
Code..... 30
Project 3 Tilt Switch Control LED..... 32
Wiring..... 33
Code..... 33
Project 4 Intruder Alarm.....36
Wiring..... 37
Code..... 38
Project 5 Photoresistor Control LED.....40
Wiring..... 41
Code..... 41
Code Explanation..... 45
Project 6 Servo.....46
Wiring..... 47
Code..... 48
Project 7 RGB LED Strip..... 50
Wiring..... 51
Code..... 52

Code Explanation..... 54
Components Introduction..... 55
- Breadboard..... 56
- LED..... 58
- Button..... 59
- RGB LED..... 60
- Resistor..... 62
- Transistor..... 64
- Buzzer..... 66
- Potentiometer..... 68
- IR Proximity Sensor Module..... 69
- Photoresistor..... 71
- Thermistor..... 73
- Tilt Switch..... 75
- Servo..... 76
- I2C LCD1602..... 77
- PIR Motion Sensor..... 79
- WS2812 RGB 8 LEDs Strip..... 82

FAQ..... 84

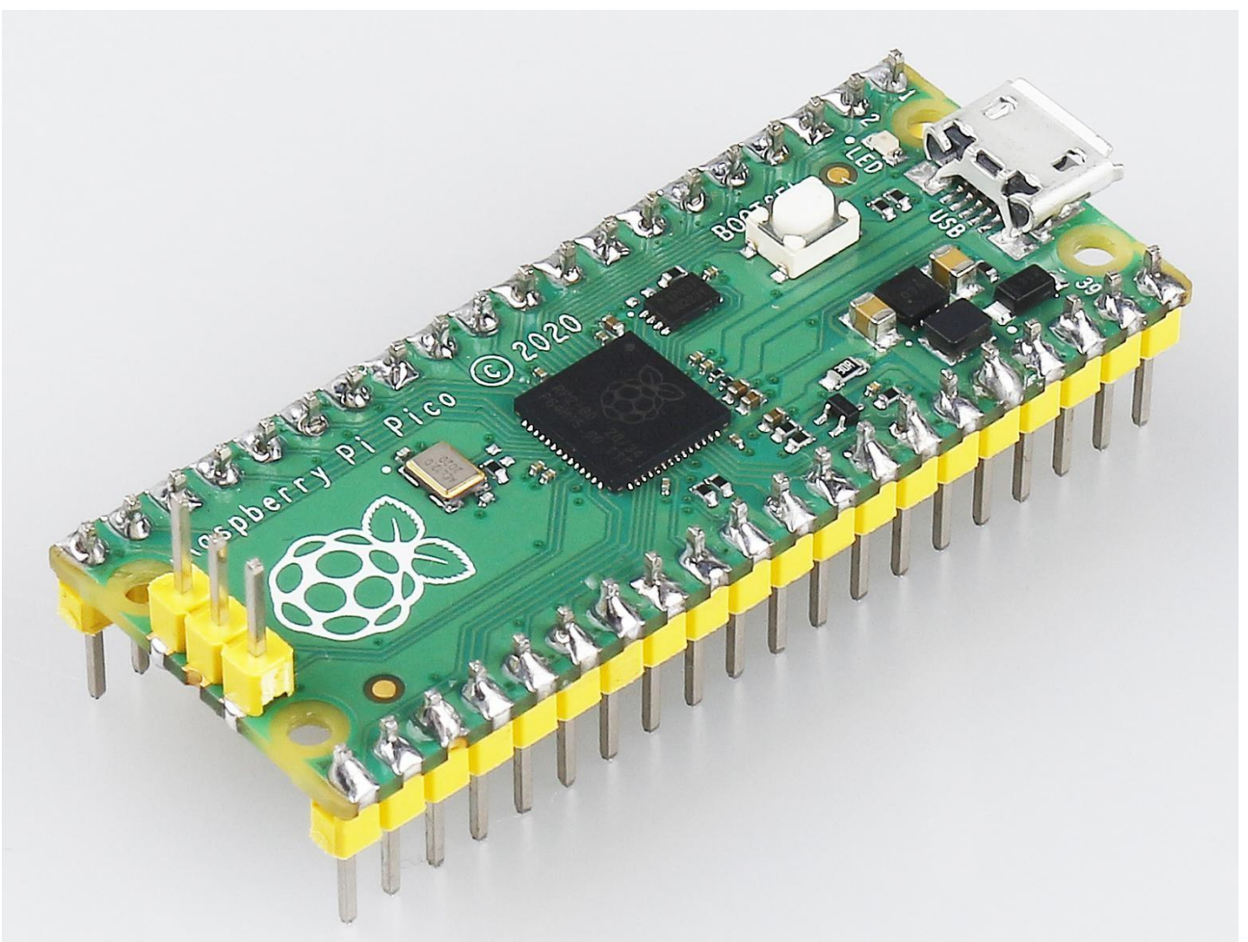
Packing List

 Raspberry Pi Pico x1	 IIC LCD 1602 x1	 830 Tie-Points Breadboard x1	 Obstacle Avoidance Module x1
 HC-SR501 PIR Motion Sensor x1	 SG90 Servo x1	 Potentiometer (10k) x1	 Passive Buzzer x1 Active Buzzer x1
 Thermistor x1 Tilt Switch x1	 Photoresistor x2 S8050 Transistor x2	 Button Switch x8	 Resistor-220R/1k/10k x30
 F-M DuPont Cable x10	 65 Jumper Wire x1	 Micro-USB Cable x1	 WS2812 RGB 8 LEDSs Strip x1
 LED-RGB x5	 LED-Yellow x5	 LED-Green x5	 LED-RGB x1

Raspberry Pi Pico Introduction

The Raspberry Pi Pico is a microcontroller board based on the Raspberry Pi RP2040 microcontroller chip.

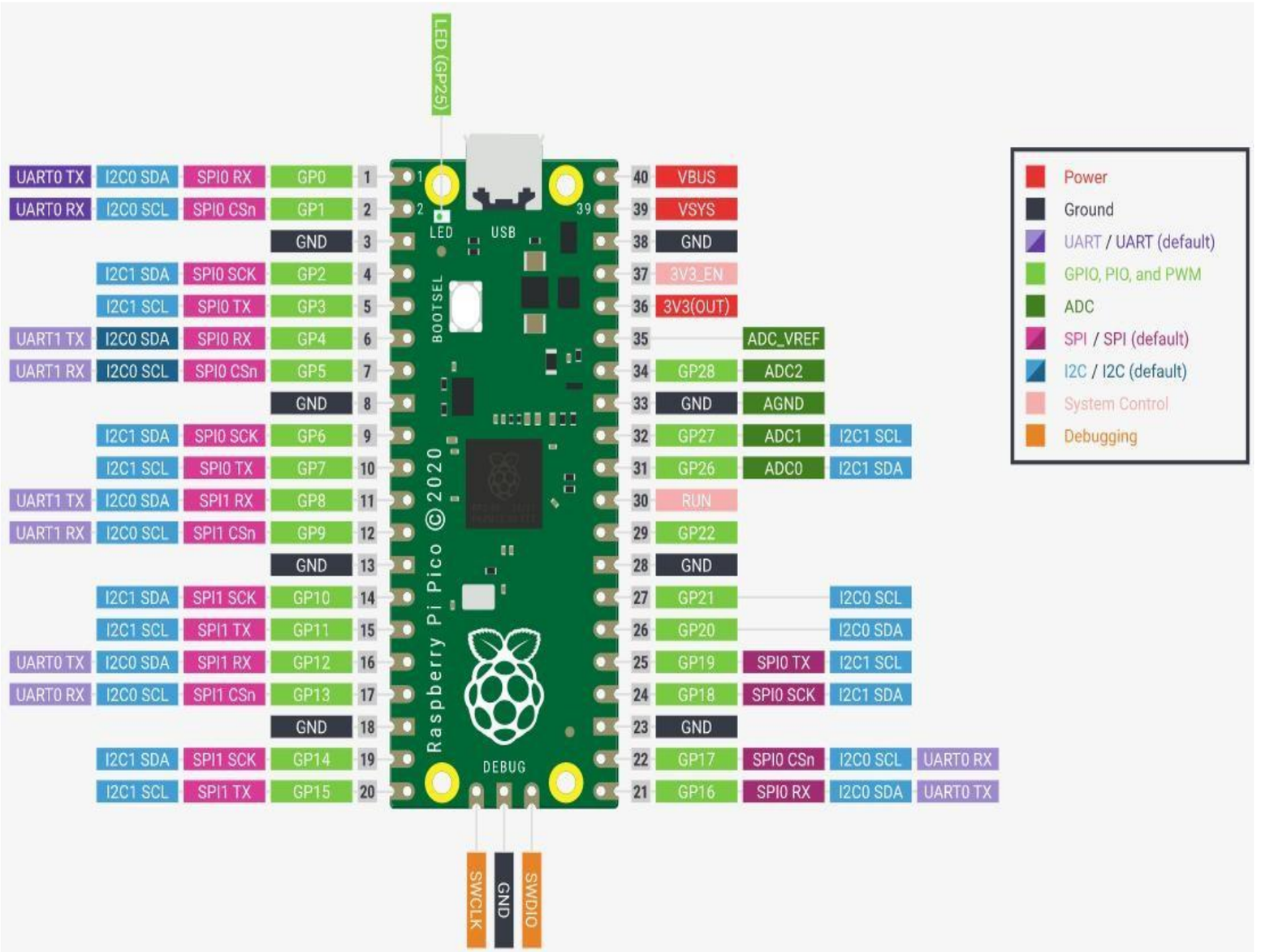
Whether you want to learn the MicroPython programming language, take the first step in physical computing, or want to build a hardware project, Raspberry Pi Pico and its amazing community – will support you every step of the way. In the project, it can control anything, from LEDs and buttons to sensors, motors, and even other microcontrollers.



Specifications

- 21 mm × 51 mm form factor
- RP2040 microcontroller chip designed by Raspberry Pi in the UK
- Dual-core Arm Cortex-M0+ processor, flexible clock running up to 133 MHz
- 264KB on-chip SRAM
- 2MB on-board QSPI Flash
- 26 multifunction GPIO pins, including 3 analog inputs
- 2 × UART, 2 × SPI controllers, 2 × I2C controllers, 16 × PWM channels
- 1 × USB 1.1 controller and PHY, with host and device support
- 8 × Programmable I/O (PIO) state machines for custom peripheral support
- Supported input power 1.8–5.5V DC
- Operating temperature -20°C to +85°C
- Castellated module allows soldering direct to carrier boards
- Drag-and-drop programming using mass storage over USB
- Low-power sleep and dormant modes
- Accurate on-chip clock
- Temperature sensor
- Accelerated integer and floating-point libraries on-chip

Pinout


LED (GP25)
LED
USB
BOOTSEL
DEBUG
Raspberry Pi Pico © 2020
SWCLK
GND
SWDIO
Power
Ground
UART / UART (default)
GPIO, PIO, and PWM
ADC
SPI / SPI (default)
I2C / I2C (default)
System Control
Debugging
UART0 TX I2C0 SDA SPI0 RX GP0 1
UART0 RX I2C0 SCL SPI0 CSn GP1 2
GND 3
I2C1 SDA SPI0 SCK GP2 4
I2C1 SCL SPI0 TX GP3 5
UART1 TX I2C0 SDA SPI0 RX GP4 6
UART1 RX I2C0 SCL SPI0 CSn GP5 7
GND 8
I2C1 SDA SPI0 SCK GP6 9
I2C1 SCL SPI0 TX GP7 10
UART1 TX I2C0 SDA SPI1 RX GP8 11
UART1 RX I2C0 SCL SPI1 CSn GP9 12
GND 13
I2C1 SDA SPI1 SCK GP10 14
I2C1 SCL SPI1 TX GP11 15
UART0 TX I2C0 SDA SPI1 RX GP12 16
UART0 RX I2C0 SCL SPI1 CSn GP13 17
GND 18
I2C1 SDA SPI1 SCK GP14 19
I2C1 SCL SPI1 TX GP15 20
40 VBUS
39 VSYS
38 GND
37 3V3_EN
36 3V3(OUT)
35 ADC_VREF
34 GP28 ADC2
33 GND AGND
32 GP27 ADC1 I2C1 SCL
31 GP26 ADC0 I2C1 SDA
30 RUN
29 GP22
28 GND
27 GP21 I2C0 SCL
26 GP20 I2C0 SDA
25 GP19 SPI0 TX I2C1 SCL
24 GP18 SPI0 SCK I2C1 SDA
23 GND
22 GP17 SPI0 CSn I2C0 SCL UART0 RX
21 GP16 SPI0 RX I2C0 SDA UART0 TX

Name	Description	Function
GP0-GP28	General-purpose input/output pins	Act as either input or output and have no fixed purpose of their own
GND	0 volts ground	Several GND pins around Pico to make wiring easier.
RUN	Enables or diables your Pico	Start and stop your Pico from another microcontroller.
GPxx_ADCx	General-purpose input/output or analog input	Used as an analog input as well as a digital input or output – but not both at the same time.
ADC_VREF	Analog-to-digital converter (ADC) voltage reference	A special input pin which sets a reference voltage for any analog inputs.
AGND	Analog-to-digital converter (ADC) 0 volts ground	A special ground connection for use with the ADC_VREF pin.
3V3(O)	3.3 volts power	A source of 3.3V power, the same voltage your Pico runs at internally, generated from the VSYS input.
3v3(E)	Enables or disables the power	Switch on or off the 3V3(O) power, can also switches your Pico off.
VSYS	2-5 volts power	A pin directly connected to your Pico's internal power supply, which cannot be switched off without also switching Pico off.
VBUS	5 volts power	source of 5 V power taken from your Pico's micro USB port, and used to power hardware which needs more than 3.3 V.

The best place to find everything you need to get started with your Raspberry Pi Pico.

Or you can click on the links below:

- Raspberry Pi Pico product brief
- Raspberry Pi Pico datasheet
- Getting started with Raspberry Pi Pico: C/C++ development
- Raspberry Pi Pico C/C++ SDK
- API-level Doxygen documentation for the Raspberry Pi Pico C/C++ SDK
- Raspberry Pi Pico Python SDK
- Raspberry Pi RP2040 datasheet
- Hardware design with RP2040
- Raspberry Pi Pico design files
- Raspberry Pi Pico STEP file

Getting Started with Piper Make

Piper Make is a super easy and fun way to make projects using Raspberry Pi Pico. It uses blocks like Scratch, so you don't need any programming experience to use it. The underlying principle is to use CircuitPython with auxiliary libraries.

Burning piper circuitpython Firmware (important)

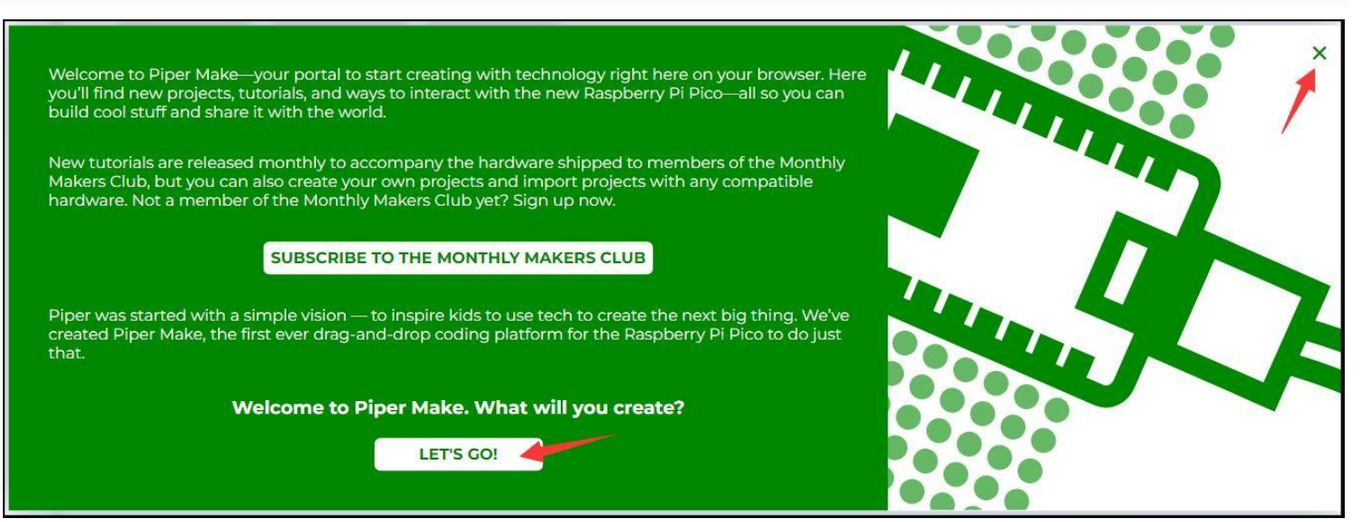
There There are two ways to burn the firmware, choose the one that suits you:

- **Drag and drop** is like copying a downloaded file from your computer to a USB memory stick.
- **Direct Burning** sends your firmware from the code editor direct to your Pico. It works on any computer in two popular web browsers.

● Drag and drop

① First,visit Piper Make through the following link：https://make.playpiper.com/

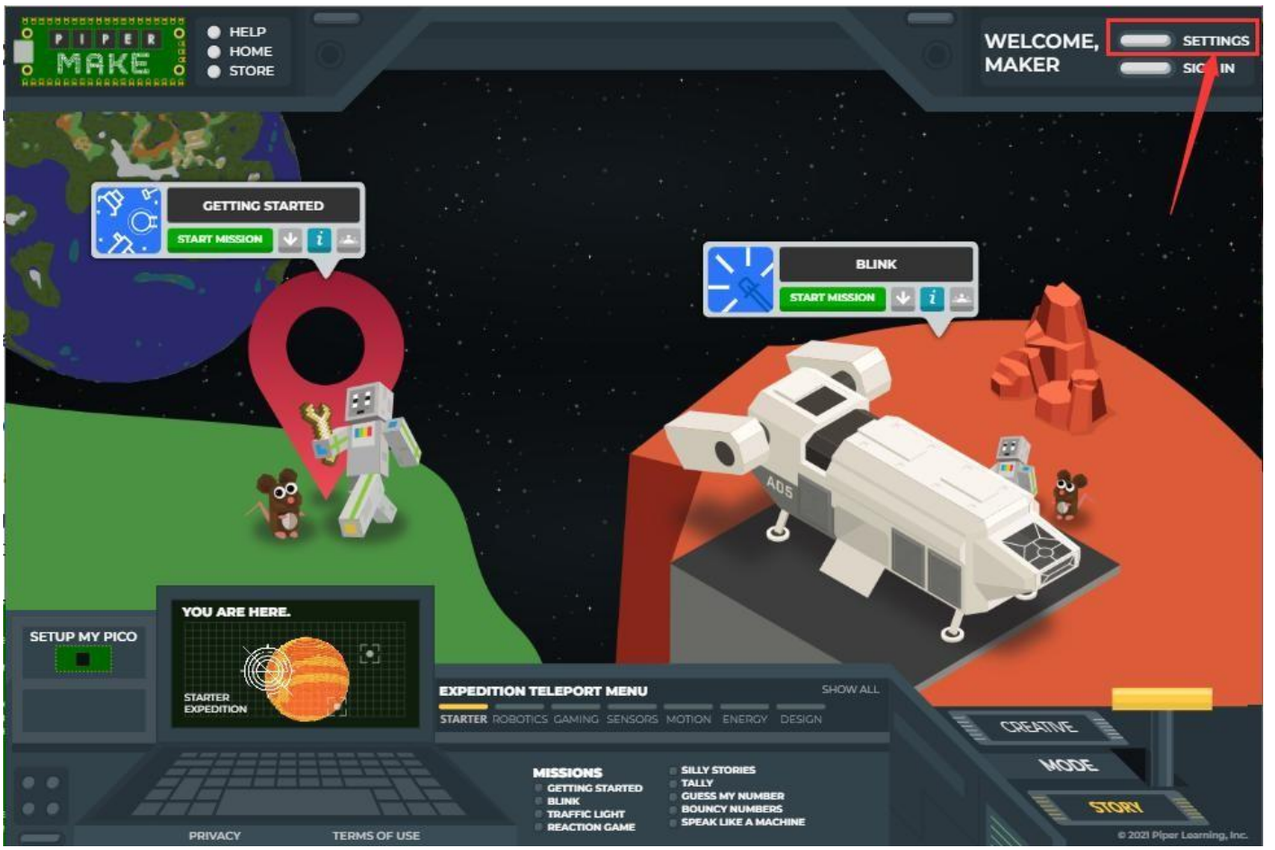
② In the pop-up page, if you don't need to subscribe for more tutorials, you can just click **Let's Go!** or the **X** button.



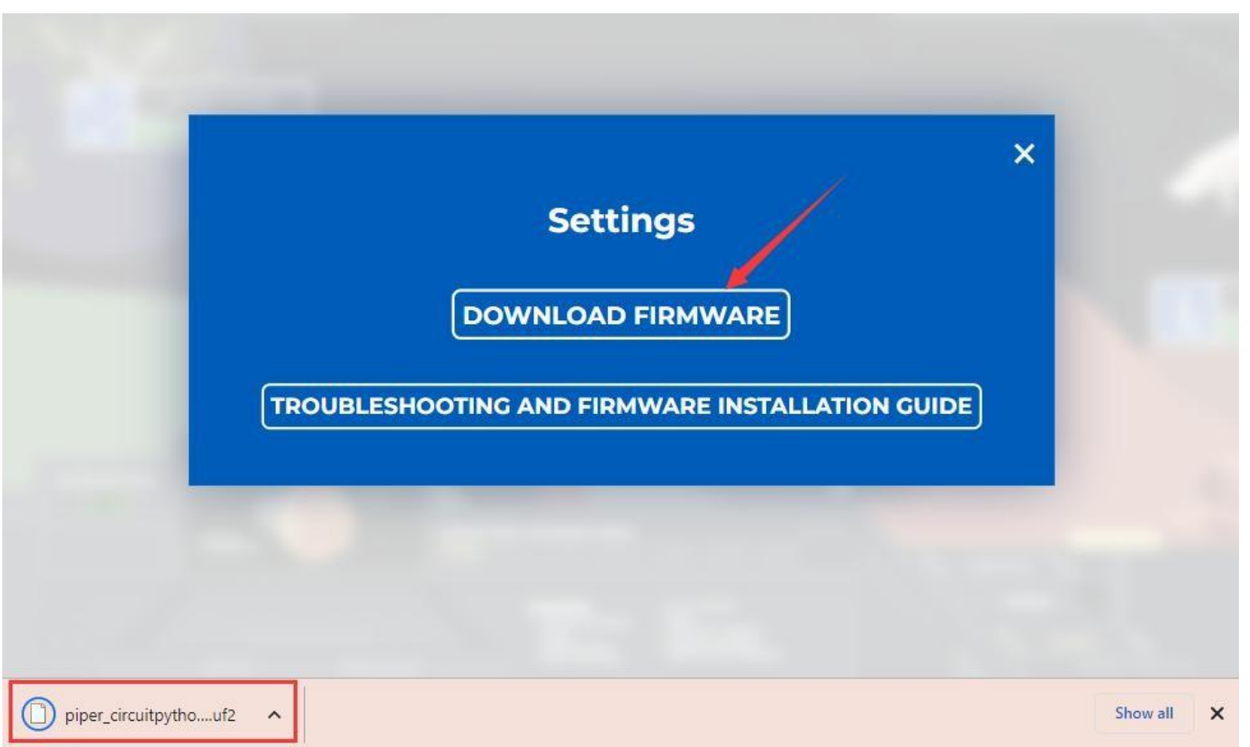
Note

If you see a different pop-up window, your browser version is not supported, please update your browser and try again.

③ Click "**SETTINGS**" in the upper right corner of the page

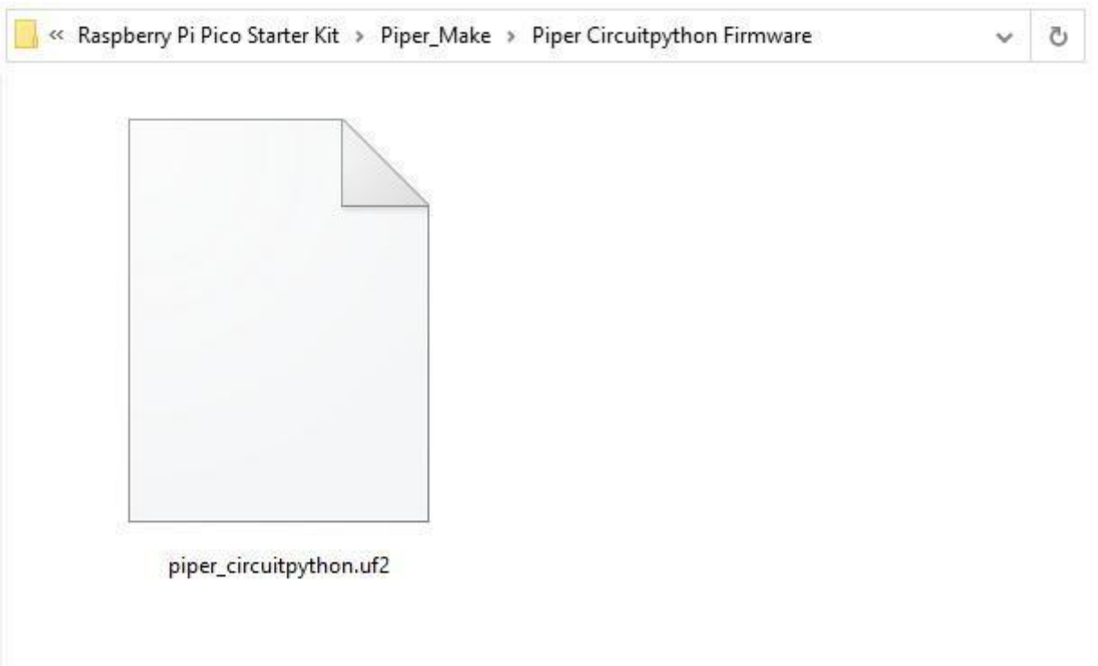


④ Click "**DOWNLOAD FIRMWARE**"



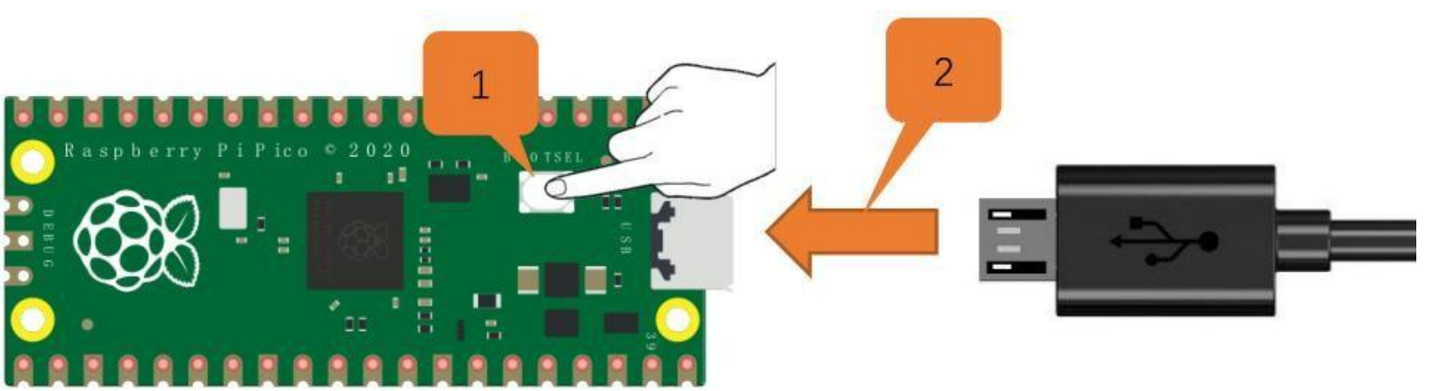
If you cannot download it due to network issue or other reasons, you can use the one we have prepared,which locates at the following file path:

Raspberry Pi Pico Starter Kit/Piper Make/Piper Circuitpython Firmware/piper_circuitpython.uf2

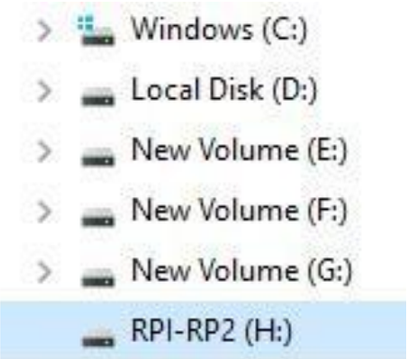


⑤ connect a USB cable to your computer.

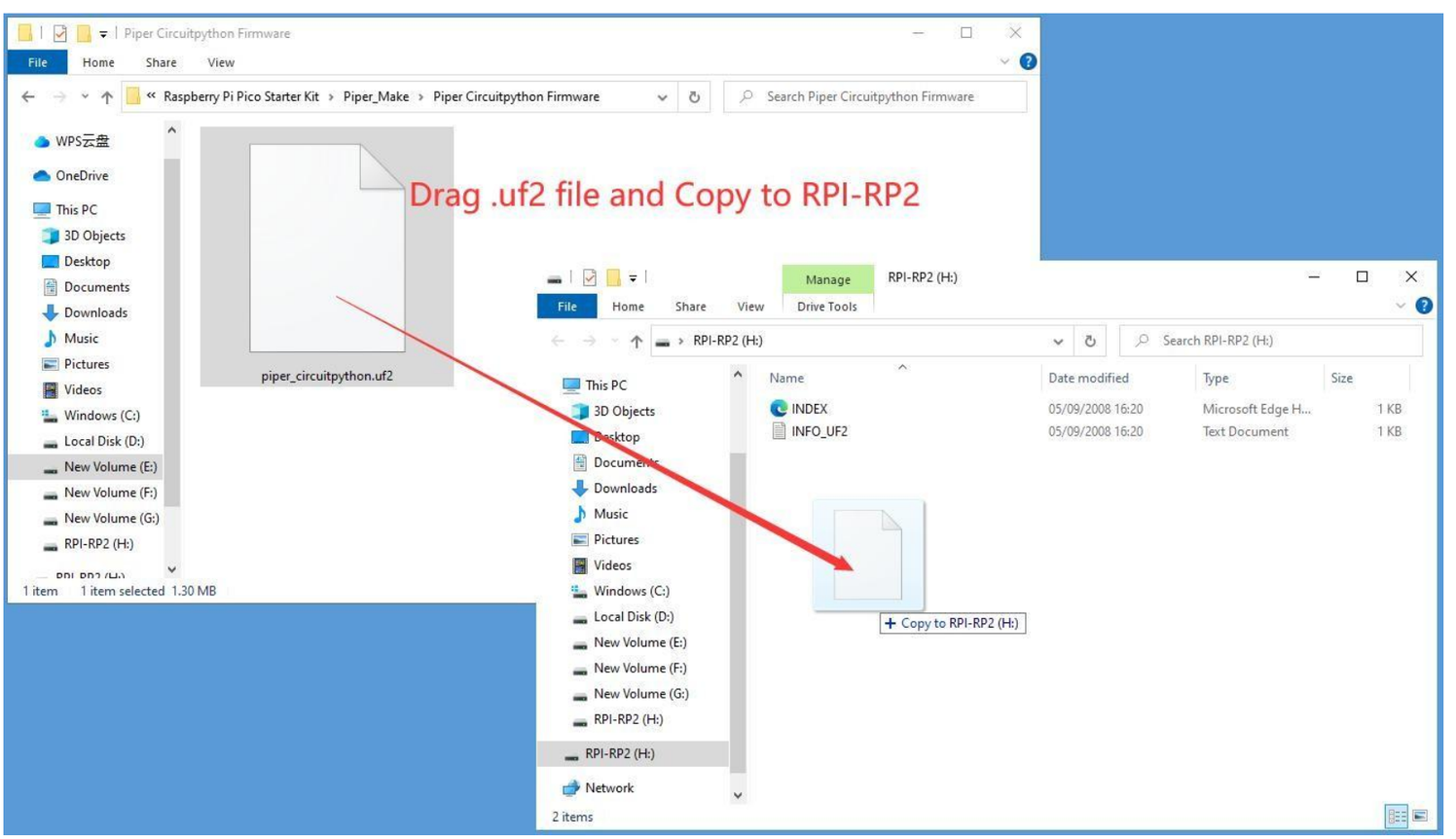
⑥ Long press BOOTSEL button on Raspberry Pi Pico and connect it to your computer with the USB cable.



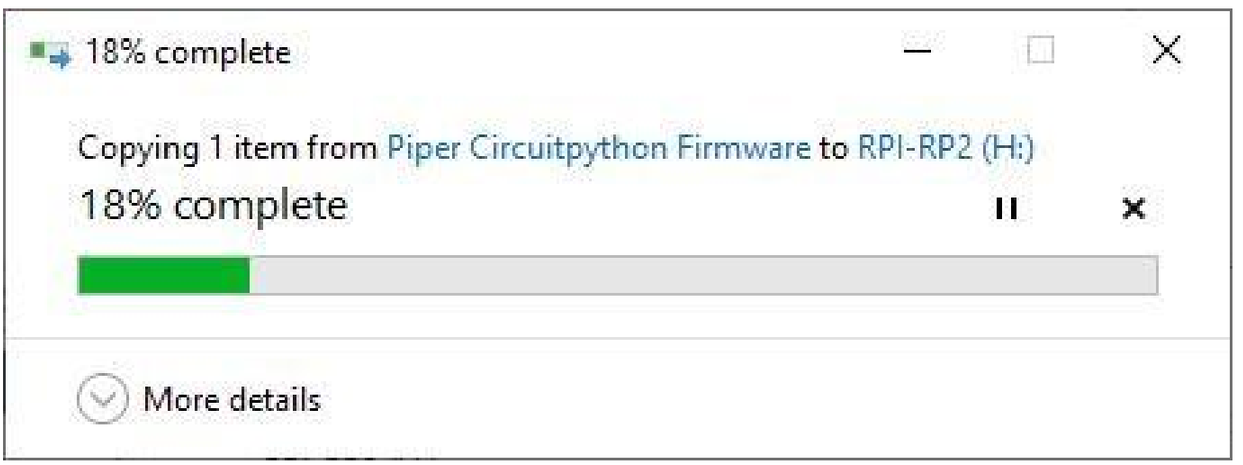
⑦ When the connection succeeds, A new disk called **RPI-RP2(:)** will appear in the file manager.

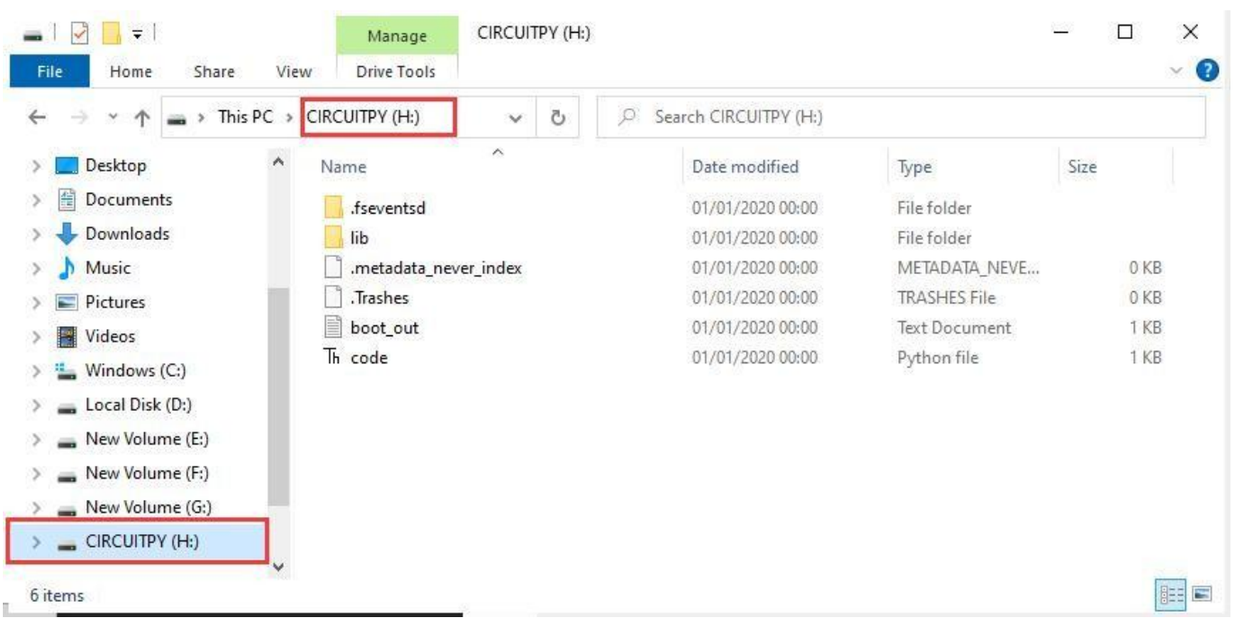


⑧ Copy the file(**piper_circuitpython.uf2**) to RPI-RP2 and wait for it to finish, just like copy file to a U disk.



⑨ When the firmware finishes burning, Raspberry Pi Pico will reboot automatically.The Raspberry Pi Pico will be recognized as a disk named “CIRCUITPY”.



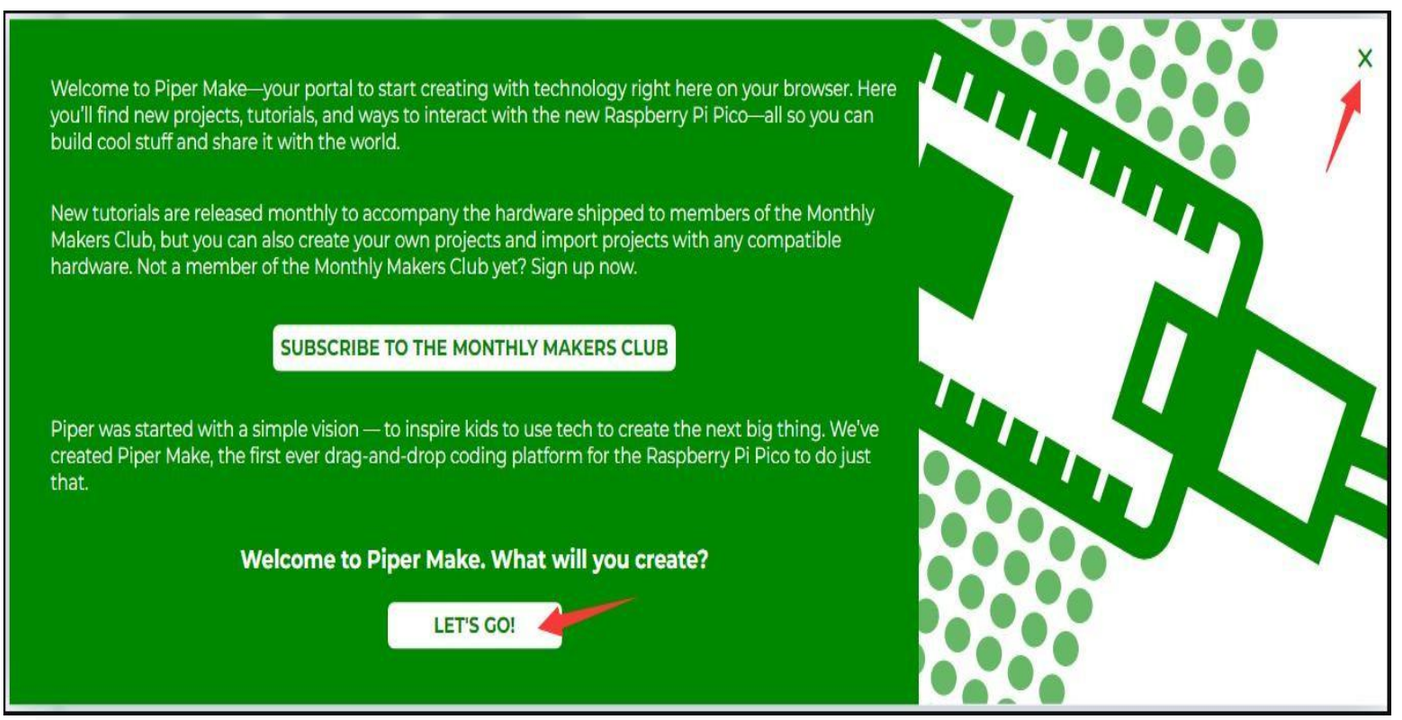


Tip:Have a question about burn firmware?

● Direct Burning

① First,visit Piper Make through the following link：https://make.playpiper.com/

② In the pop-up page, if you don’t need to subscribe for more tutorials, you can just click **Let’s Go!** or the **X** button.

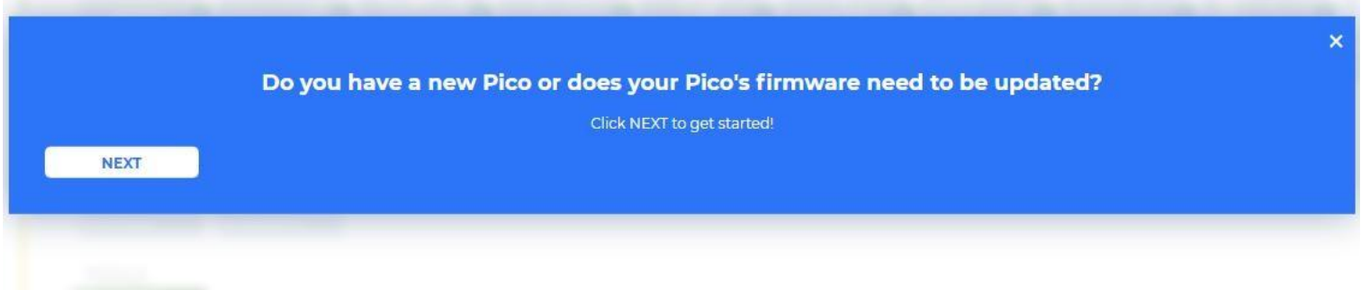


Note

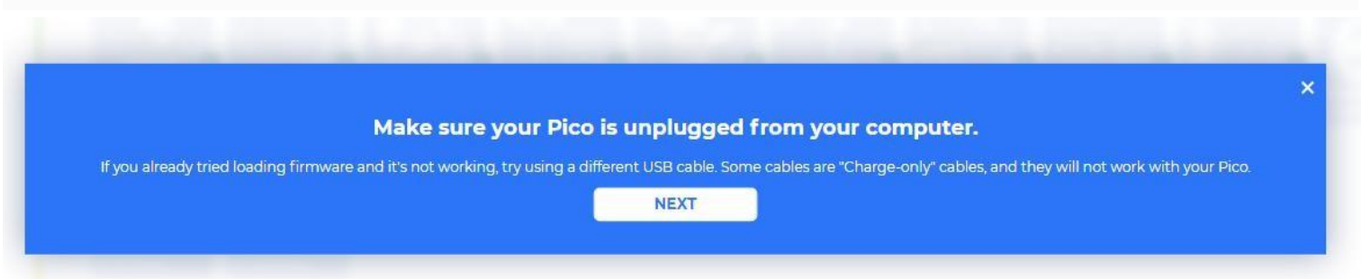
If you see a different pop-up window, your browser version is not supported, please update your browser and try again.

③ Scroll to the bottom of this page and click on the **Set up my Pico** under the **Tools** section and follow the prompts to configure it.

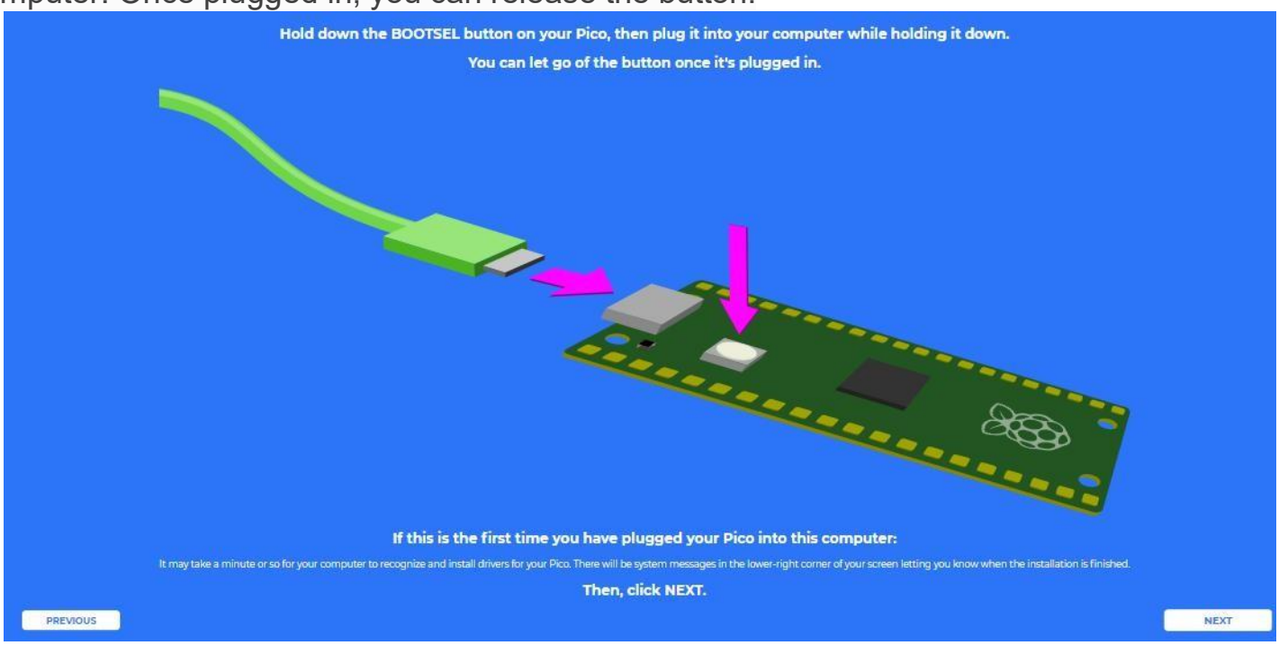
④ Click **Next** to start configuring your Pico, even if you have set it up before, these are the same steps you will use to update your Pico firmware.



⑤ In this step, you need to make sure that your Pico is unplugged from your computer, as it needs to be plugged in in a specific way in the next step. Make sure your cable can handle power and data, as many micro USB cables only have power.



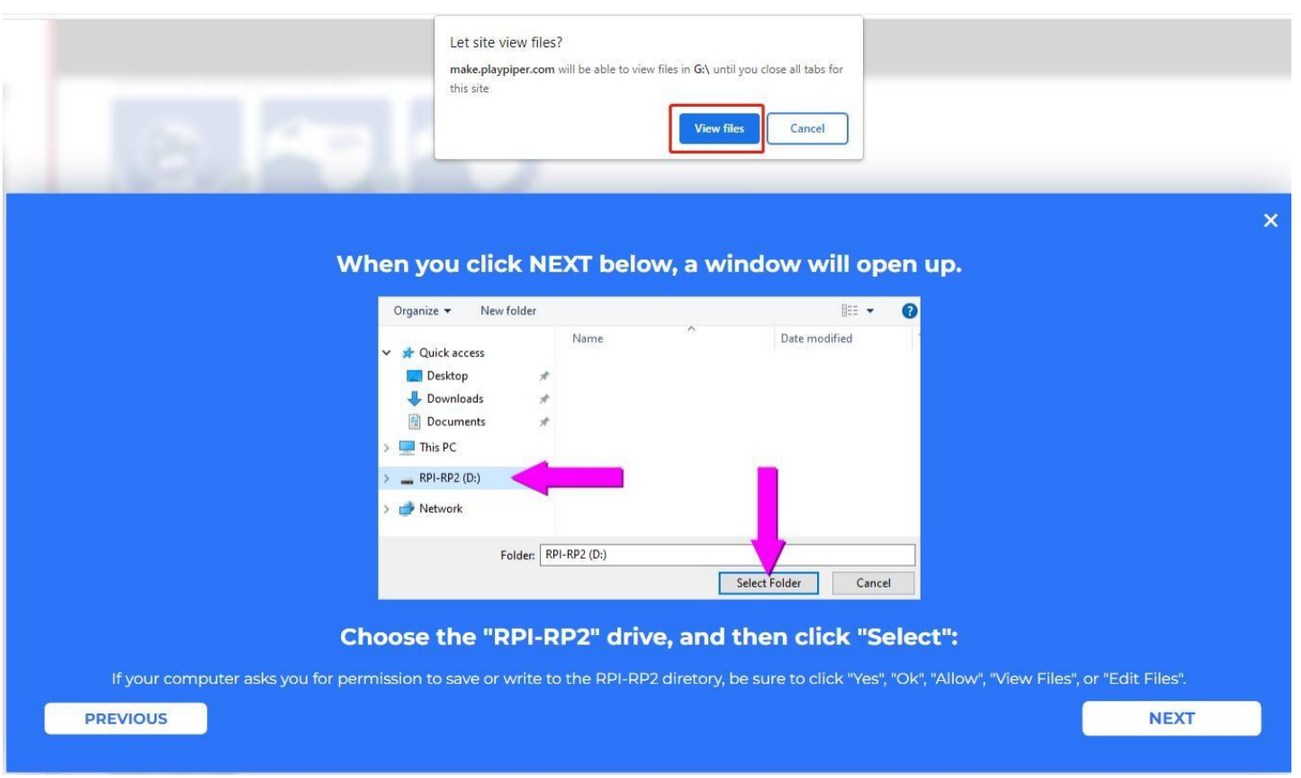
⑥ Now, press and hold the RST (white) button on the Pico and plug the Pico into your computer. Once plugged in, you can release the button.



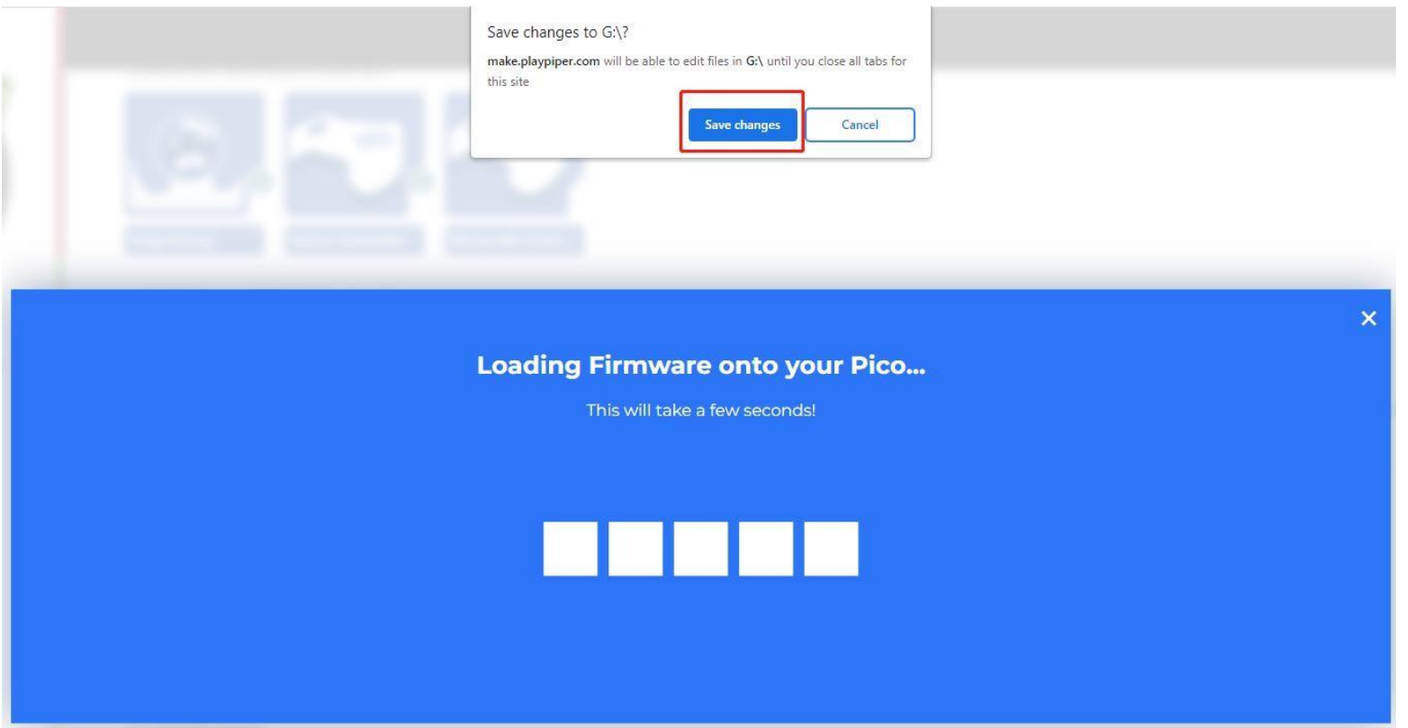
⑦ Your Pico will appear as a USB drive, click **Next** after that select **RPI-RP2** drive.

Note

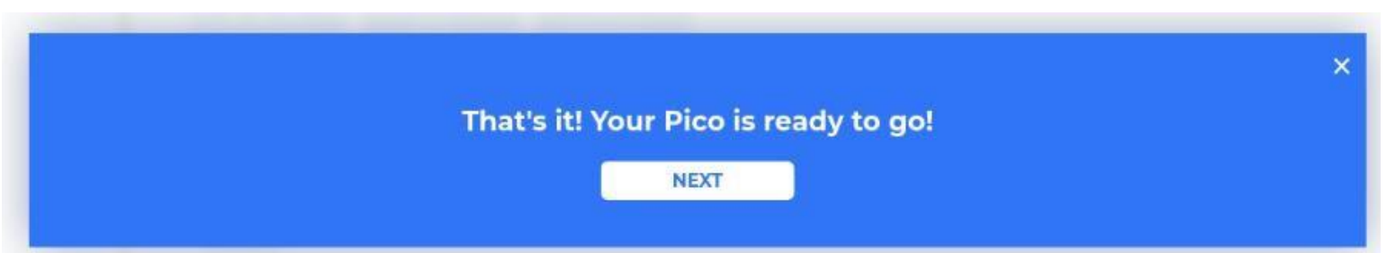
After select **RPI-RP2** drive, there will be a pop up window at the top that you need to allow the web page to view files.



⑧ Now Piper Make will load the firmware to your Pico, again you need to allow save changes to the hard drive where the Pico is located.



⑨ When this prompt appears, it means your Pico is set up and you can start using it.

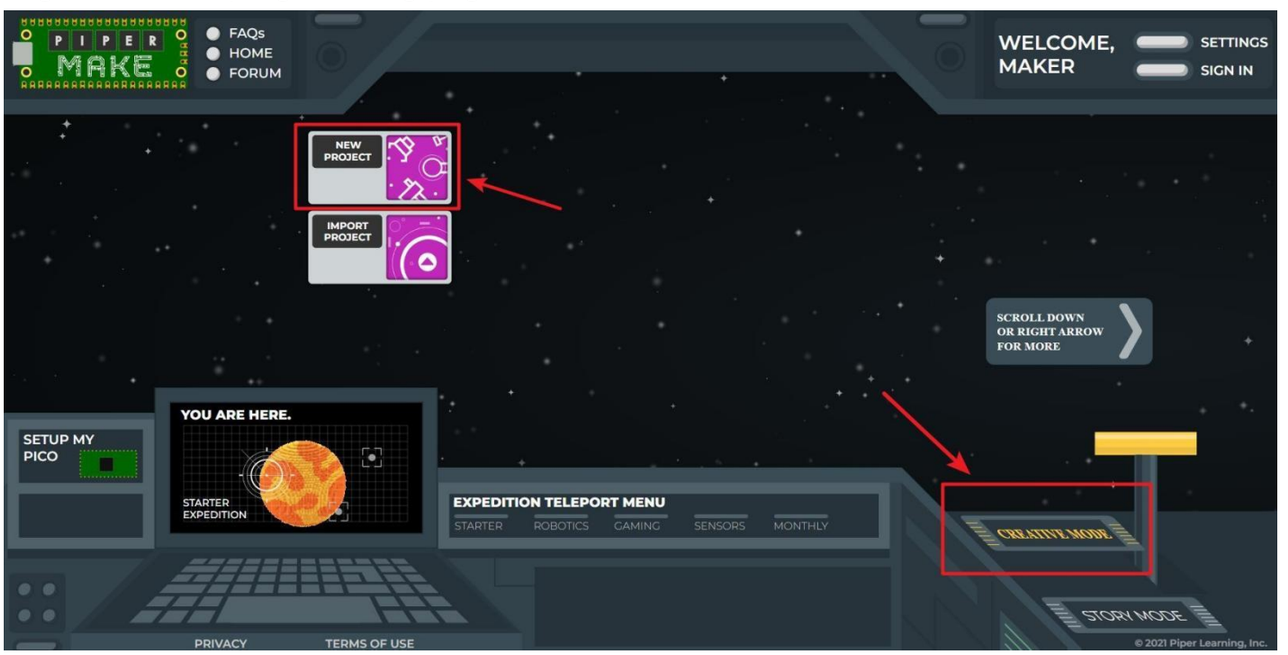


Tip:Have a question about burn firmware?

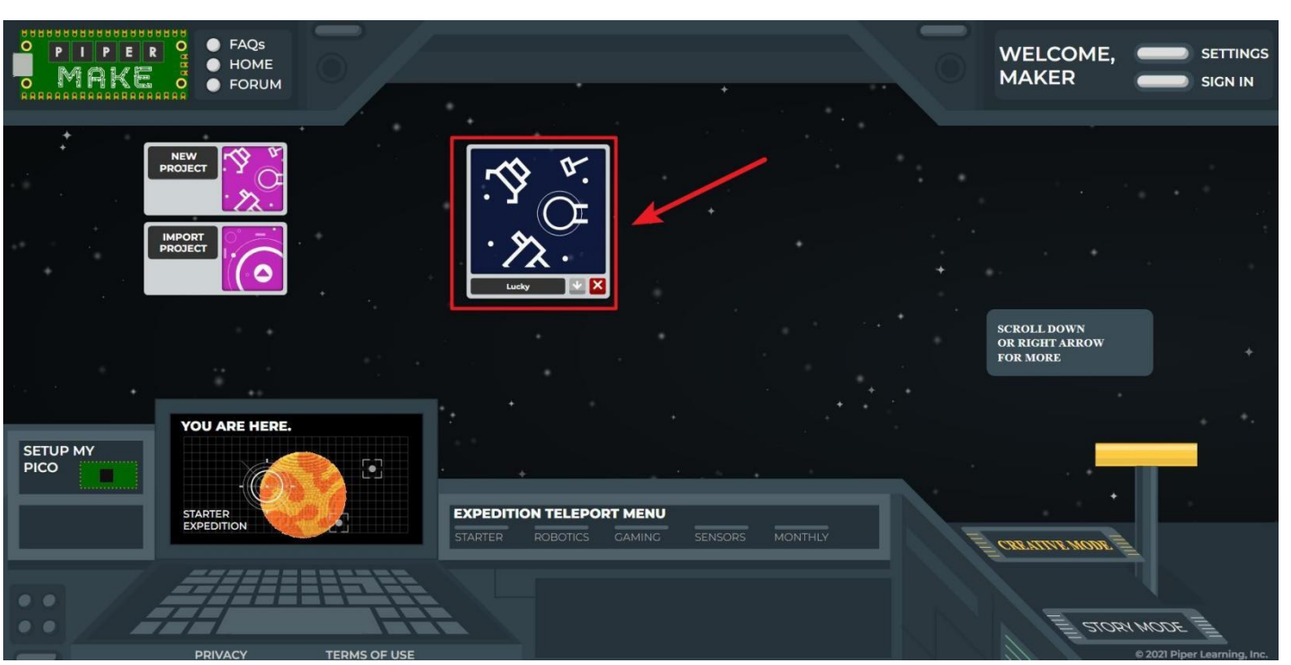
How to use Piper Make(important)

Now that you have set up Pico, it is time to learn how to program it. Now let's light up the onboard LED.

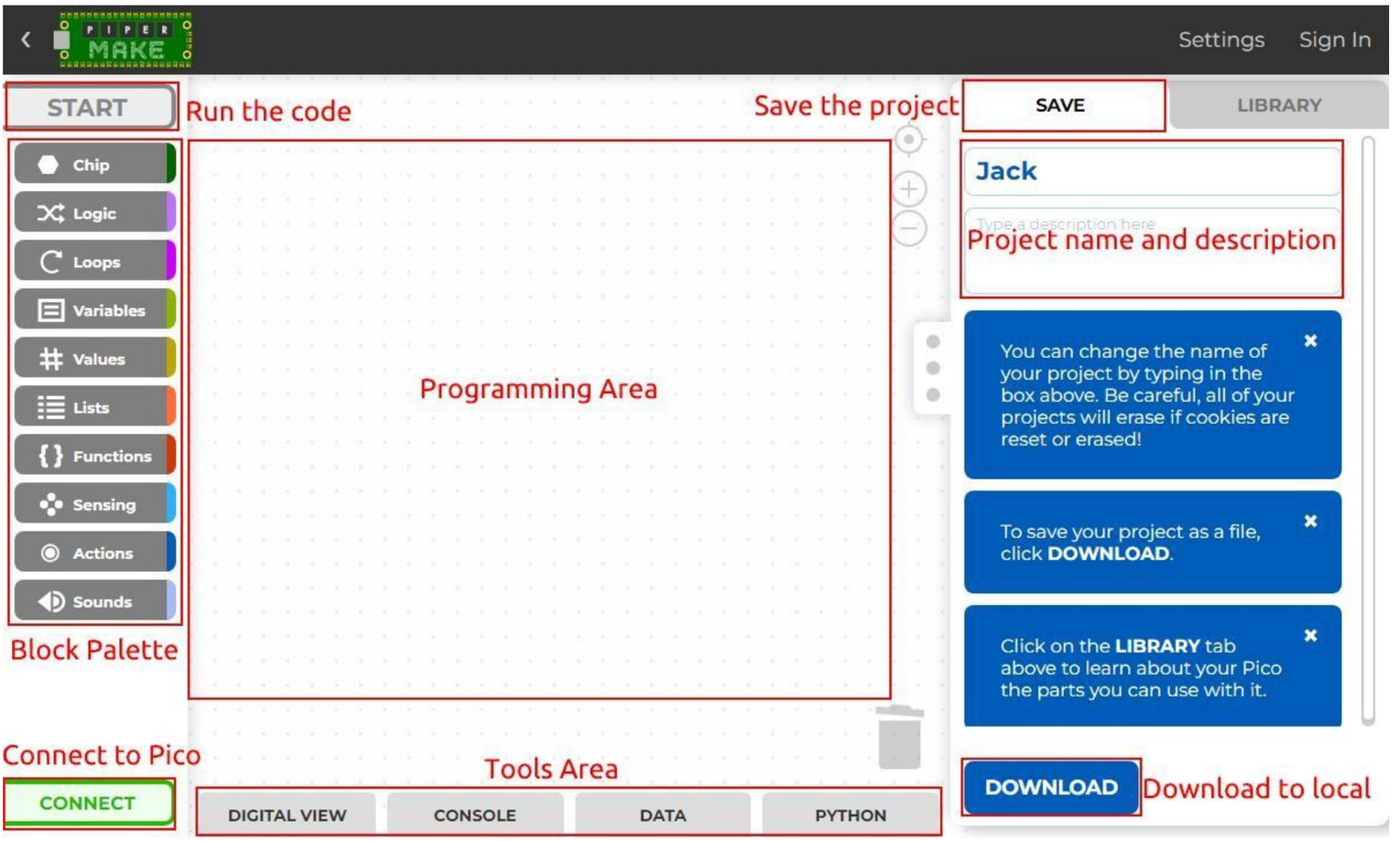
Switch to CREATIVE MODE and click on the **New Project** button, and a new project will appear in the **MY PROJECTS** section and will be assigned a random name that can be changed from the programming page.



Then open the new project just created.

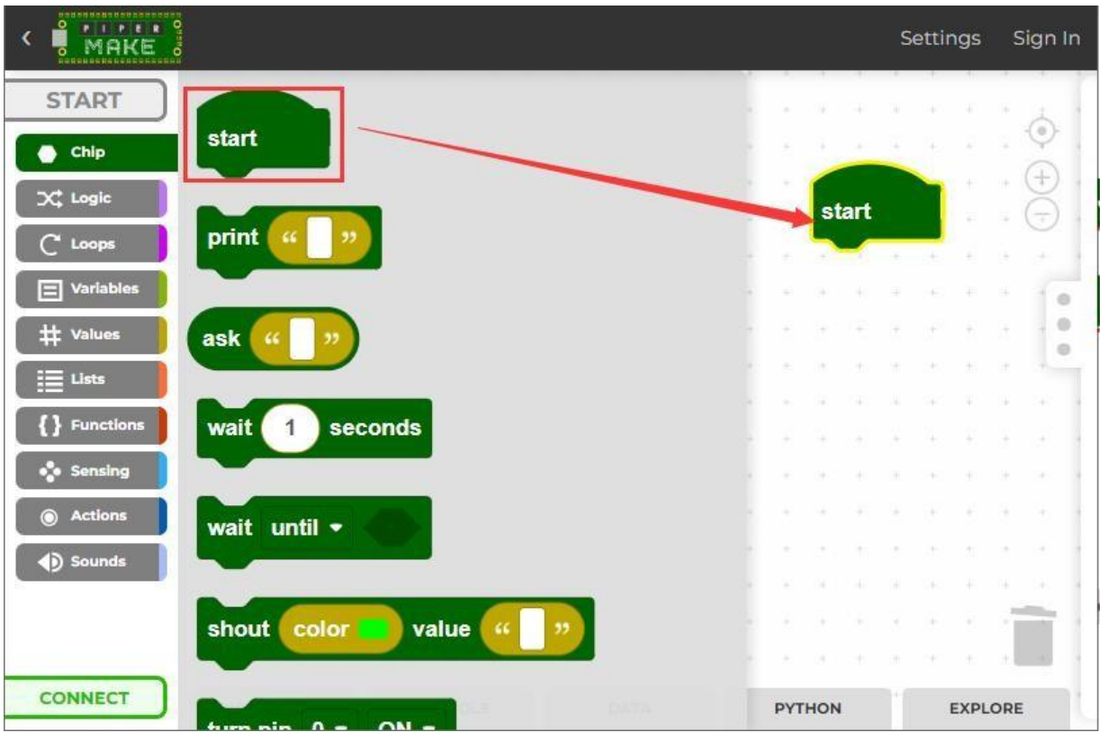


Now go to the Piper Make programming page.

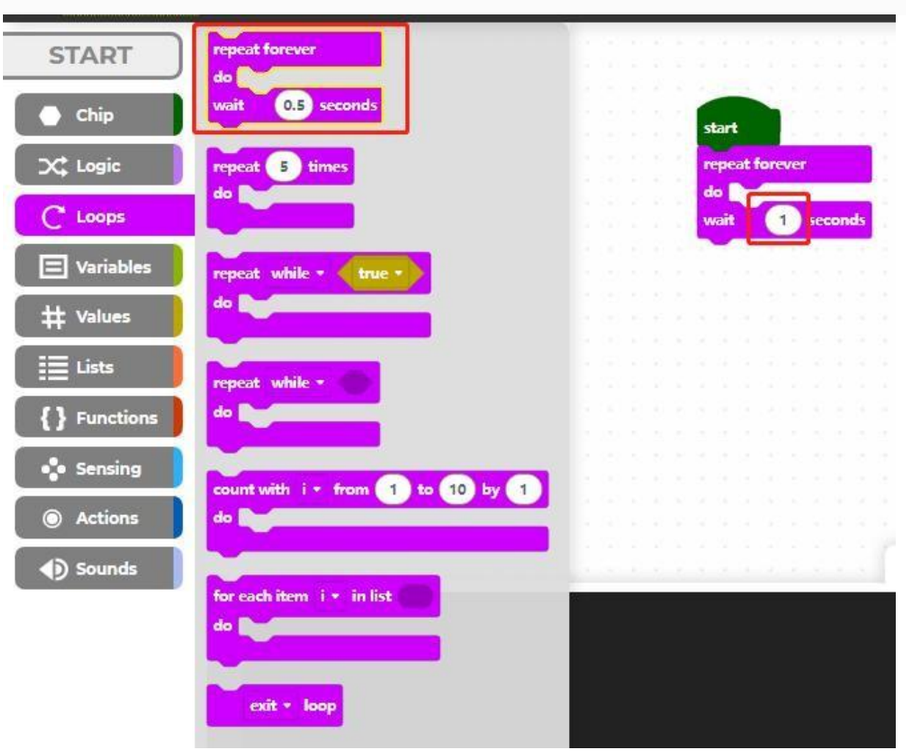


- **START**: Used to run the code, if it's gray, it's not connected to Pico at this time.
- **Block palette**: contains different types of blocks.
- **CONNECT**: Used to connect to Pico, it is green when not connected to Pico, when connected it will become **DISCONNECT(red)**.
- **Programming Area**: Drag blocks here to finish programming by stacking them.
- **Tools Area**: You can click **DIGITAL VIEW** to see the pin distribution of Pico; you can view the print information in **CONSOLE**; you can read data from **DATA**, and you can click **Python** to view the Python source code.
- **Project name and description**: You can change the project name and description.
- **DOWNLOAD**: You can click the **DOWNLOAD** button to save it locally, usually in **.png** format. Next time you can import it via the **Import Project** button on the home page.

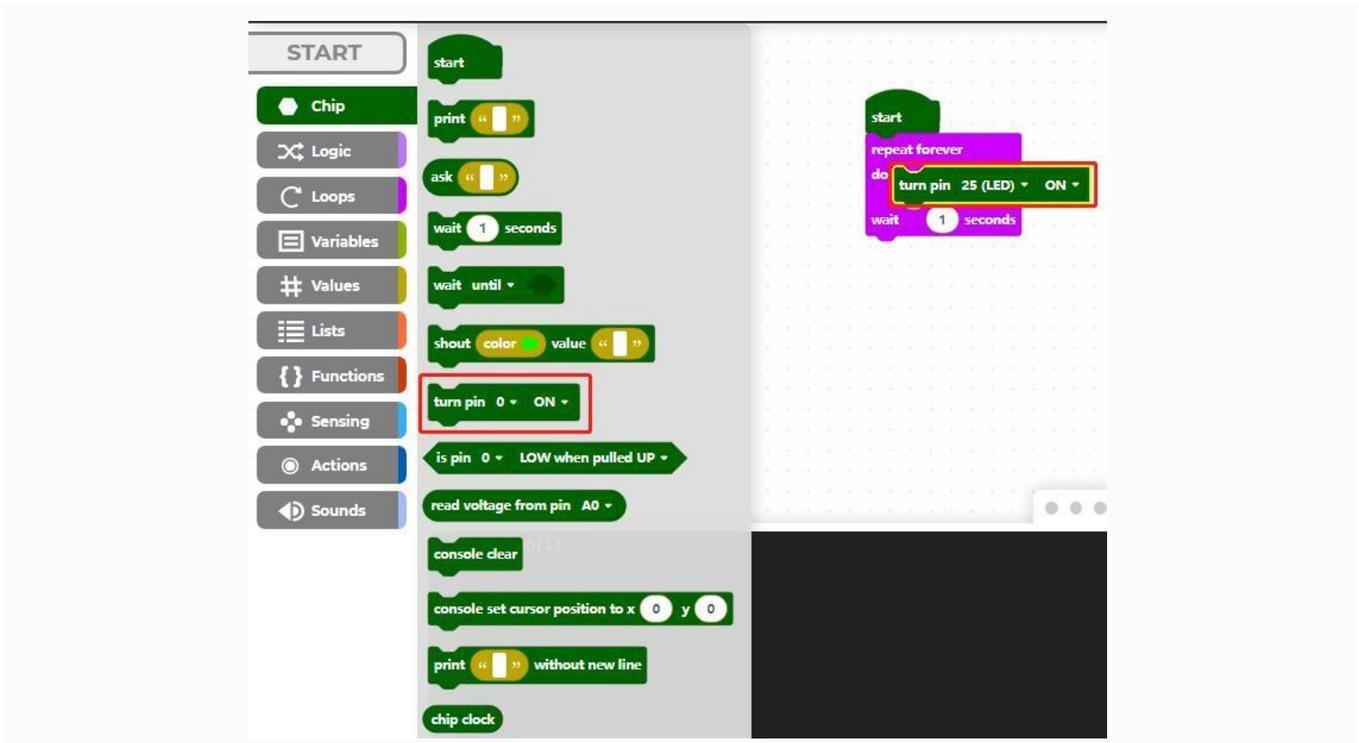
Click on the **Chip** palette and drag the [start] block to the **Programming Area**.



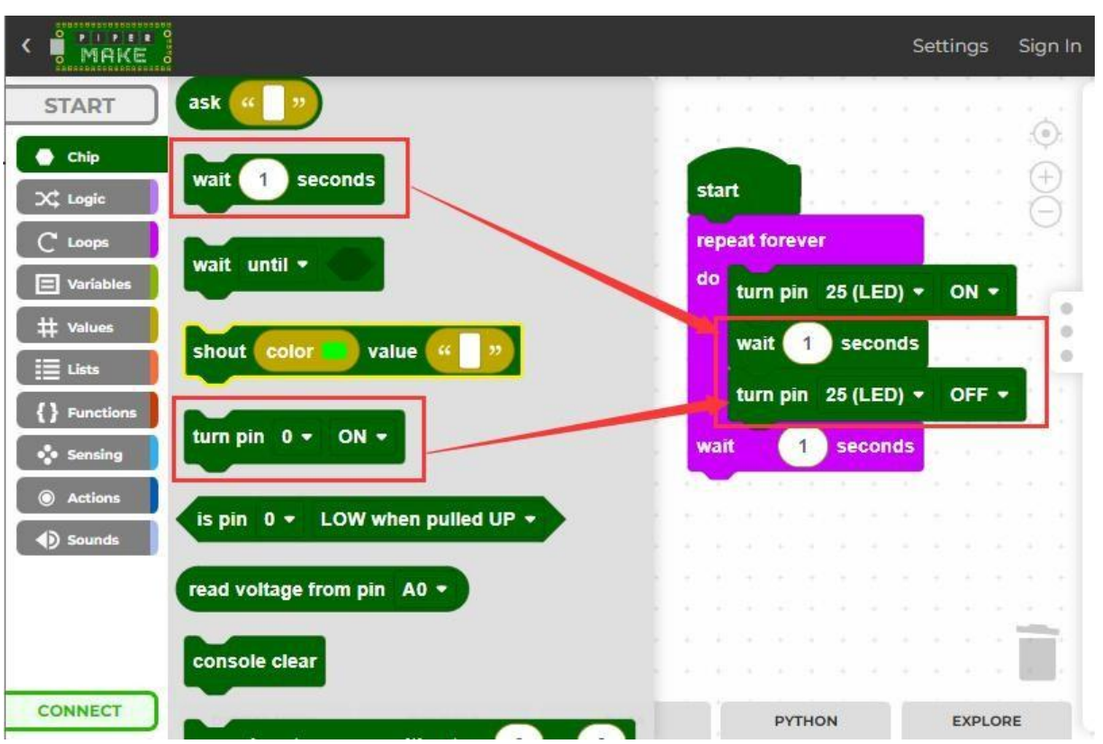
Then drag the [loop] block in **loops** palette to the bottom of the [start] block, and set the loop interval to 1 second.



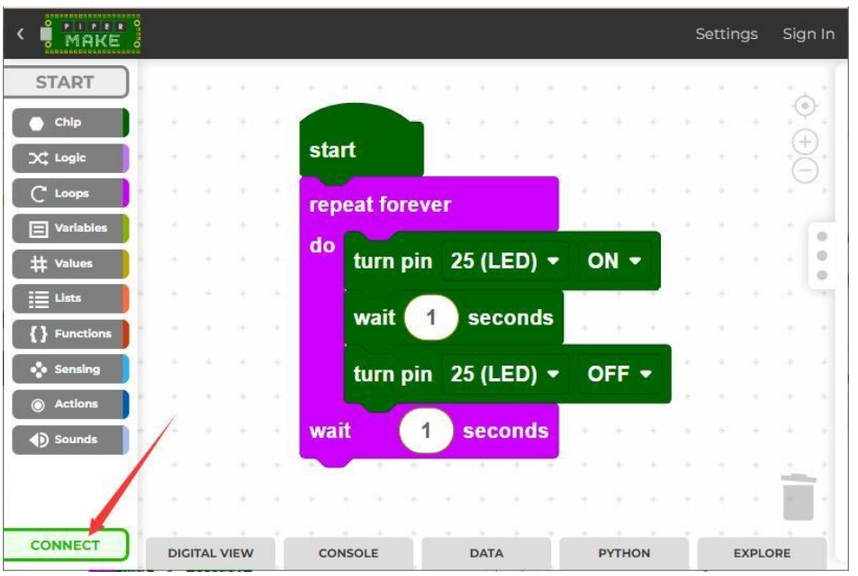
The Raspberry Pi Pico's onboard LED is at pin25, so we use the [turn pin () ON/OFF] block on the **Chip** palette to control it.

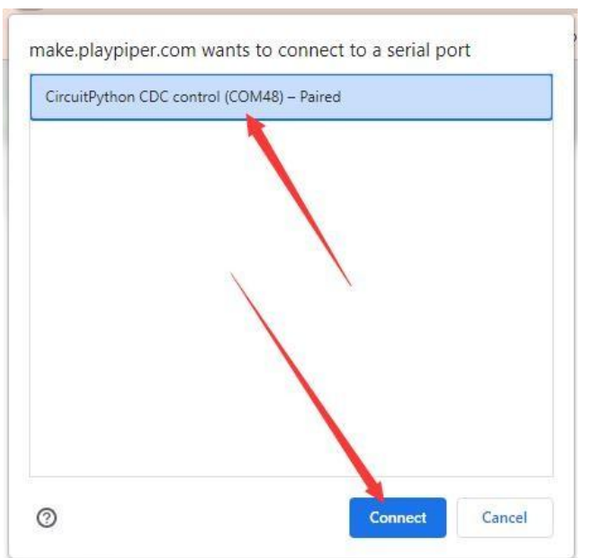


Then turn off **pin25** every second in the cycle.

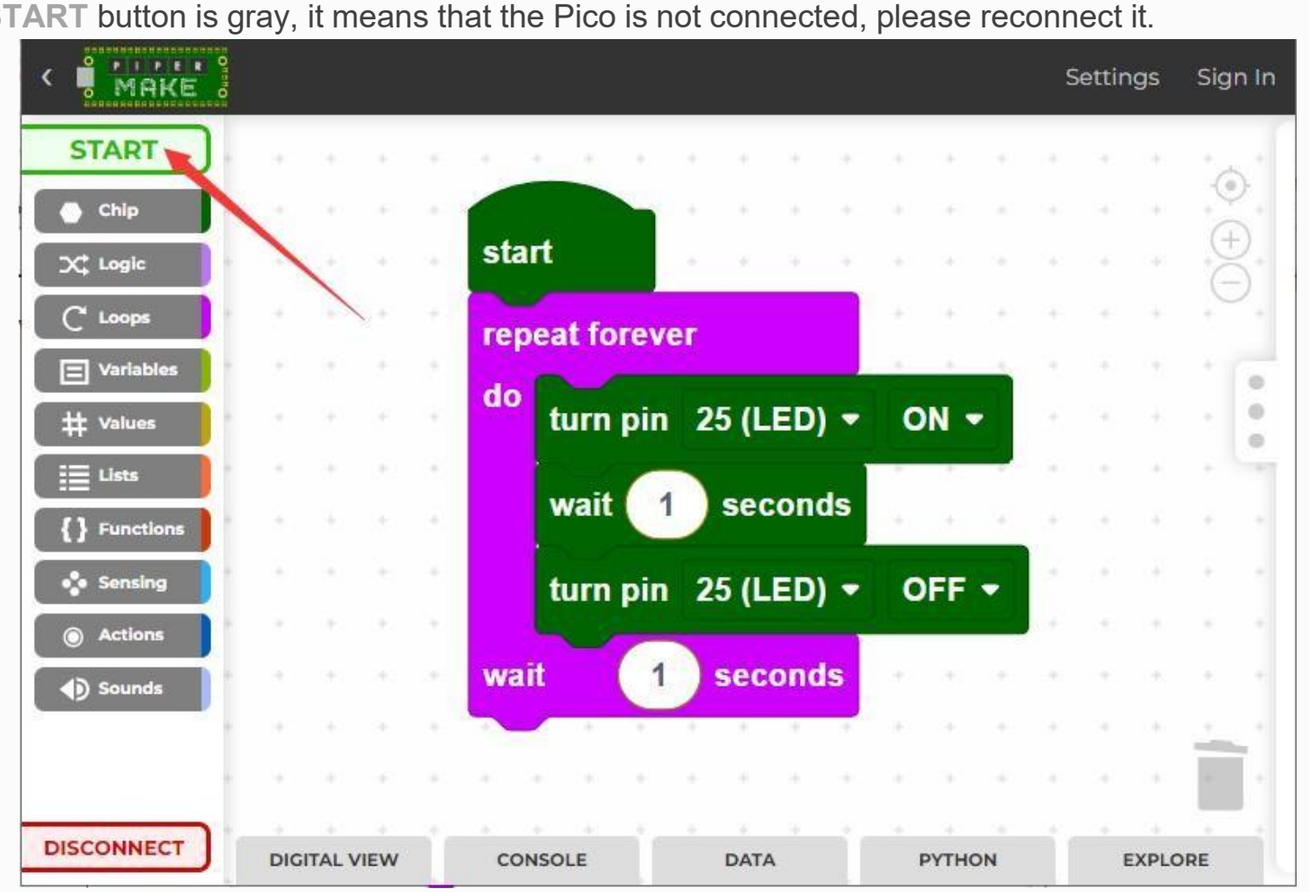


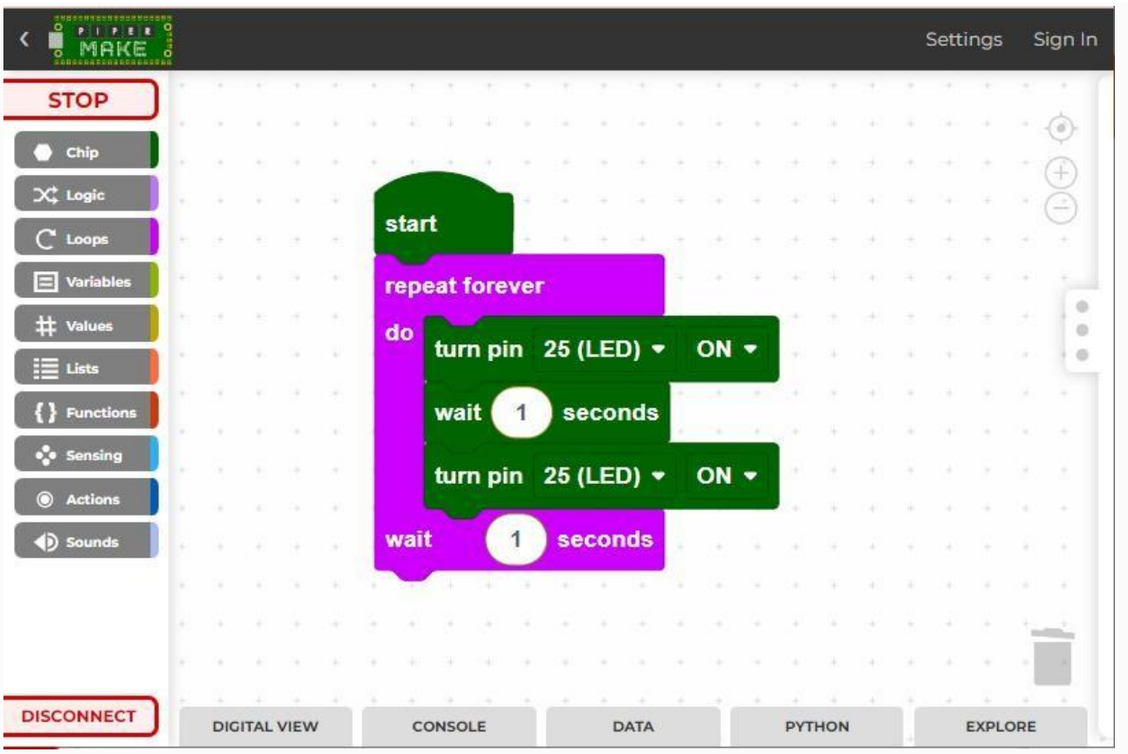
Now click on the **CONNECT** button to connect to pico, after clicking on it a new popup will appear, select the recognized **CircuitPython CDC control (COMXX)** port, then click on **CONNECT**. When the connection is successful, the green **CONNECT** in the bottom left corner will change to a red **DISCONNECT**.[Piper Make - Troubleshooting Guide]

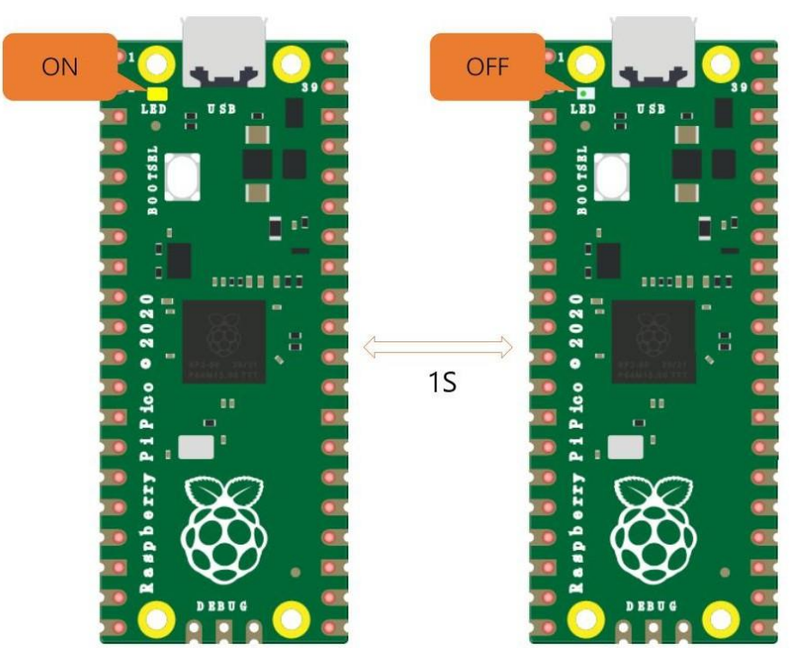




Now click on the **START** button to run this code and you will see the LED on the Pico blink. If START button is gray, it means that the Pico is not connected, please reconnect it.




PIPER MAKE
Settings
Sign In
STOP
Chip
Logic
Loops
Variables
Values
Lists
Functions
Sensing
Actions
Sounds
start
repeat forever
do
turn pin 25 (LED) ON
wait 1 seconds
turn pin 25 (LED) ON
wait 1 seconds
DISCONNECT
DIGITAL VIEW
CONSOLE
DATA
PYTHON
EXPLORE

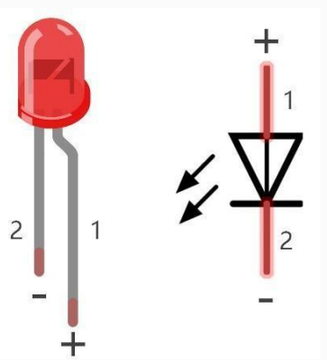

ON
OFF
1S
LED
USB
BOOTSEL
DEBUG
Raspberry Pi Pico © 2020

Project 1 Hello, LED!(important)

Just as printing “Hello, world!” is the first step in learning programming, letting the LED light up is the traditional entry to learning physical programming.Knowledge point: **digital signal output**.

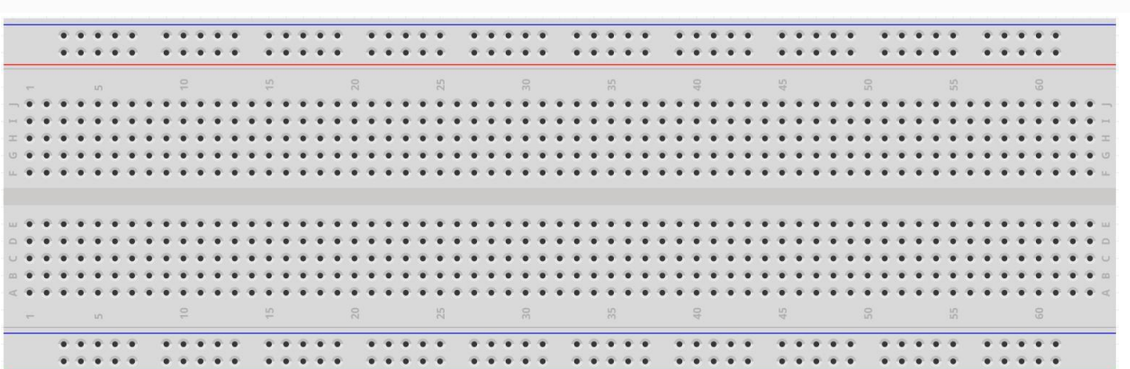
LED

An LED is a type of diode. All diodes only work if current is flowing in the correct direction and have two Poles.



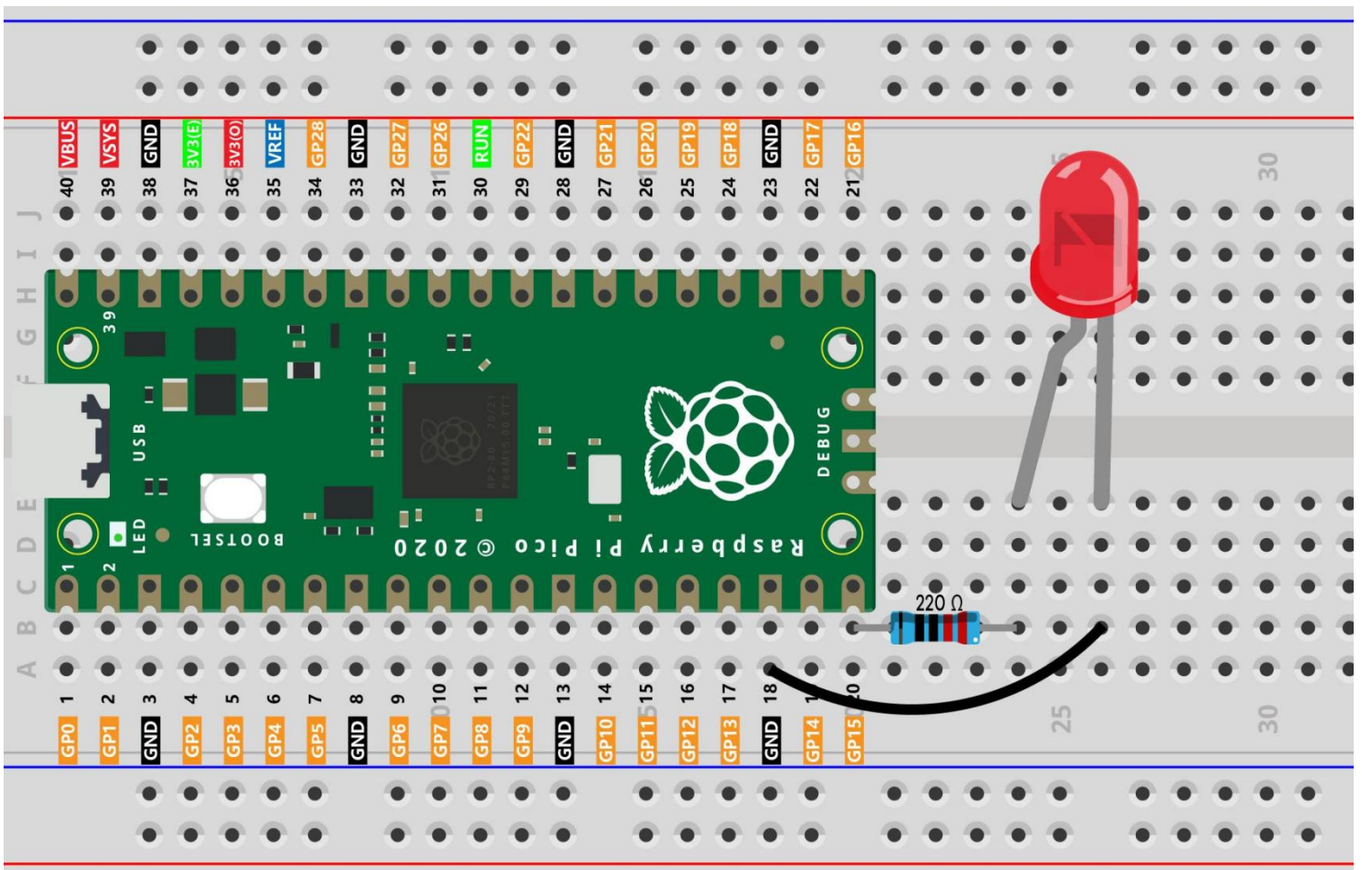
Breadboard

The breadboard is a rectangular plastic plate with a bunch of small holes in it. These holes allow us to easily insert electronic components and build electronic circuits. The breadboard does not permanently fix the electronic components, which makes it easy for us to repair the circuit and start over when we make a mistake.



Tip:Learn more about LED breadboard

Wiring

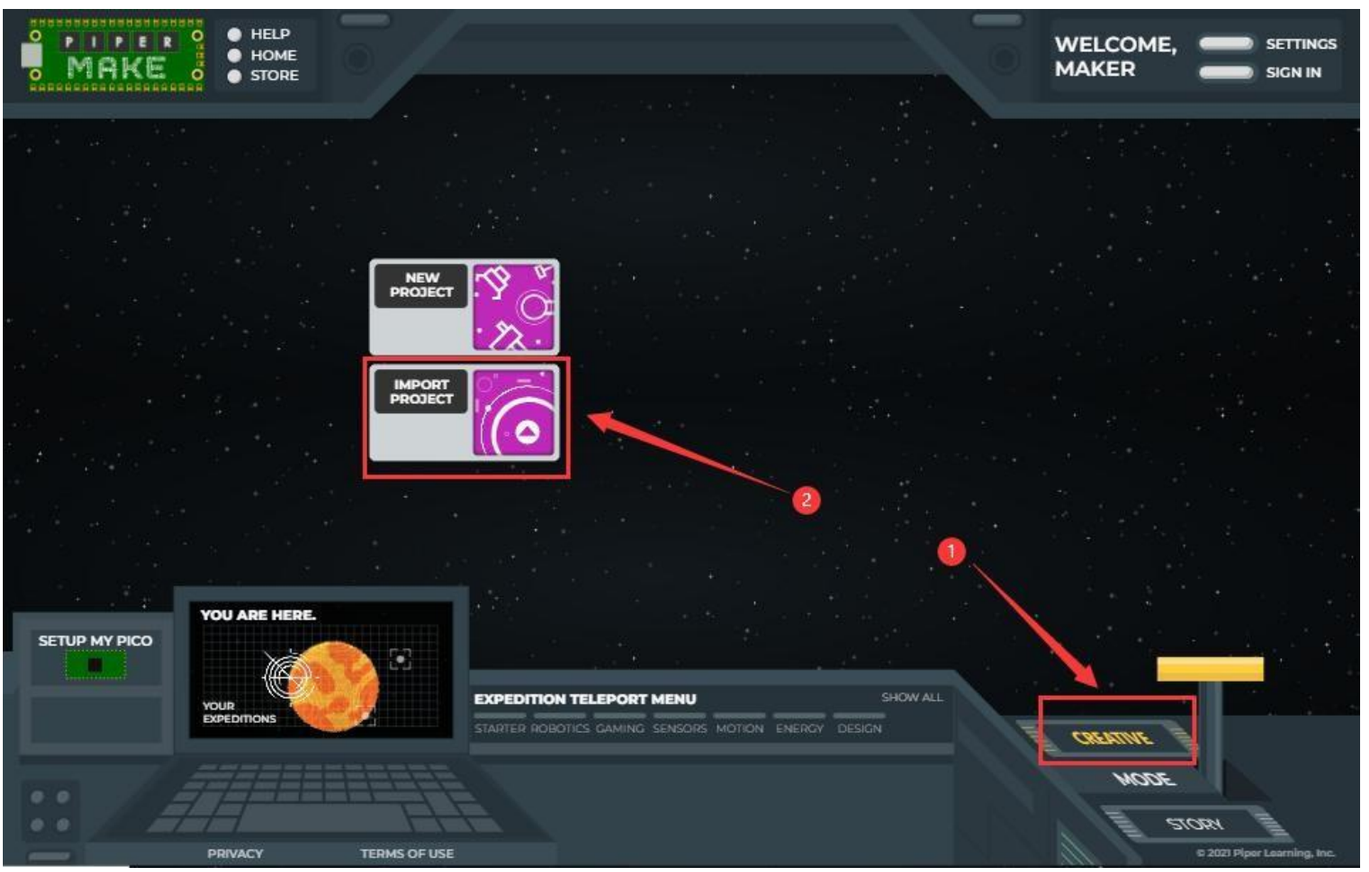

VBUS
VSYS
GND
3V3(E)
3V3(O)
VREF
GP28
GND
GP27
GP26
RUN
GP22
GND
GP21
GP20
GP19
GP18
GND
GP17
GP16
USB
LED
BOOTSEL
DEBUG
Raspberry Pi Pico ©2020
220 Ω
GP0
GP1
GND
GP2
GP3
GP4
GP5
GND
GP6
GP7
GP8
GP9
GND
GP10
GP11
GP12
GP13
GND
GP14
GP15

Code

● Import project code

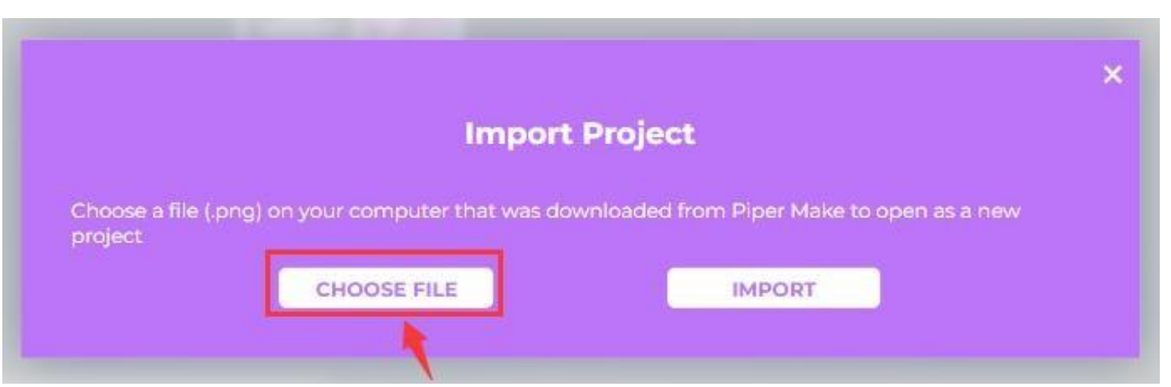
First,visit Piper Make through the following link：https://make.playpiper.com/

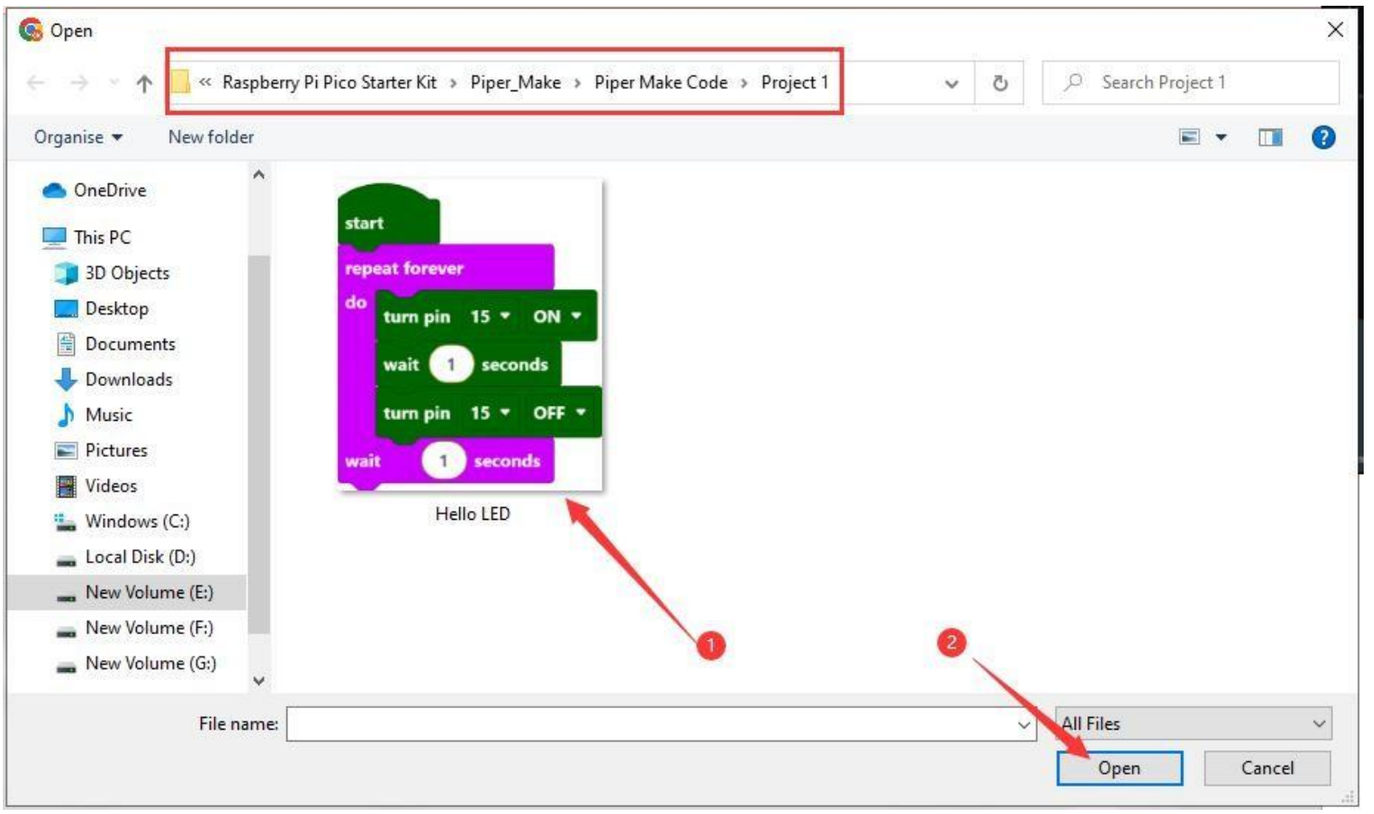
Switch to CREATIVE MODE and click on the ”**IMPORT PROJECT**” button,

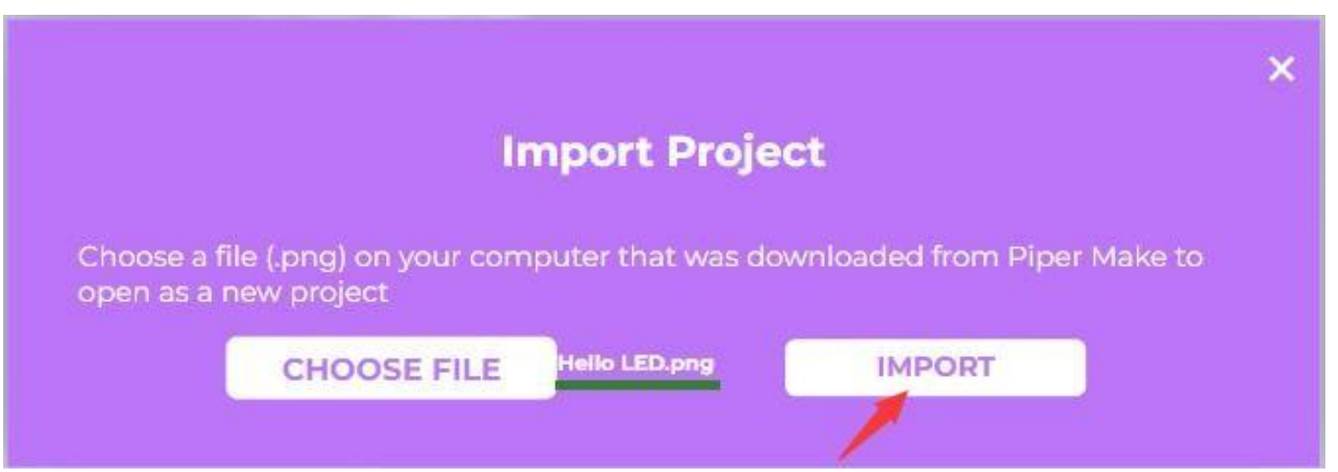


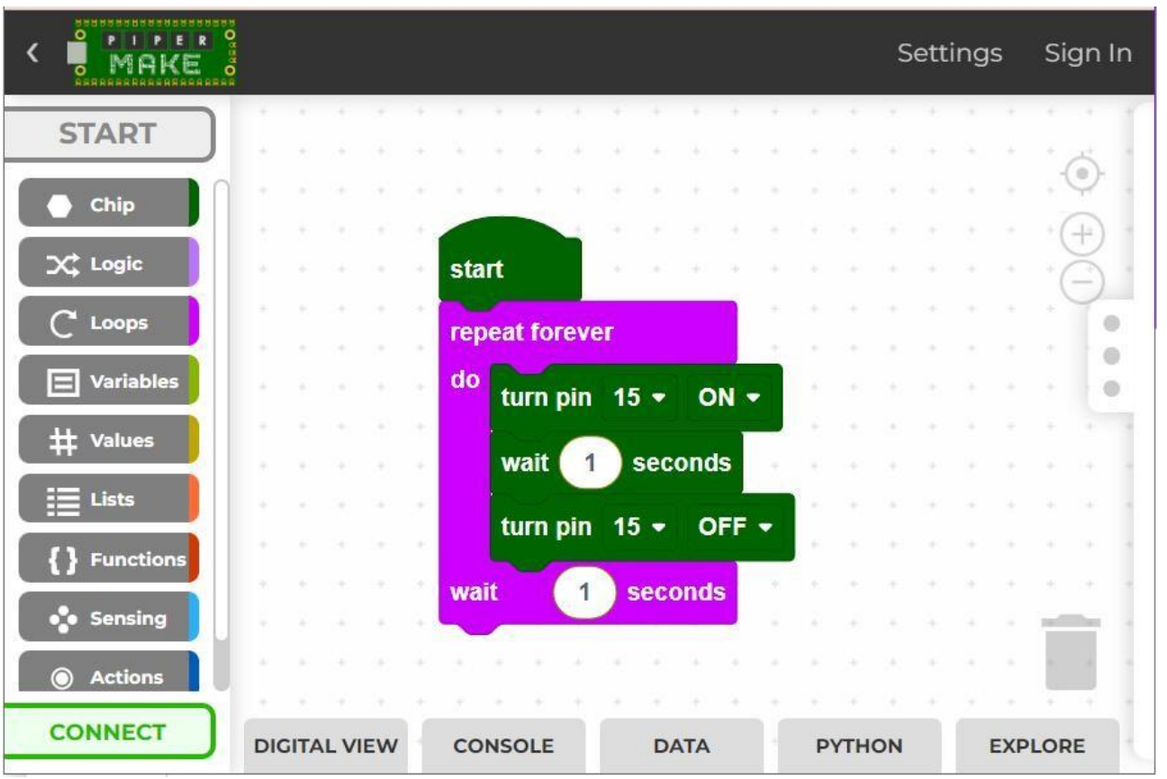
Choose **Hello_LED.png** file and then click “**IMPORT**”

File path :Raspberry Pi Pico Starter Kit\Piper_Make\Piper Make Code\Project 1



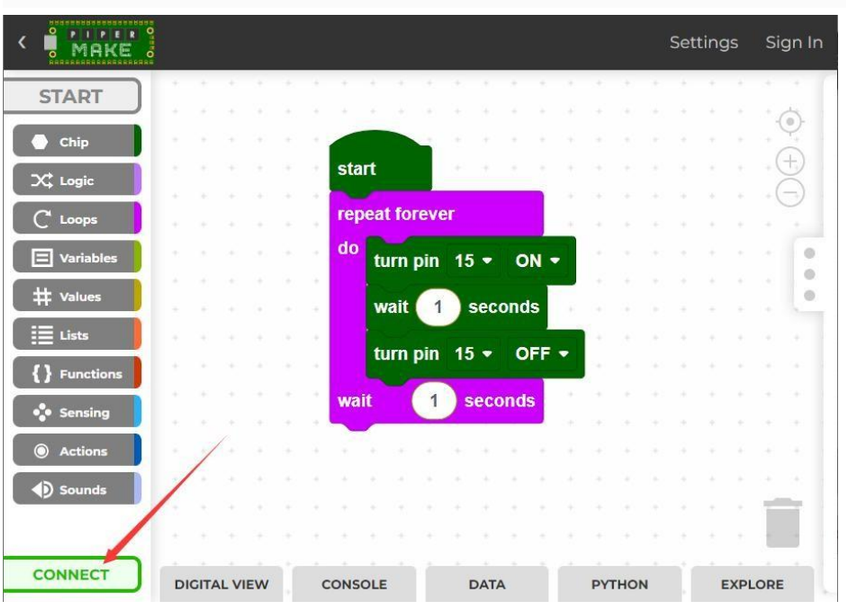

Open
Raspberry Pi Pico Starter Kit › Piper_Make › Piper Make Code › Project 1
Search Project 1
Organise
New folder
OneDrive
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Windows (C:)
Local Disk (D:)
New Volume (E:)
New Volume (F:)
New Volume (G:)
start
repeat forever
do
turn pin 15 ON
wait 1 seconds
turn pin 15 OFF
wait 1 seconds
Hello LED
1
2
File name:
All Files
Open
Cancel

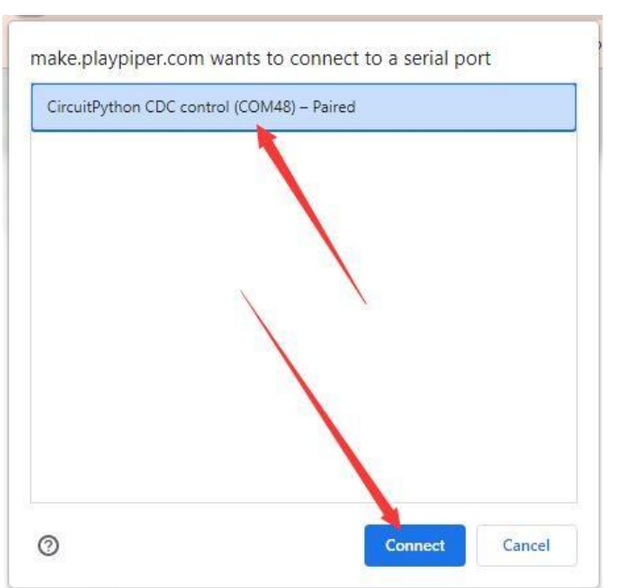

Import Project
Choose a file (.png) on your computer that was downloaded from Piper Make to open as a new project
CHOOSE FILE
Hello LED.png
IMPORT



● Piper Make connect to Raspberry Pi Pico serial port.

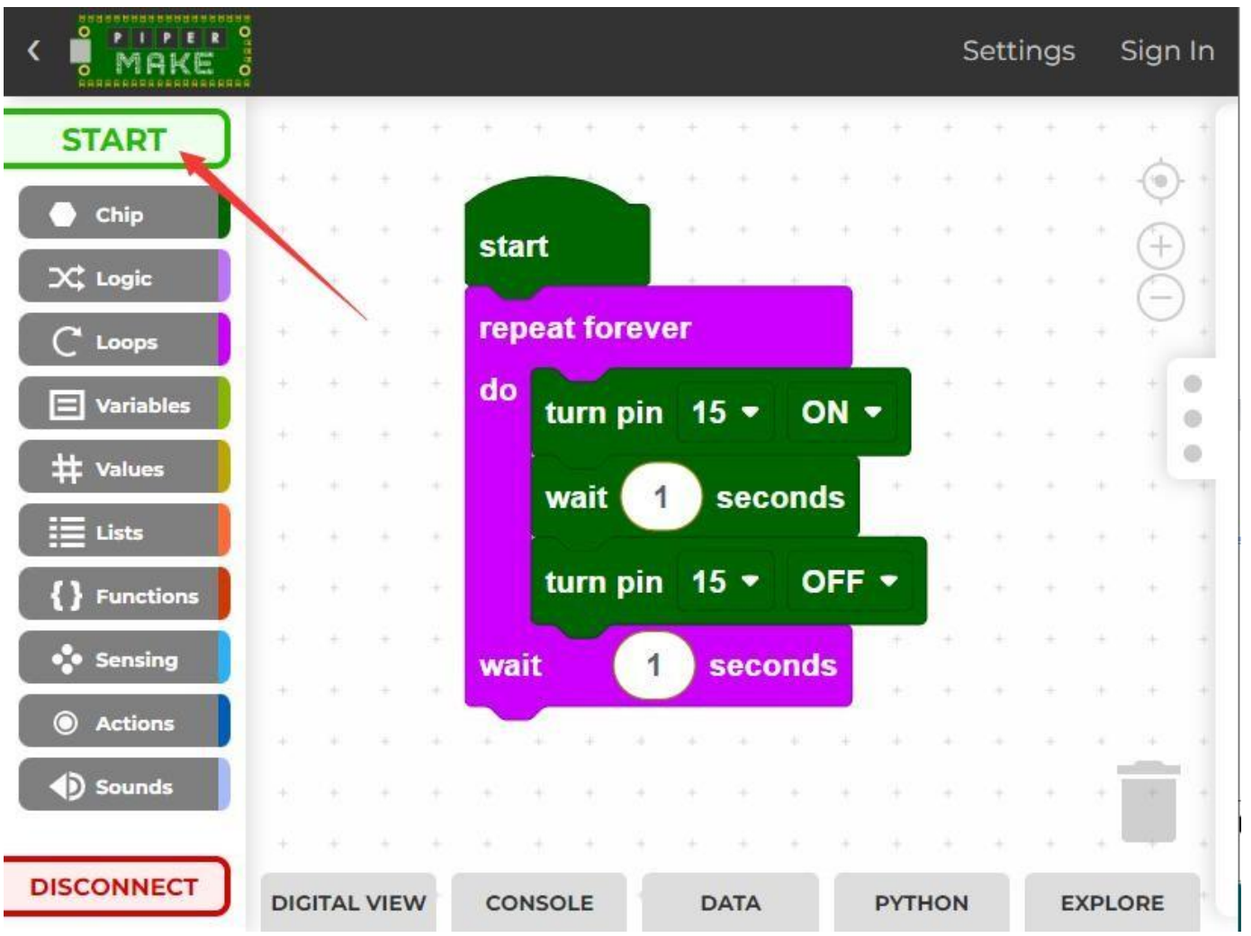
Now click on the **CONNECT** button to connect to pico, after clicking on it a new popup will appear, select the recognized **CircuitPython CDC control (COMXX)** port, then click on **CONNECT**. When the connection is successful, the green **CONNECT** in the bottom left corner will change to a red **DISCONNECT**.Piper Make - Troubleshooting Guide FAQ

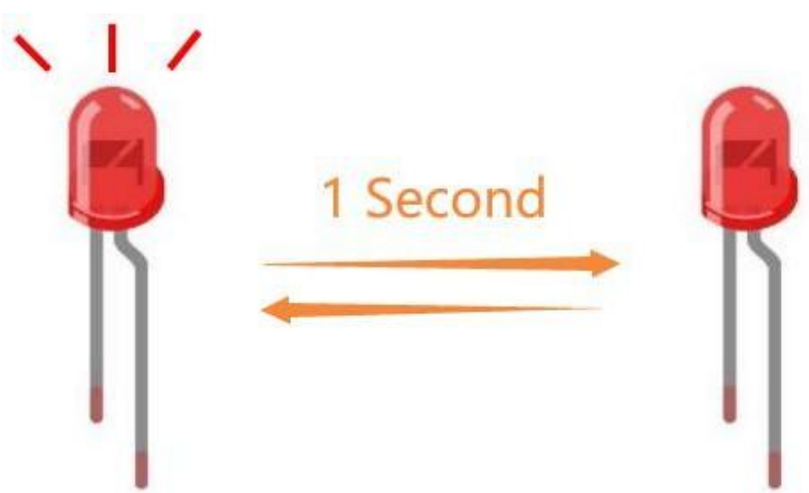




Run the code

Now click on the **START** button to run this code and you will see the LED in the circuit starts Blink. If START button is gray, it means that the Pico is not connected, please reconnect it.

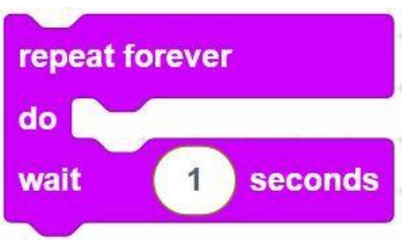




Code Explanation

This is the body of the loop: turn the pin15 on to light up the LED, wait one second, then turn the pin15 off to make the LED go off. Wait 1 second and then re-run the previous cycle, so you can see the LED has been in the state of alternating between light and off.

- [start]: This block is the basic framework of the program and represents the beginning of the program.



- [repeat forever do() wait()seconds]: Means that the blocks in it will be executed repeatedly, and the execution time interval is defined by yourself.

- [turn pin () ON/OFF]: Indicates that a certain pin is placed in a high level (ON) or a low level (OFF).

- [wait () seconds]: Set the execution interval between blocks.

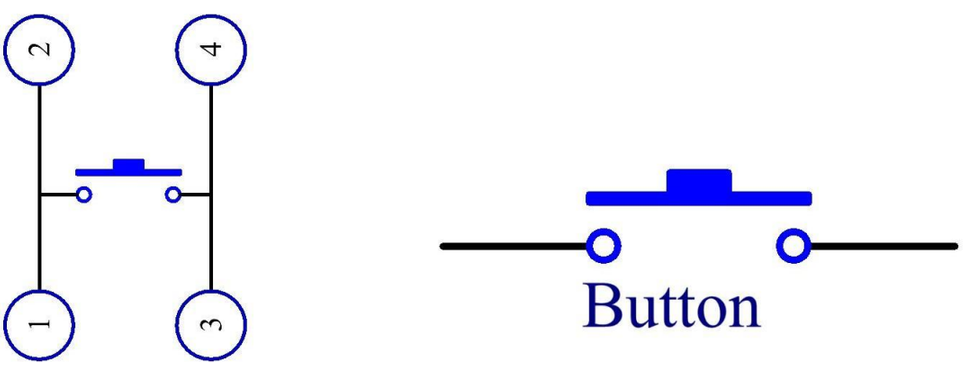
Project 2 Button Control LED

In the project, we will control the LED state through a Push Button Switch. When the button is pressed, our LED will turn ON, and when it is released, the LED will turn OFF. This describes a Momentary Switch.Knowledge point: **digital signal input**.

Component knowledge

Push button

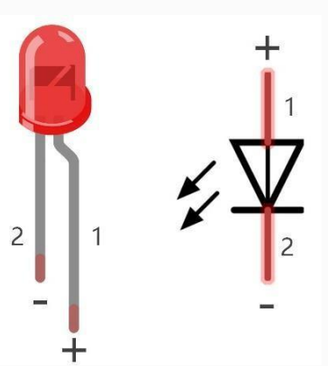
This type of Push Button Switch has 4 pins (2 Pole Switch). Two pins on the left are connected, and both left and right sides are the same per the illustration:



When the button on the switch is pressed, the circuit is completed (your project is Powered ON).

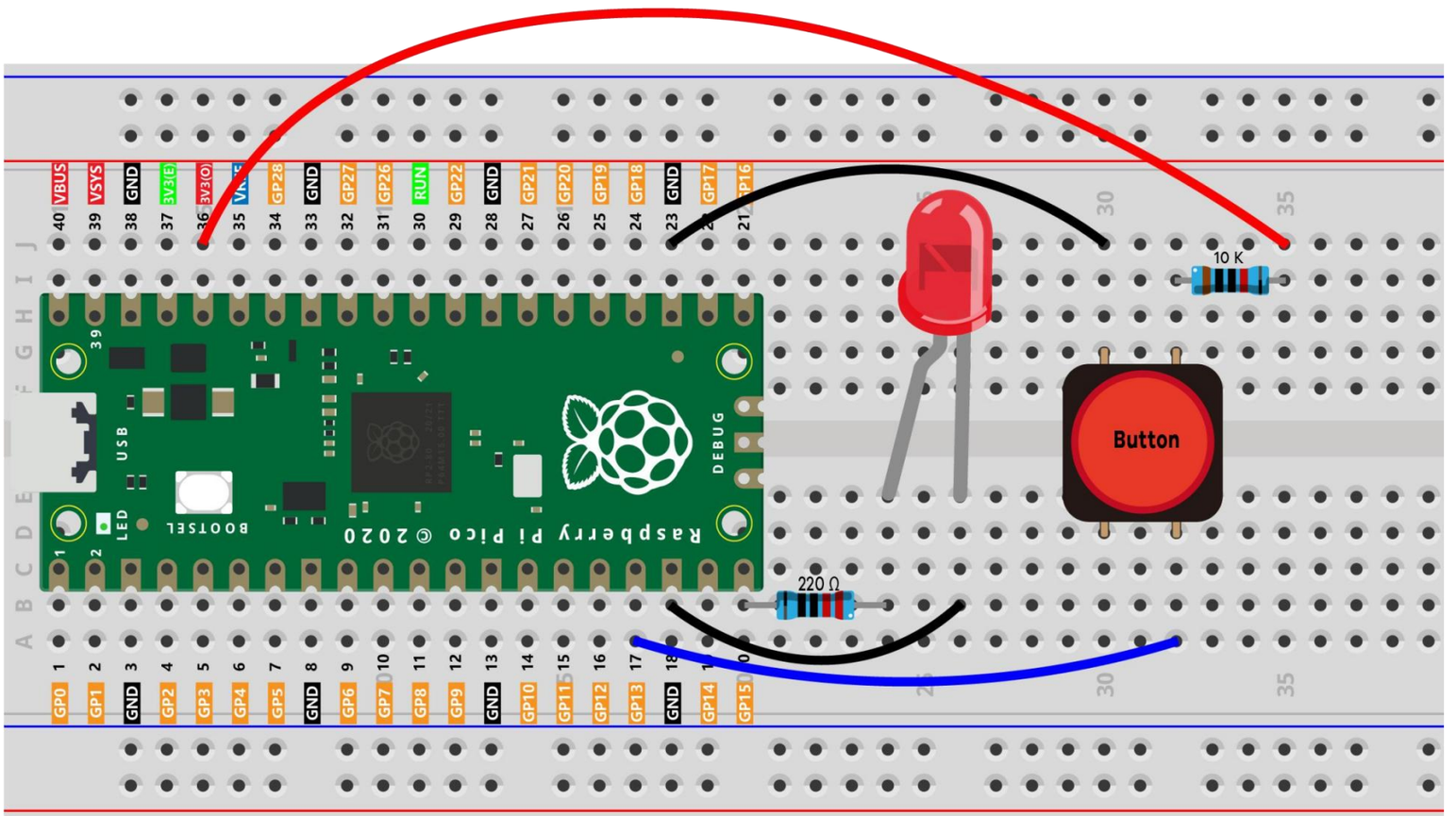
LED

An LED is a type of diode. All diodes only work if current is flowing in the correct direction and have two Poles.



Tip:Learn more about Button LED

Wiring



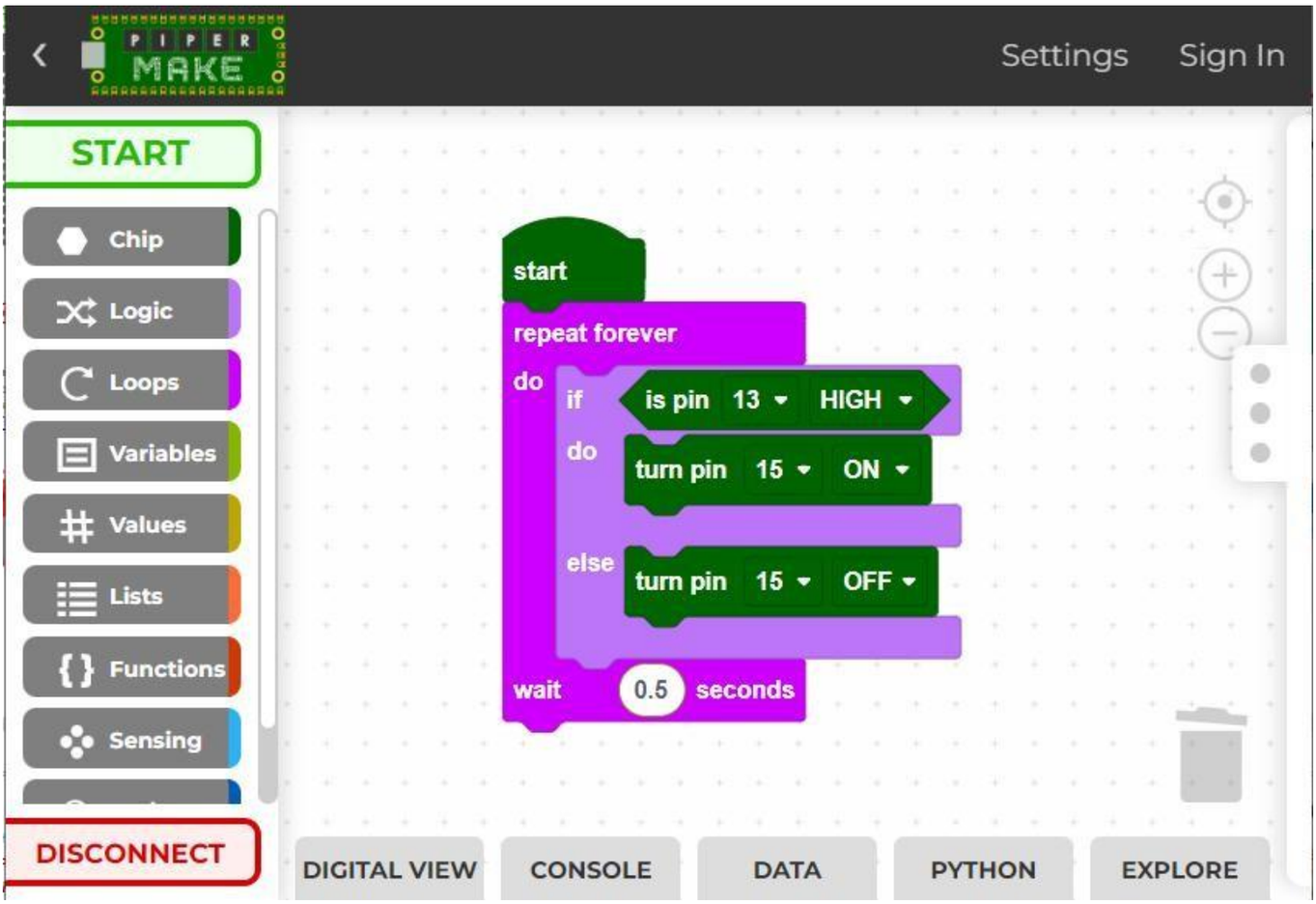
Code

Visit link： https://make.playpiper.com/. Switch to "**CREATIVE MODE**" and click on the "**IMPORT PROJECT**" button to import **Button_Control_LED.png file.**

File path :Raspberry Pi Pico Starter Kit\Piper_Make\Piper Make Code\Project 2

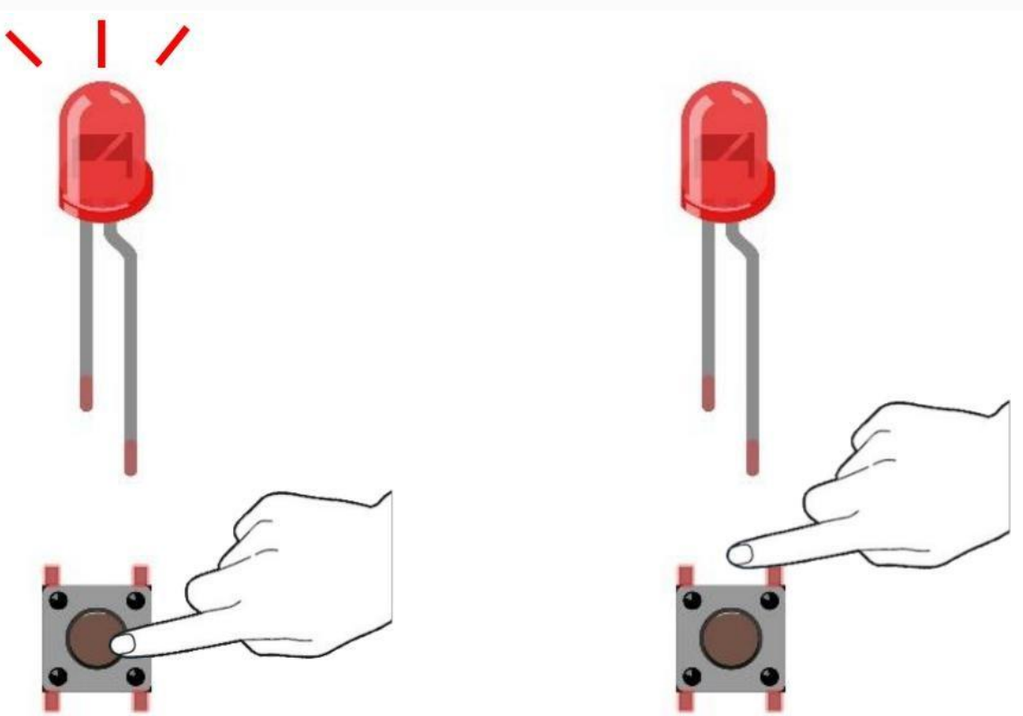
Note:

Don't forget to connect to Raspberry Pi Pico serial port——**CircuitPython CDC control (COMXX)** .Have a Question?



Now click on the **START** button to run this code.When pressing the button, LED lights up; when releasing the button, LED lights OFF. If **START** button is gray, it means that the Pico is not connected, please reconnect it.

[Piper Make - Troubleshooting Guide]



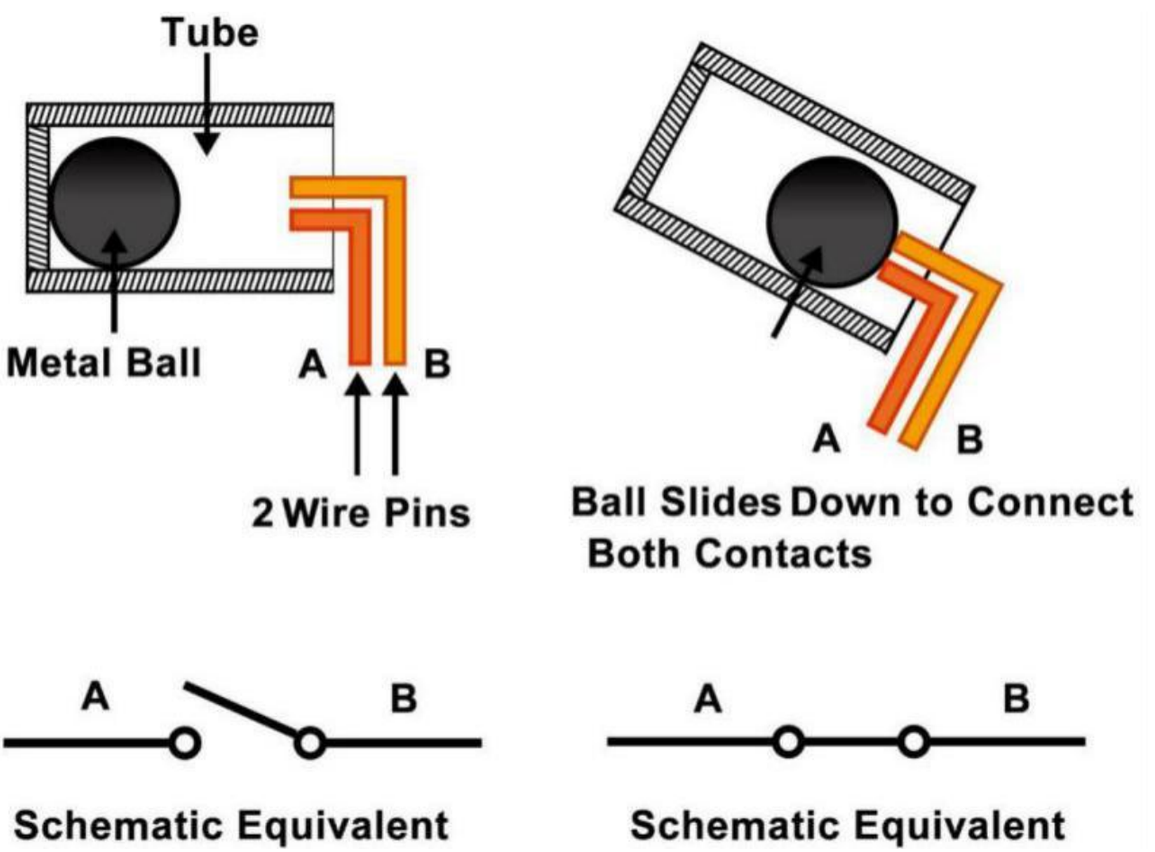
Project 3 Tilt Switch Control LED

In this project, learn how to use the tilt switch to control the LED.When the tilt switch is placed horizontally, the LED will turn on; and when placed vertically, the LED will turn off.

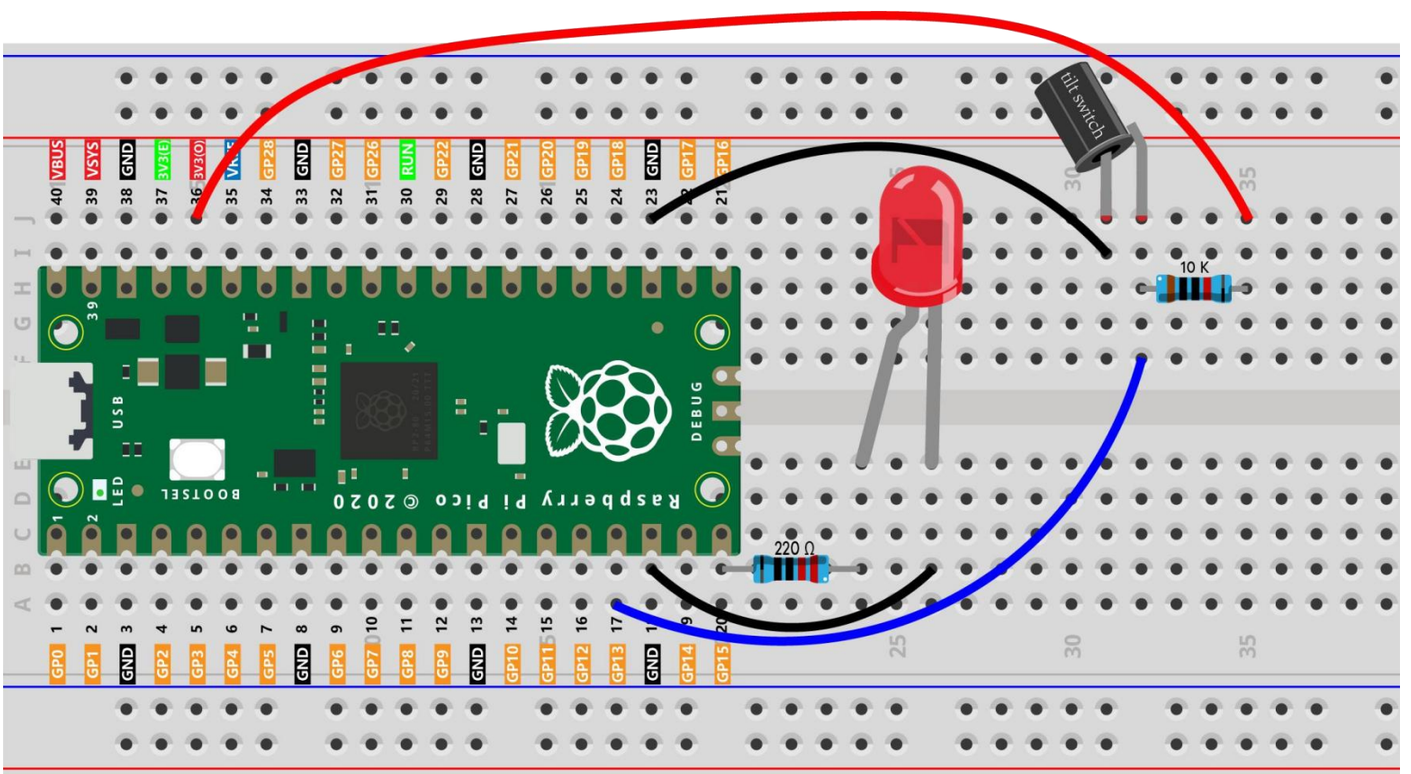
Tilt Switch

This is a ball tilt-switch with a metal ball inside. It is used to detect inclinations of a small angle.

The principle is very simple. When the switch is tilted in a certain angle, the ball inside rolls down and touches the two contacts connected to the pins outside, thus triggering circuits. Otherwise the ball will stay away from the contacts, thus breaking the circuits.



Wiring



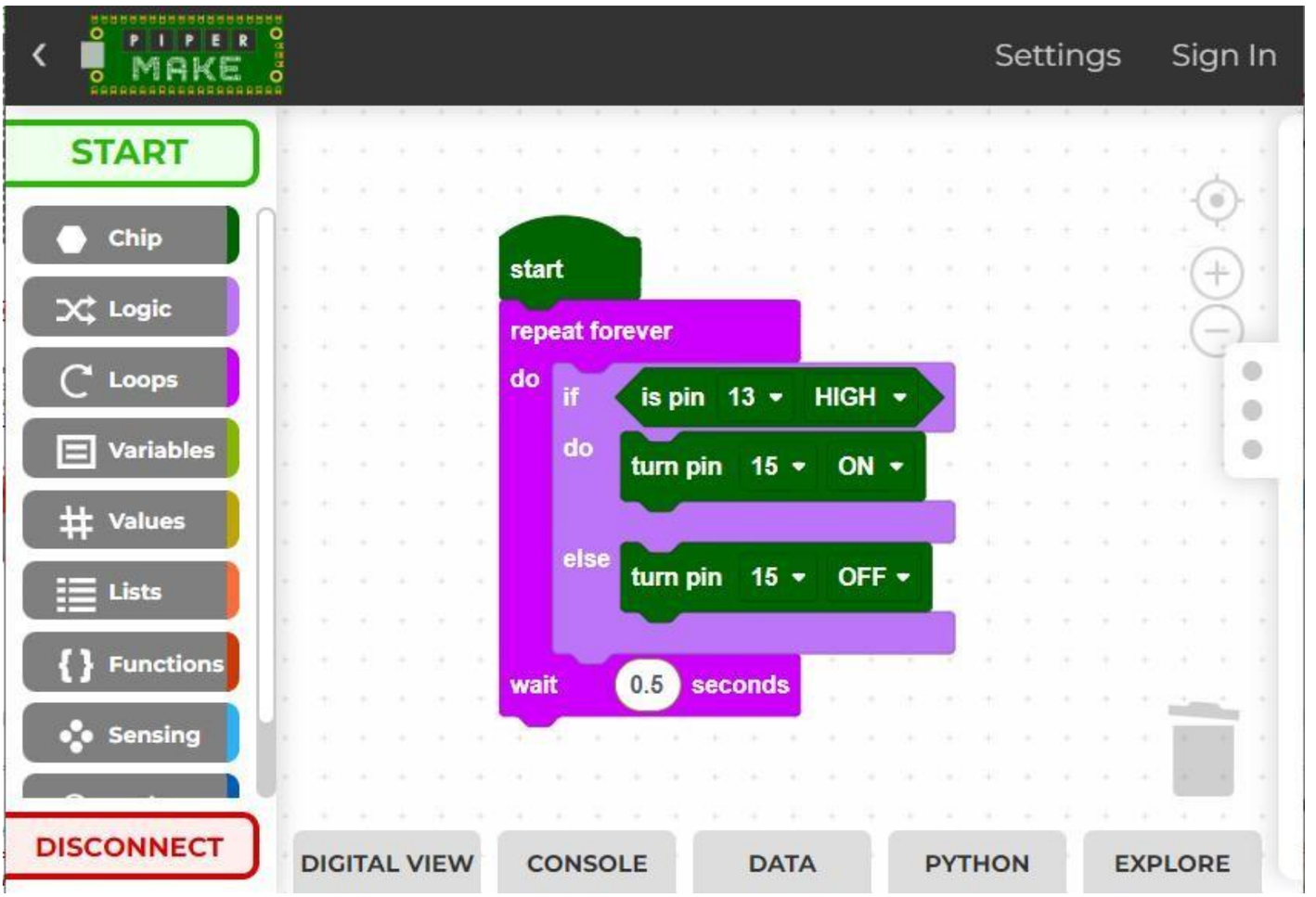
Code

Visit link： https://make.playpiper.com/. Switch to "**CREATIVE MODE**" and click on the "**IMPORT PROJECT**" button to import **Tilt_Switch_Control_LED.png file.**

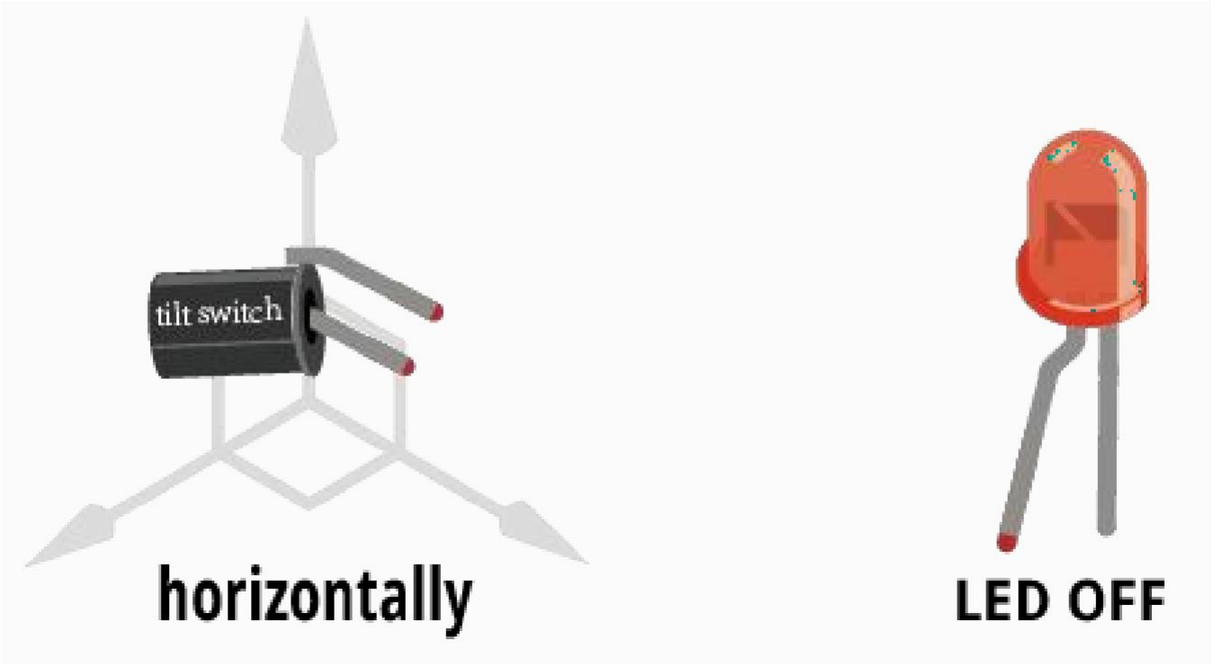
File path :Raspberry Pi Pico Starter Kit\Piper_Make\Piper Make Code\Project 3

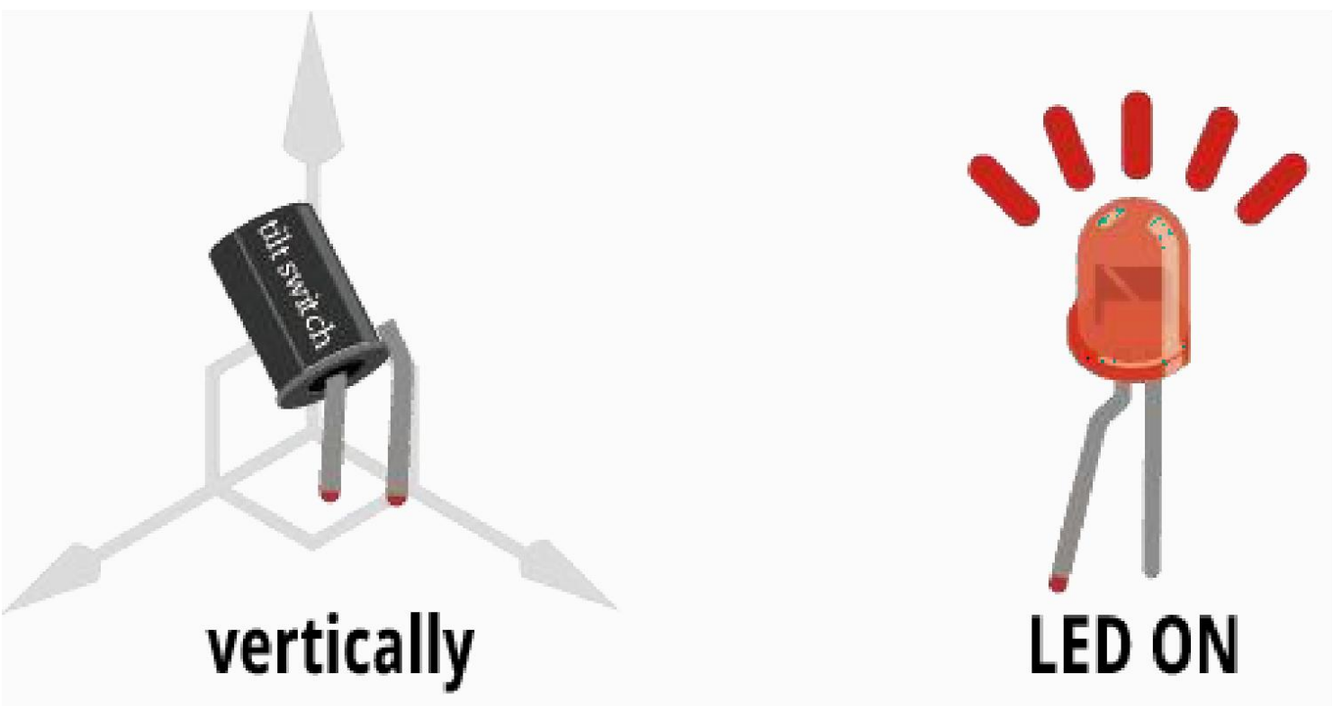
Note:

Don't forget to connect to Raspberry Pi Pico serial port——**CircuitPython CDC control (COMXX)** .Have a Question?



Now click on the **START** button to run this code.When the tilt switch is placed horizontally, the LED will turn off; and when placed vertically, the LED will turn on.. If **START** button is gray, it means that the Pico is not connected, please reconnect it.
[Piper Make - Troubleshooting Guide]

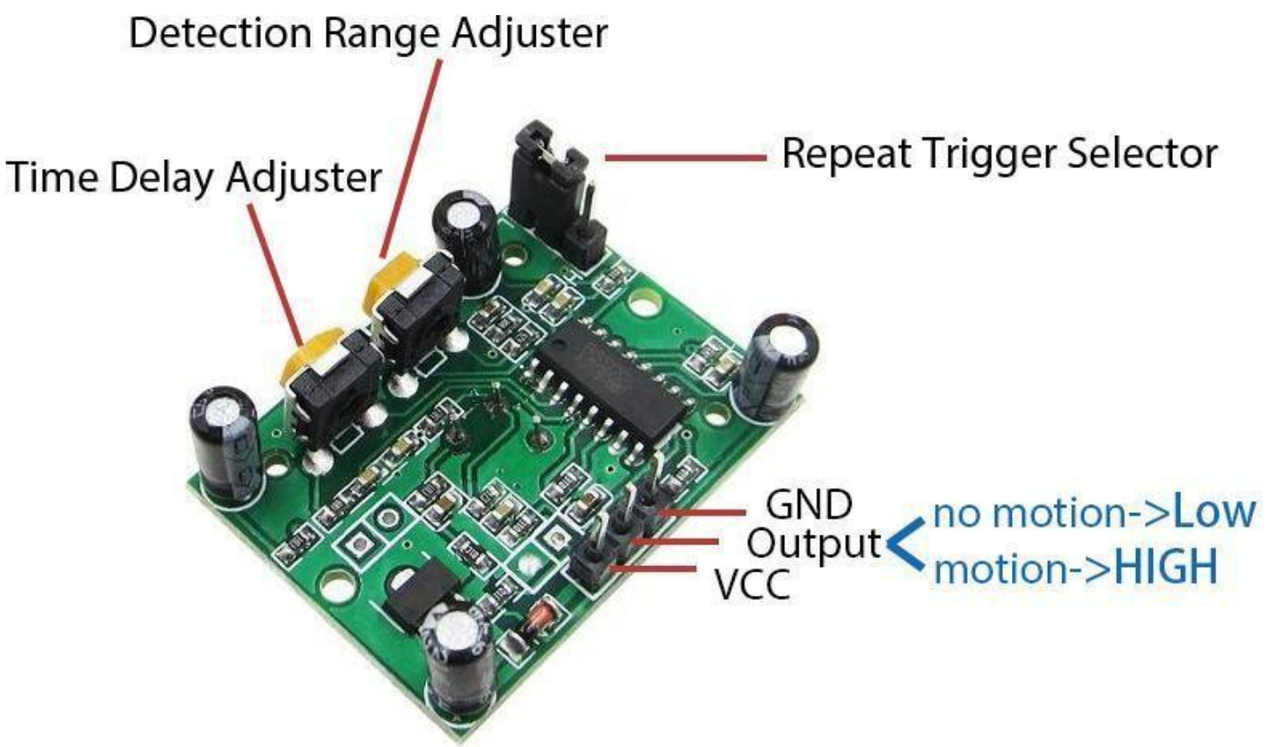

tilt switch
horizontally
LED OFF


tilt switch
vertically
LED ON

Project 4 Intruder Alarm

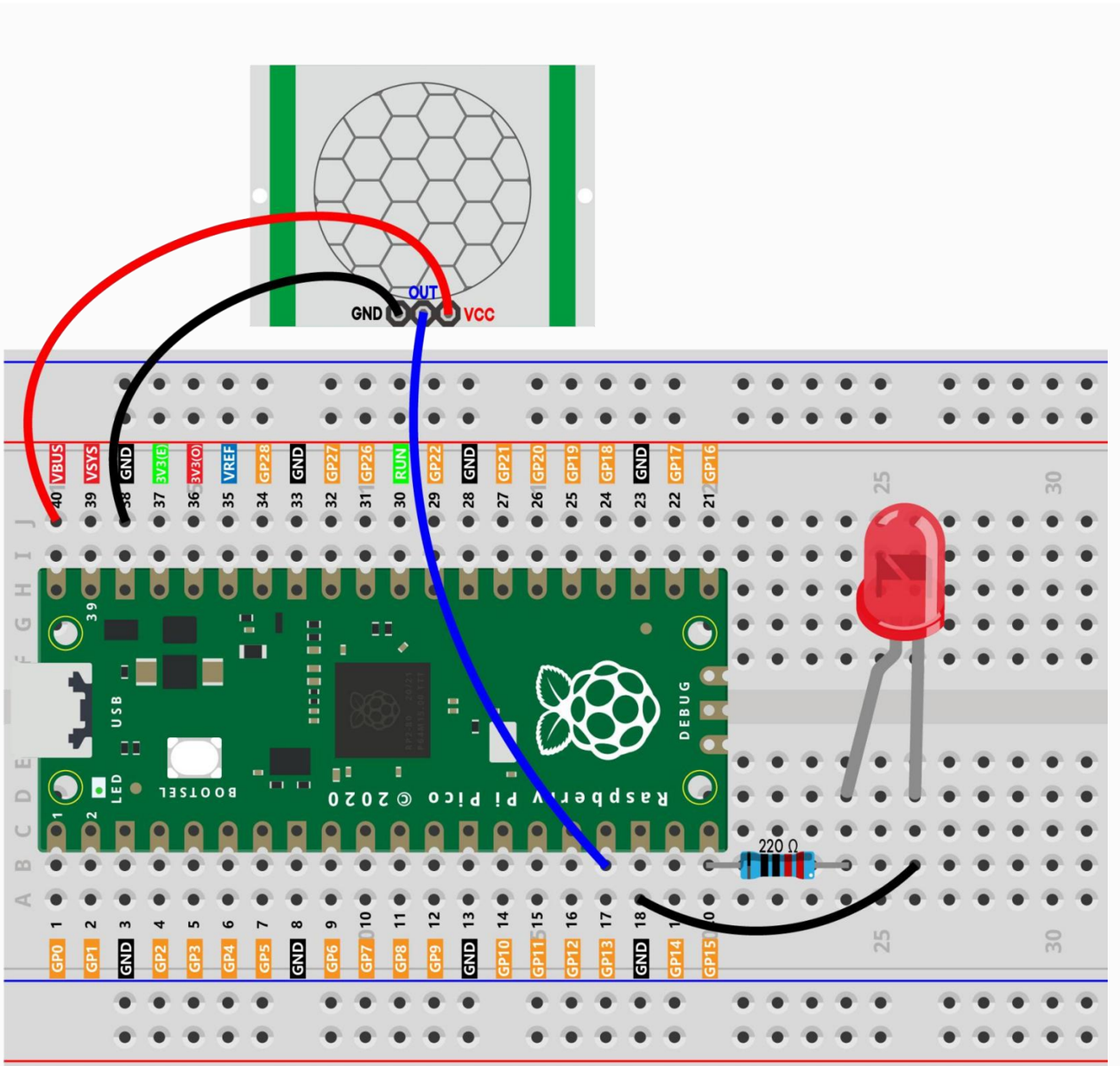
In this project,use PIR and LED to build an Intruder Alarm.

Passive infrared sensor (PIR sensor) is a common sensor that can measure infrared (IR) light emitted by objects in its field of view. Simply put, it will receive infrared radiation emitted from the body, thereby detecting the movement of people and other animals. More specifically, it tells the main control board that someone has entered your room.



Tip:Learn more about PIR Motion Sensor LED

Wiring



Note:

①The working voltage of PIR senso is 5V. Use the VBUS pin to power it.

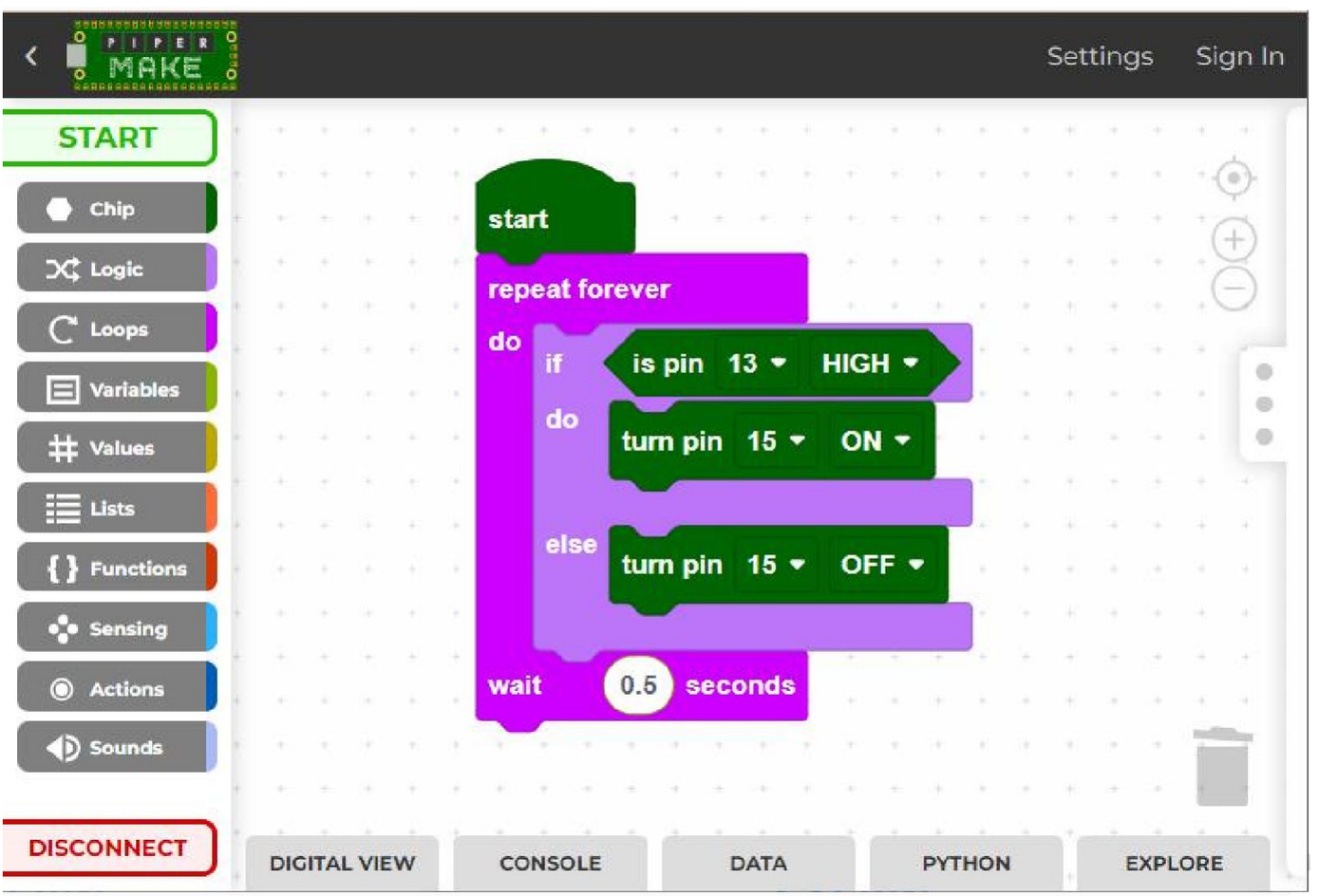
Code

Visit link： https://make.playpiper.com/. Switch to "**CREATIVE MODE**" and click on the "**IMPORT PROJECT**" button to import **Intruder_Alarm.png file.**

File path :Raspberry Pi Pico Starter Kit\Piper_Make\Piper Make Code\Project 4

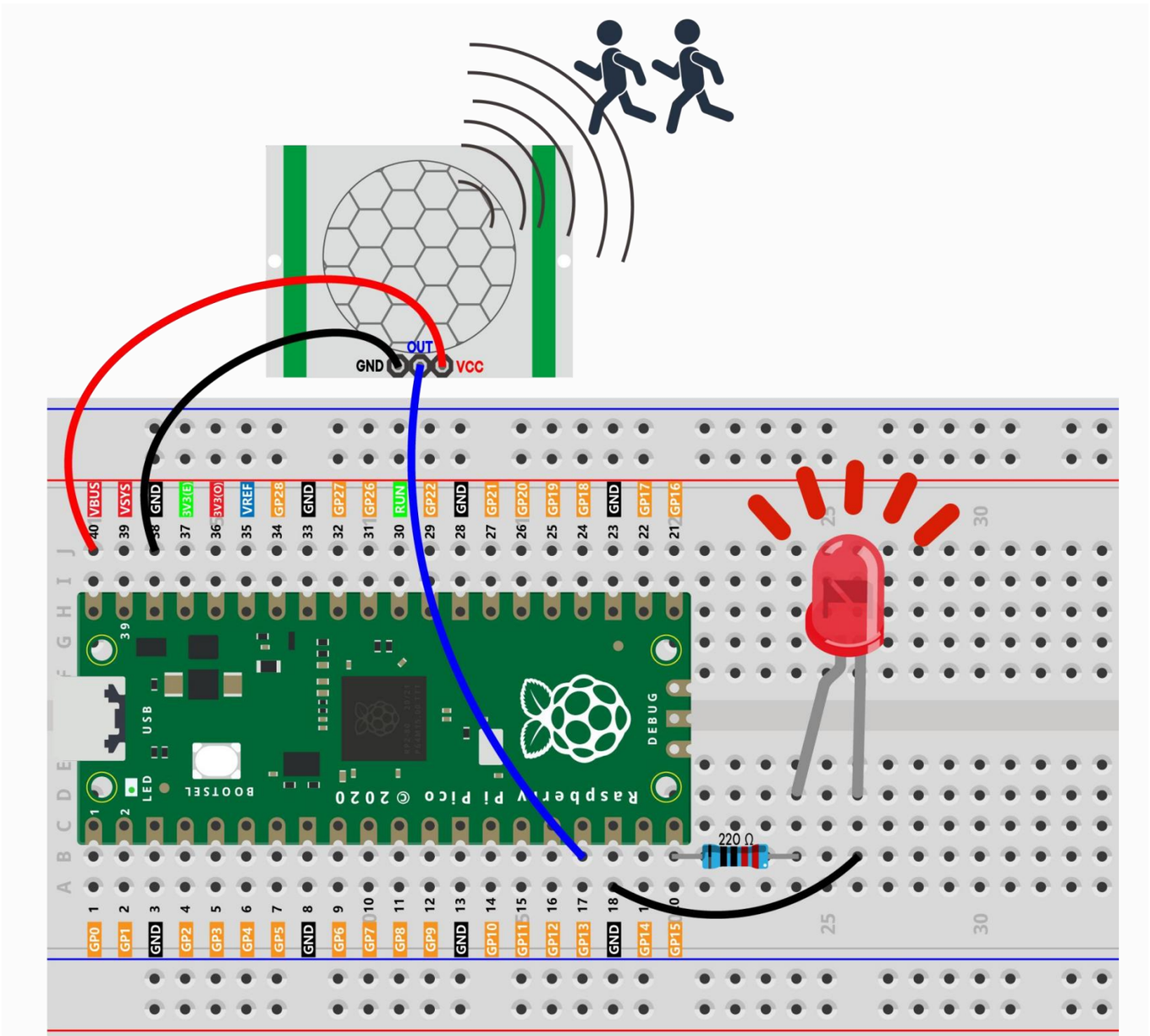
Note:

Don't forget to connect to Raspberry Pi Pico serial port——**CircuitPython CDC control (COMXX)** .Have a Question?



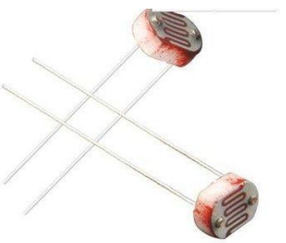
Now click on the **START** button to run this code.if someone walks into the PIR detection range, the led will be turn on. If START button is gray, it means that the Pico is not connected, please reconnect it.

Piper Make - Troubleshooting Guide Have a question about using PIR Motion Sensor?


OUT
GND
VCC
220 Ω
Raspberry Pi Pico ©2020
BOOTSEL
LED
USB
DEBUG

Project 5 Photoresistor Control LED

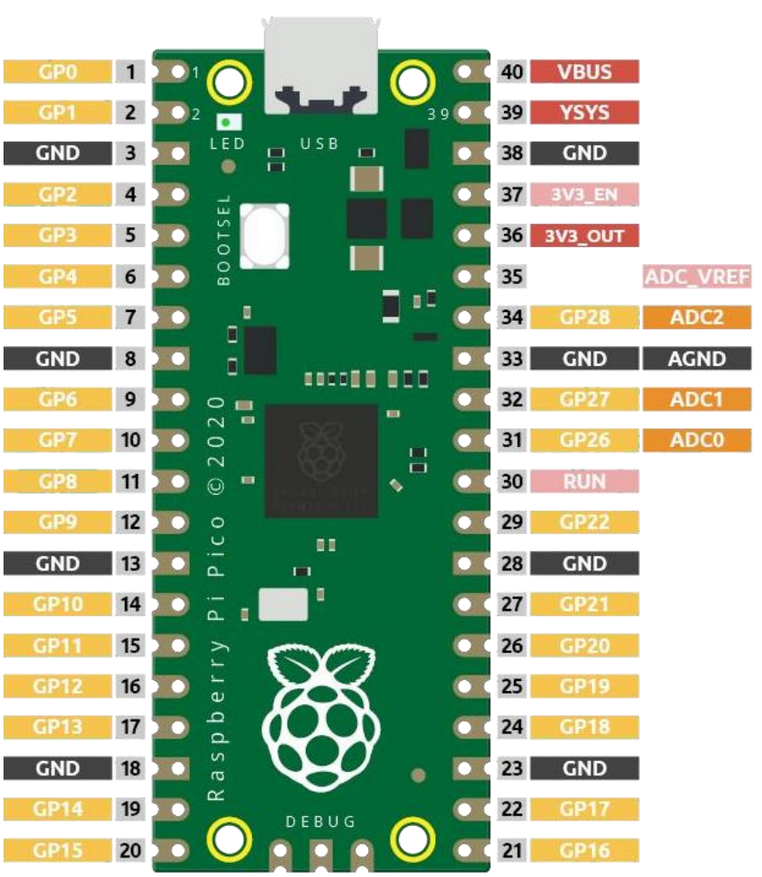
A Photoresistor is very sensitive to the amount of light present. We can take advantage of the characteristic to make a night lamp with the following function: when the ambient light is less (darker environment) the LED will automatically turn on and when the ambient light is greater (brighter environment) the LED will automatically turn off.



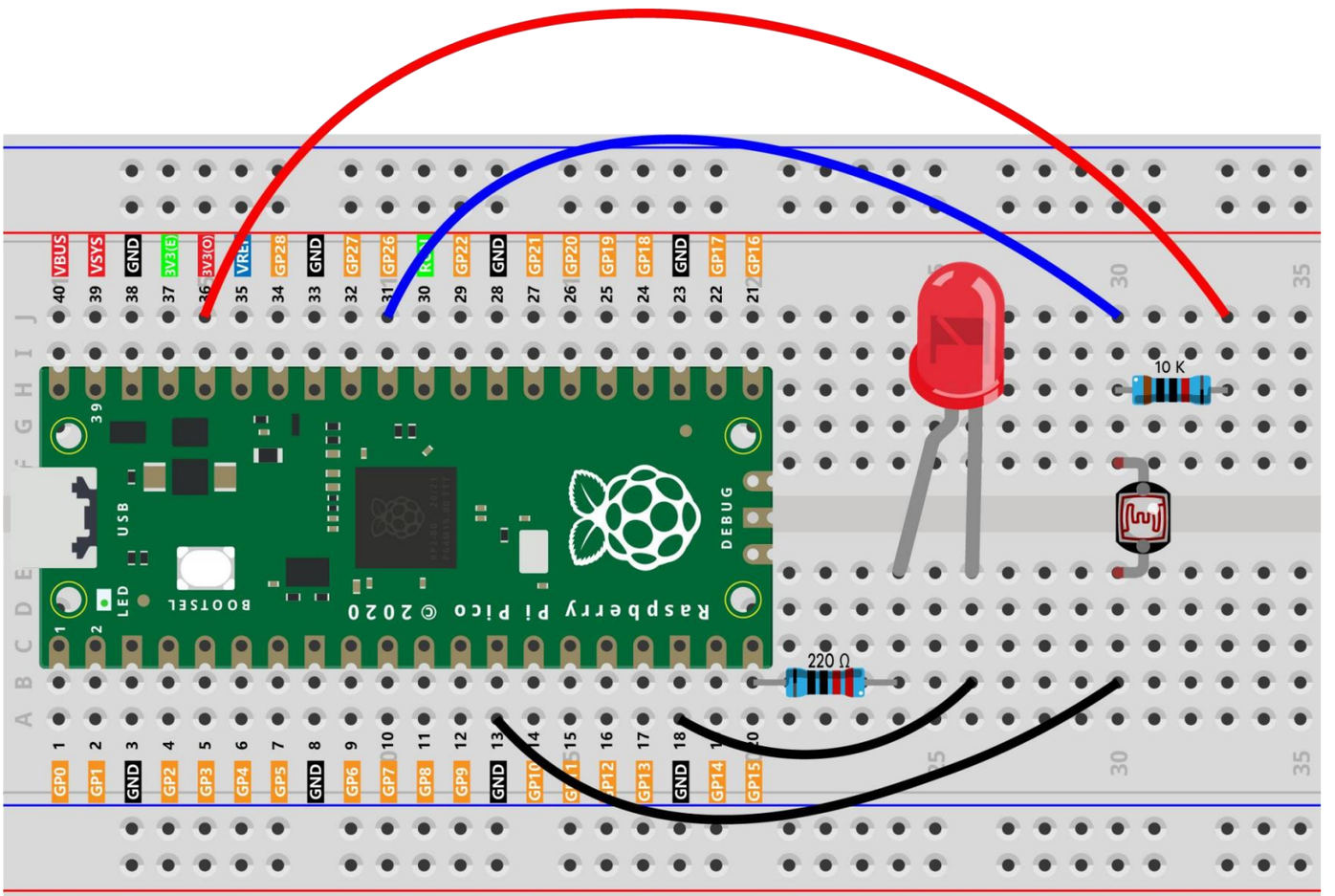
Tip:Learn more about Photoresistor

ADC ChannelsRaspberry Pi Pico

Raspberry Pi Pico has 5 ADC channels, which are ADC0(GP26), ADC1(GP27), ADC2(GP28), ADC3(GP29): used to measure VSYS on Pico board, and ADC4, which directly connects to the built-in temperature sensor of RP2040 chip. Therefore, there are only three generic ADC channels that can be directly used, namely, ADC0,ADC1 and ADC2.



Wiring

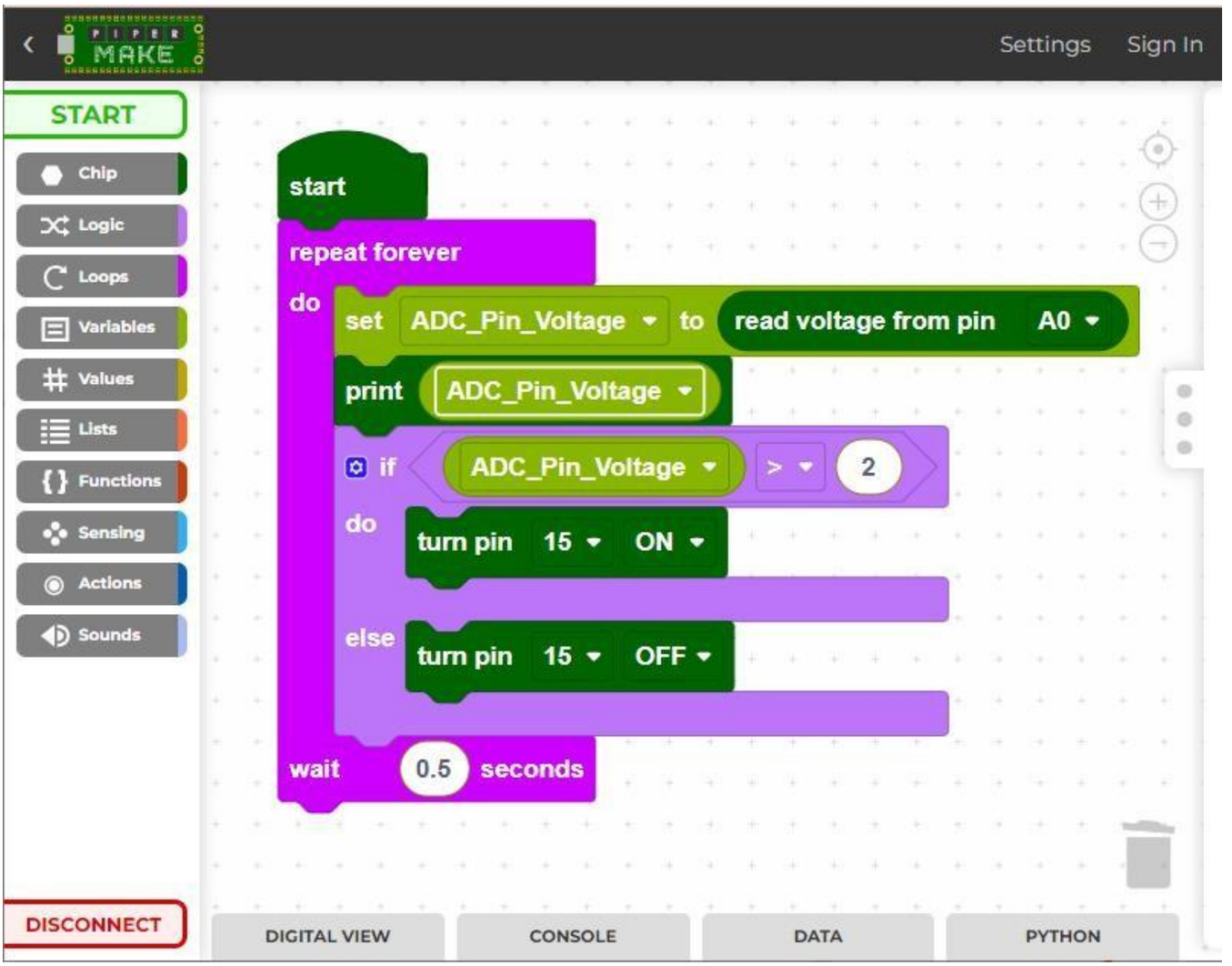


Code

Visit link： https://make.playpiper.com/. Switch to "**CREATIVE MODE**" and click on the "**IMPORT PROJECT**" button to import **Photoresistor_Control_LED.png file.**

File path :Raspberry Pi Pico Starter Kit\Piper_Make\Piper Make Code\Project 5

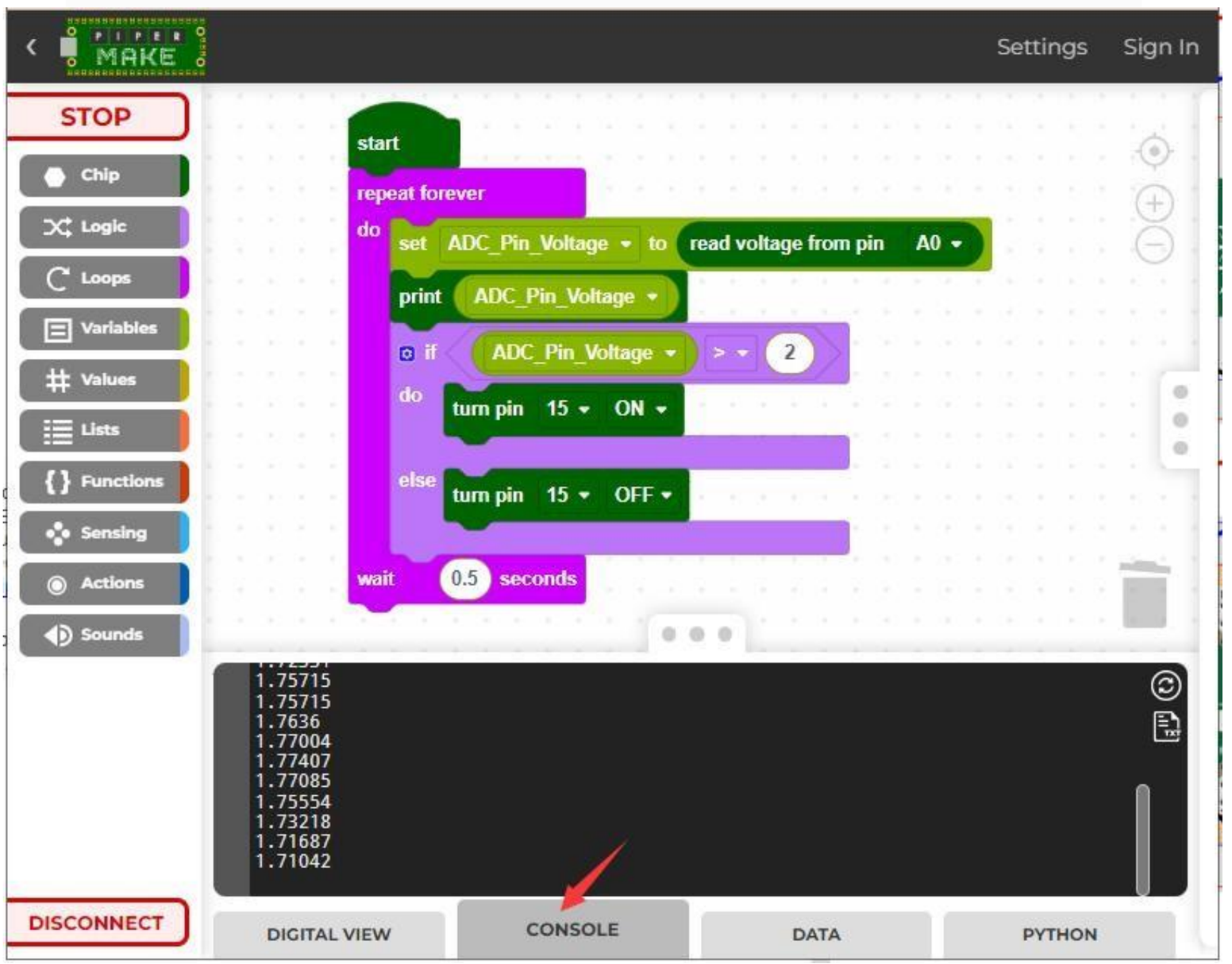
Note:
Don't forget to connect to Raspberry Pi Pico serial port——**CircuitPython CDC control (COMXX)** .Have a Question?

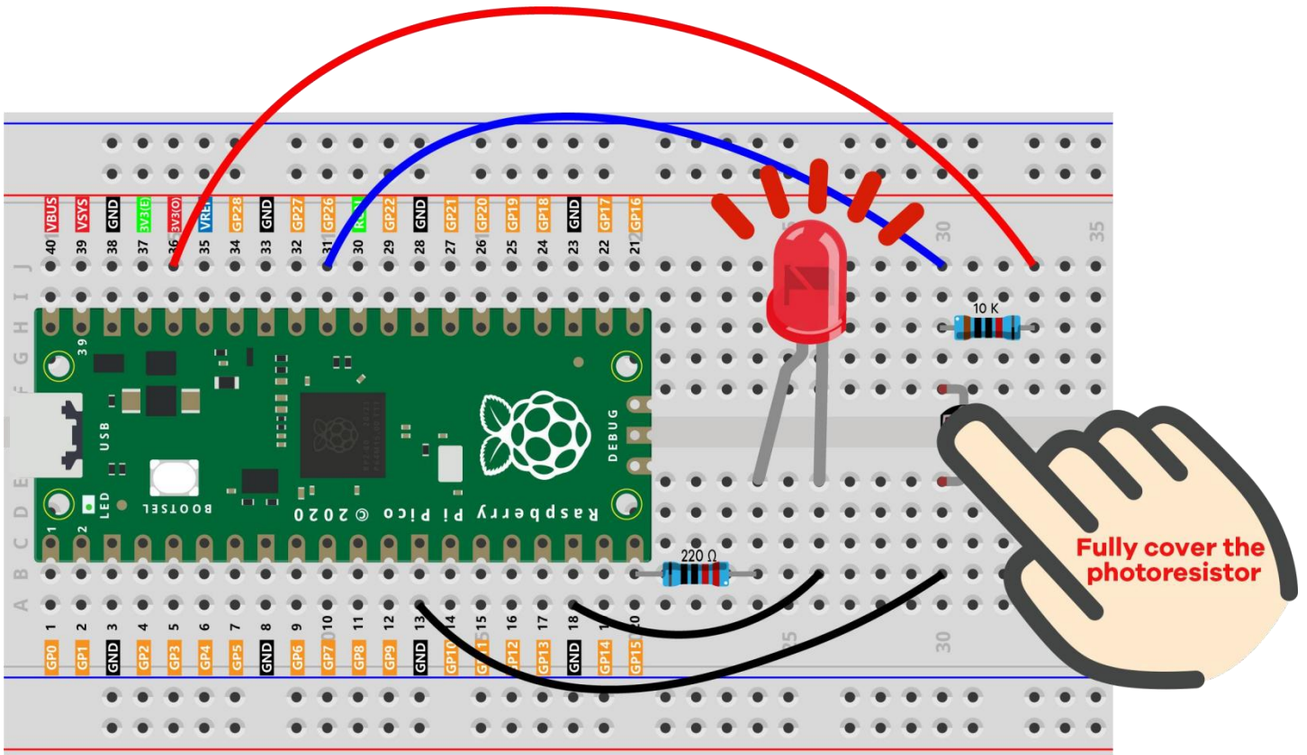


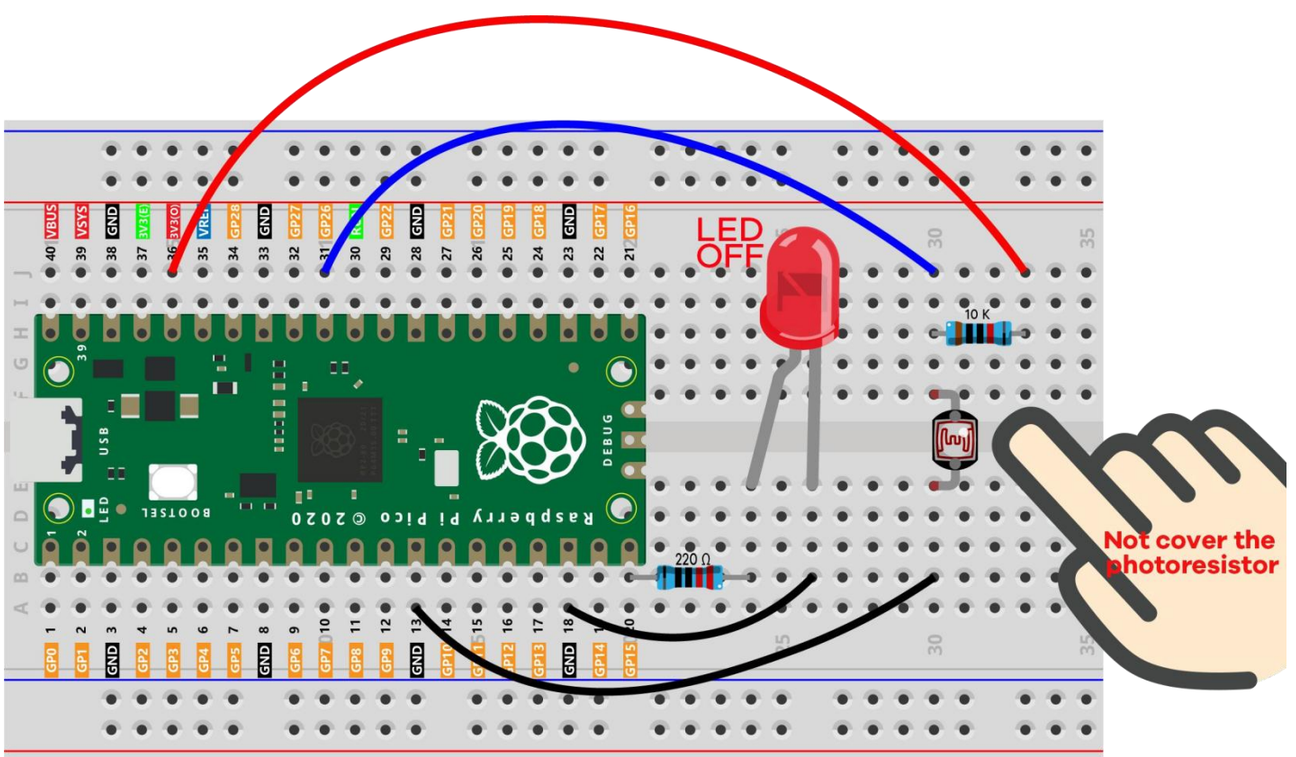
Now click on the **START** button to run this code.Cover the photoresistor with your hands, the LED will turn on.when the ambient light is greater (brighter environment) the LED will automatically turn off. If **START** button is gray, it means that the Pico is not connected, please reconnect it.

[Piper Make - Troubleshooting Guide]

You can click on the “Console” tag at the bottom to see the “ADC_Pin_Voltage” printed in the console.

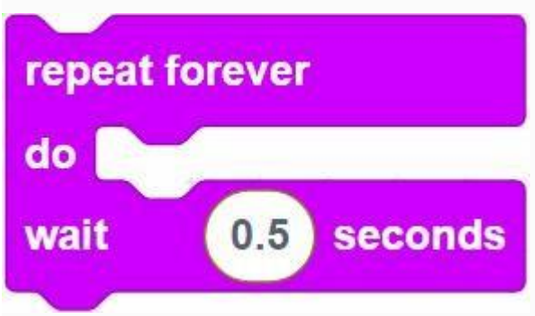



10 K
220 Ω
Fully cover the photoresistor


LED OFF
10 K
220 Ω
Not cover the photoresistor

Code Explanation

- [read voltage from pin A0/A1/A2]:This block is used to read the voltage value of the Pico ADC pin, and the output value ranges from 0-3.3V.ADC0(GP26), ADC1(GP27), ADC2(GP28)



- [repeat forever do() wait()seconds]: Means that the blocks in it will be executed repeatedly, and the execution time interval is defined by yourself.

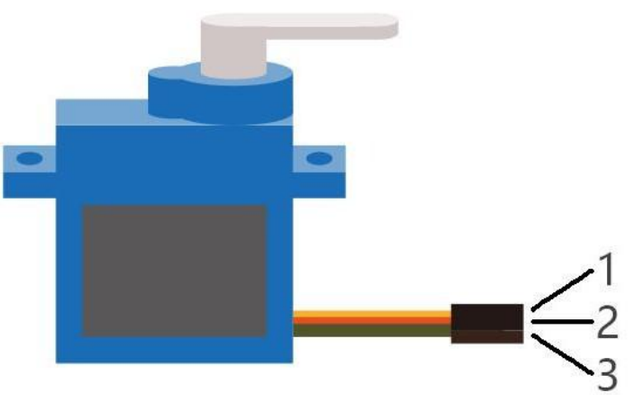
- [turn pin () ON/OFF]: Indicates that a certain pin is placed in a high level (ON) or a low level (OFF).

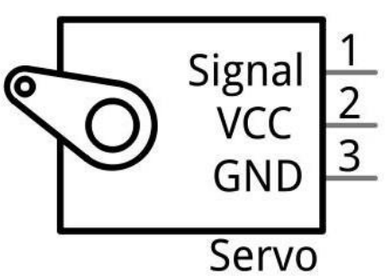
Project 6 Servo

In this project,we try to make the servo sway!

Servo is a position (angle) servo device, which is suitable for those control systems that require constant angle changes and can be maintained. It has been widely used in high-end remote control toys, such as airplanes, submarine models, and remote control robots.

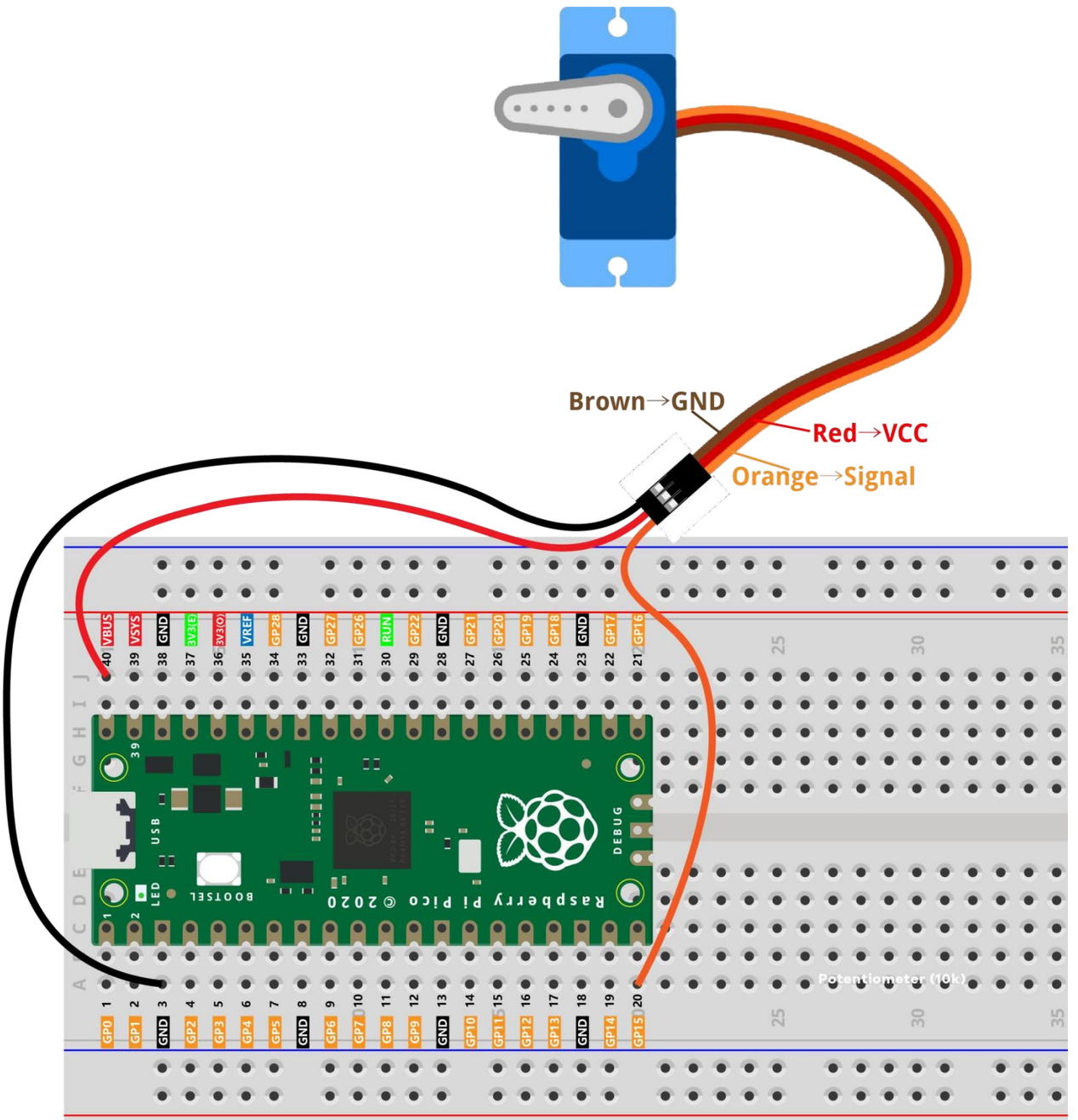
Servo is a compact package which consists of a DC Motor, a set of reduction gears to provide torque, a sensor and control circuit board. Most Servos only have a 180-degree range of motion via their “horn”. Servos can output higher torque than a simple DC Motor alone and they are widely used to control motion in model cars,model airplanes, robots, etc. Servos have three wire leads which usually terminate to a male or female 3-pin plug. Two leads are for electric power: Positive (2-VCC, Red wire), Negative (3-GND, Brown wire), and the signal line (1-Signal, Orange wire) as represented in the Servo provided in your Kit.





Tip:Learn more about servo

Wiring



Note:

The working voltage of Servo is 5V. Use the VBUS pin to power it.

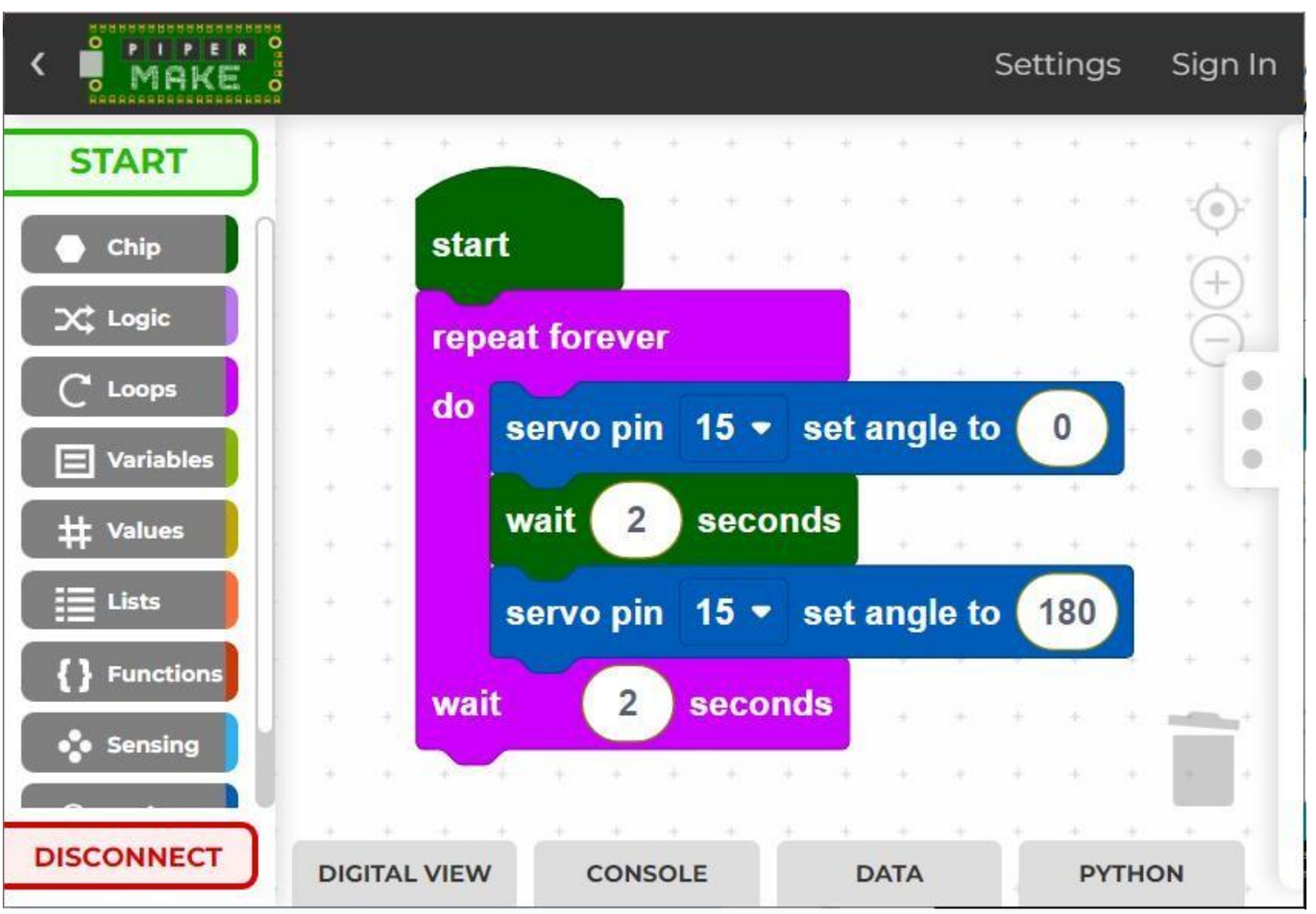
Code

Visit link： https://make.playpiper.com/. Switch to "**CREATIVE MODE**" and click on the "**IMPORT PR1OJECT**" button to import **Servo.png file.**

File path :Raspberry Pi Pico Starter Kit\Piper_Make\Piper Make Code\Project 6

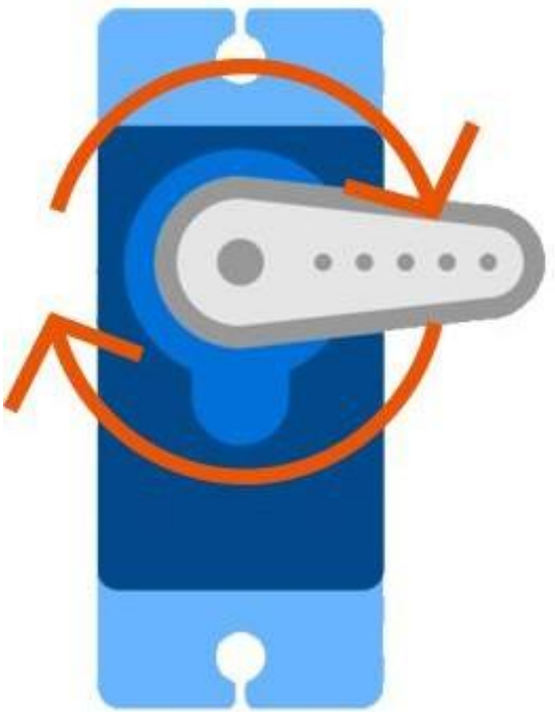
Note:

Don't forget to connect to Raspberry Pi Pico serial port——**CircuitPython CDC control (COMXX)** .Have a Question?



Now click on the **START** button to run this code.If necessary, press the Servo Arm into the Servo output shaft, then you can see the Servo Arm swinging back and forth from 0° to 180°. If **START** button is gray, it means that the Pico is not connected, please reconnect it.

Piper Make - Troubleshooting Guide



Code Explanation

- [servo pin () set angle to ()]: This block is used to set the rotation angle of the servo, the range is 0~180 degrees.

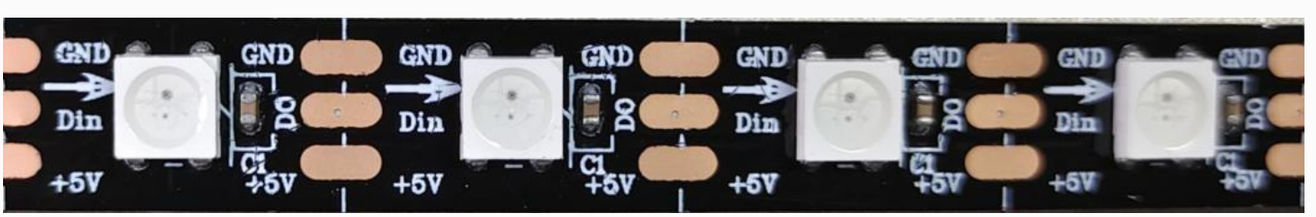
Project 7 RGB LED Strip

The kit is equipped with a WS2812 RGB LED Strip, which can display colorful colors, and each LED can be independently controlled.

RGB LED Strip

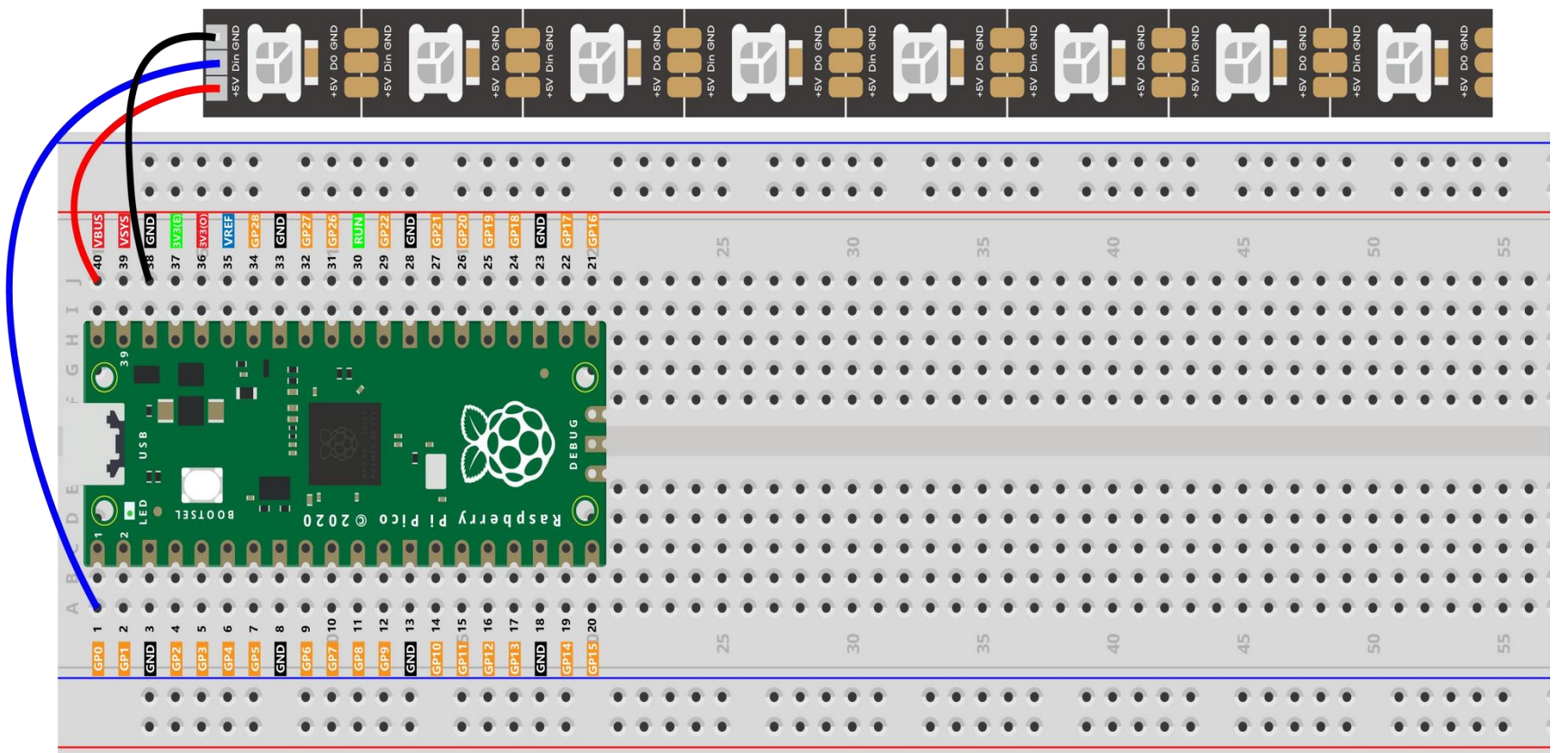
WS2812 is a intelligent control LED light source that the control circuit and RGB chip are integrated in a package of 5050 components. It internal include intelligent digital port data latch and signal reshaping amplification drive circuit. Also include a precision internal oscillator and a 12V voltage programmable constant current control part, effectively ensuring the pixel point light color height consistent.

The data transfer protocol use single NZR communication mode. After the pixel power-on reset, the DIN port receive data from controller, the first pixel collect initial 24bit data then sent to the internal data latch, the other data which reshaping by the internal signal reshaping amplification circuit sent to the next cascade pixel through the DO port. After transmission for each pixel，the signal to reduce 24bit. pixel adopt auto reshaping transmit technology, making the pixel cascade number is not limited the signal transmission, only depend on the speed of signal transmission.



Tip:Learn more about RGB LED Strip

Wiring



Note:

The working voltage of servo is 5V. Use the VBUS pin to power it.

Although the LED Strip with any number of LEDs can be used in Pico, the power of its VBUS pin is limited. Here, we will use eight LEDs, which are safe. But if you want to use more LEDs, you need to add a separate power supply.

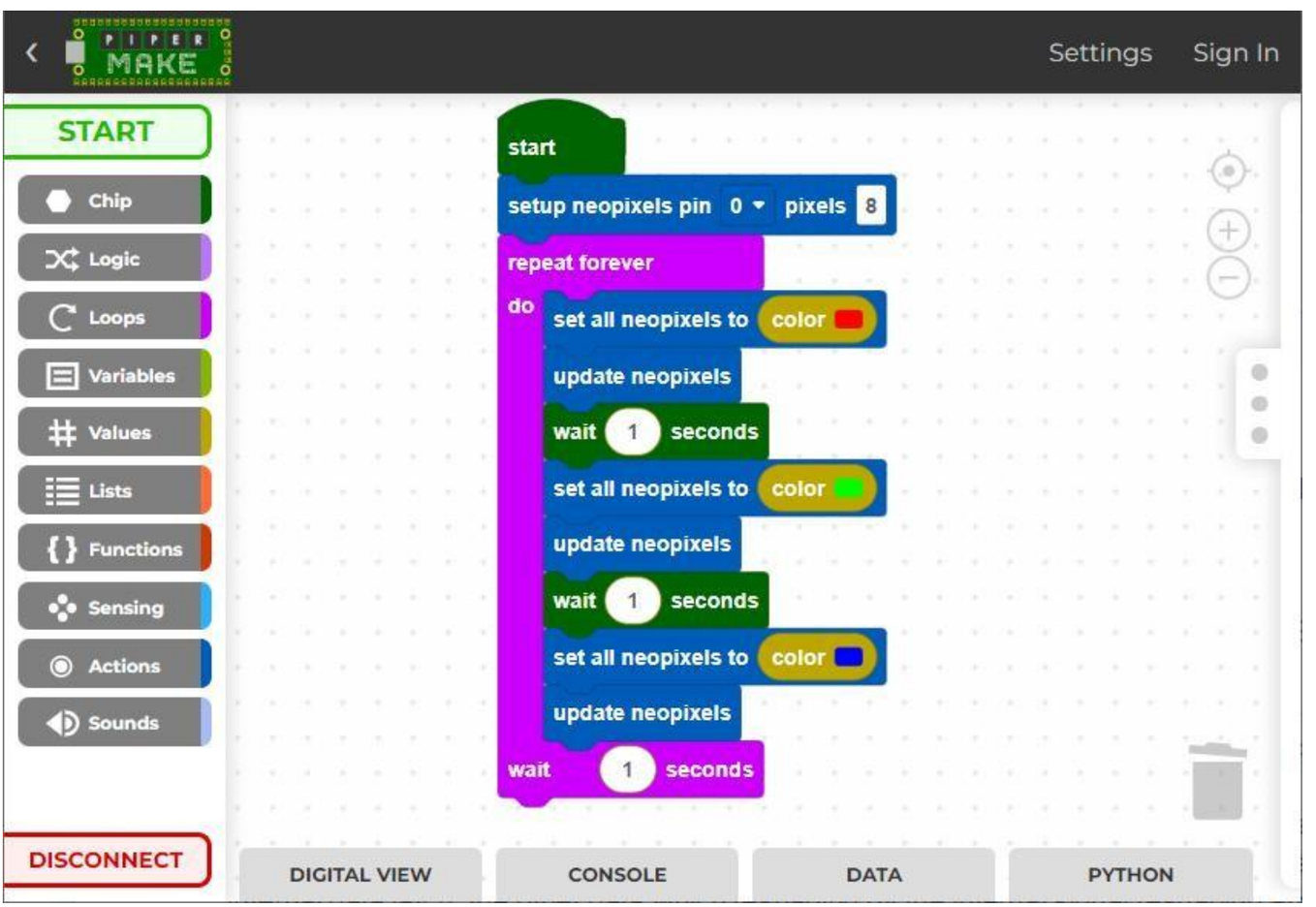
Code

Visit link： https://make.playpiper.com/. Switch to "**CREATIVE MODE**" and click on the "**IMPORT PR1OJECT**" button to import **RGB_LED_Strip.png file.**

File path :Raspberry Pi Pico Starter Kit\Piper_Make\Piper Make Code\Project 7

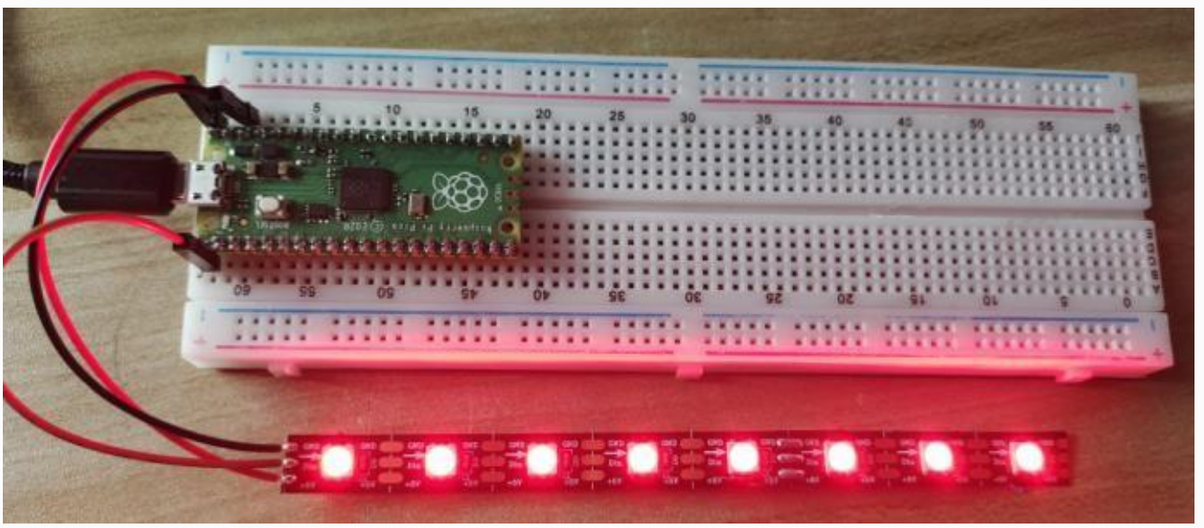
Note:

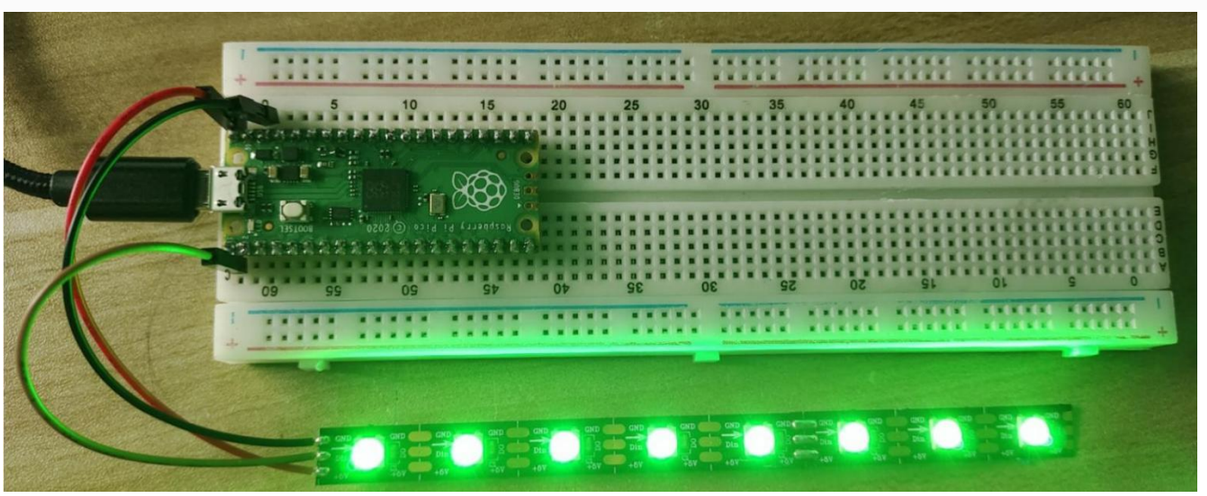
Don't forget to connect to Raspberry Pi Pico serial port——**CircuitPython CDC control (COMXX)** .Have a Question?

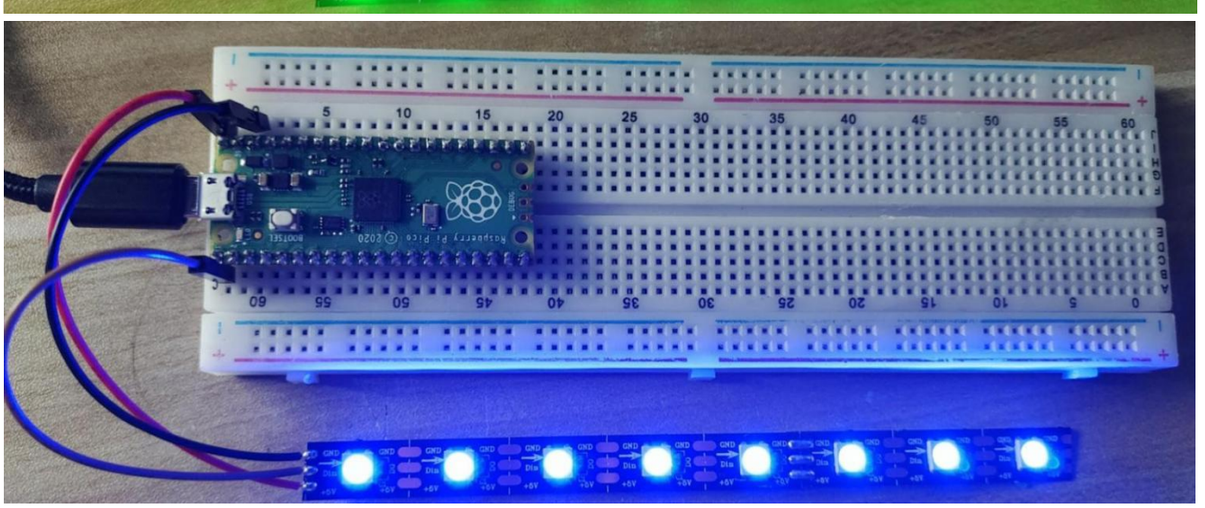


Now click on the **START** button to run this code.LED light bar display color: **red**->**green**->**blue**, changing another color every 1 second. If **START** button is gray, it means that the Pico is not connected, please reconnect it.

Piper Make - Troubleshooting Guide







Code Explanation

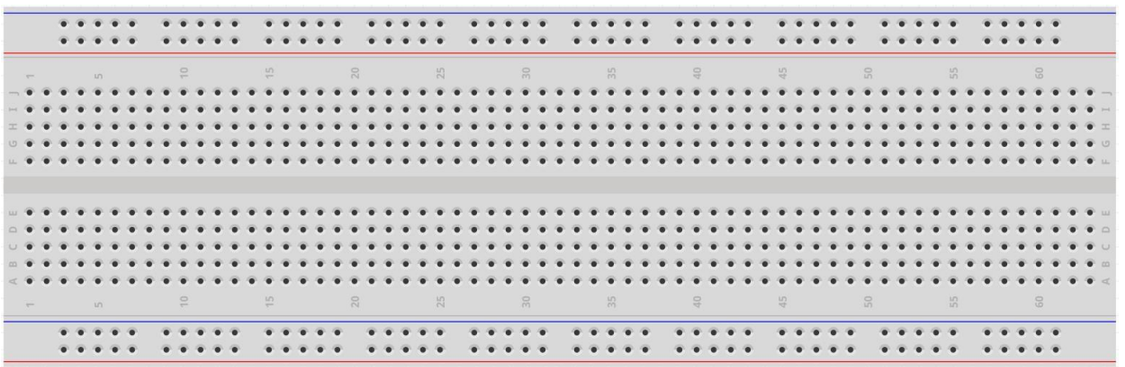
- [set all neopixels to ()]: Use to set a color for all LEDs, there are 13*9 colors, the top right color is black to make LEDs to go off.

- [updates neopixels]: Update the effect to the WS2812 RGB LED Strip.

Components Introduction

- Breadboard
- LED
- Button
- RGB LED
- Resistor
- Transistor
- Buzzer
- Potentiometer
- Photoresistor
- IR Proximity Sensor Module
- Thermistor
- Tilt Switch
- Servo
- I2C LCD1602
- PIR Motion Sensor
- WS2812 RGB 8 LEDs Strip

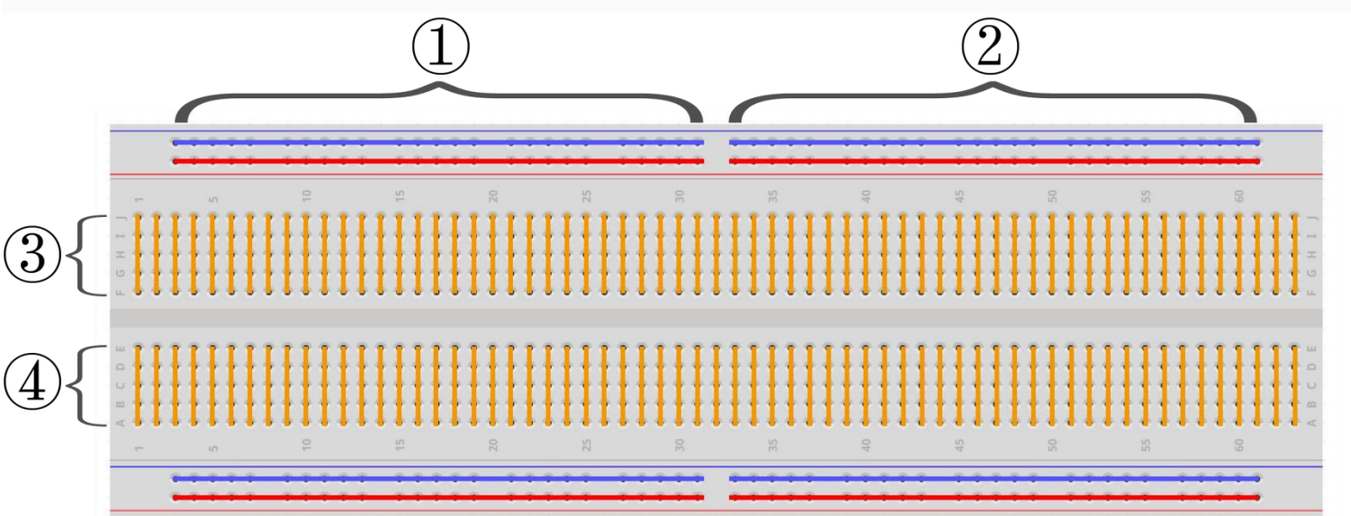
➢ Breadboard



A breadboard is a construction base for prototyping of electronics. Originally the word referred to a literal bread board, a polished piece of wood used for slicing bread.[1] In the 1970s the solderless breadboard (a.k.a. plugboard, a terminal array board) became available and nowadays the term “breadboard” is commonly used to refer to these.

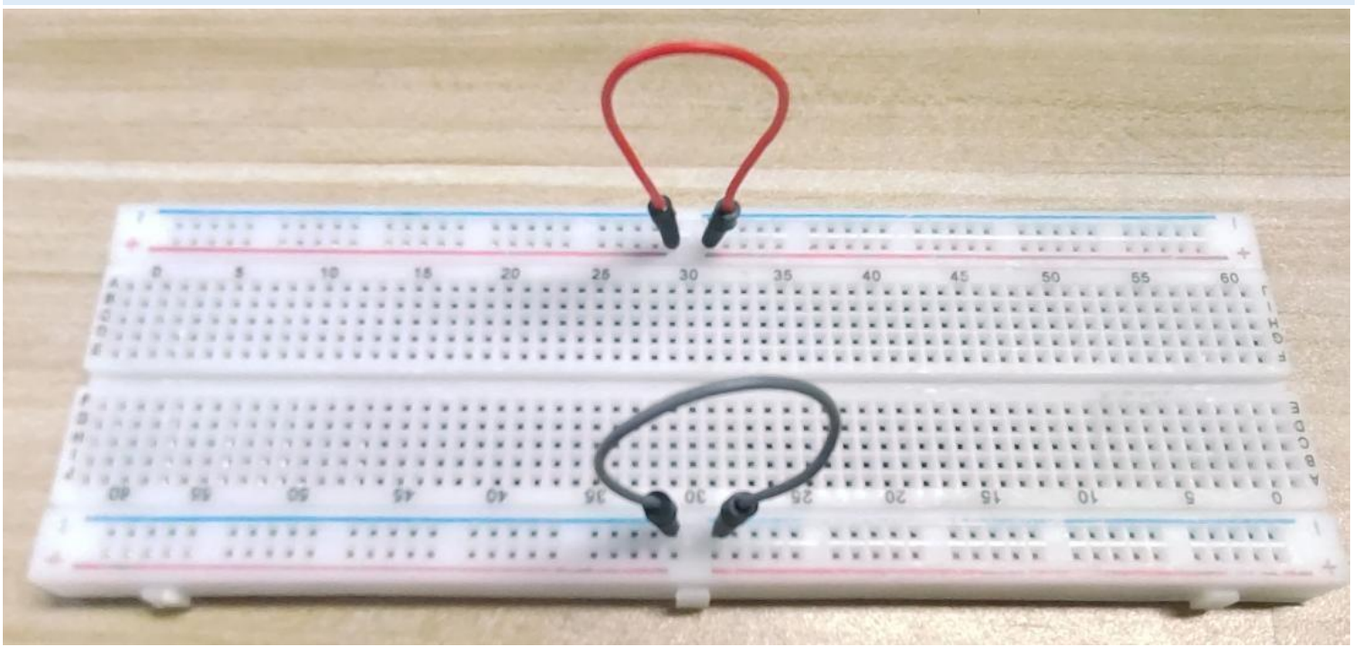
It is used to build and test circuits quickly before finishing any circuit design. And it has many holes into which components mentioned above can be inserted like ICs and resistors as well as jumper wires. The breadboard allows you to plug in and remove components easily.

The picture shows the internal structure of a breadboard. Although these holes on the breadboard appear to be independent of each other, they are actually connected to each other through metal strips internally.



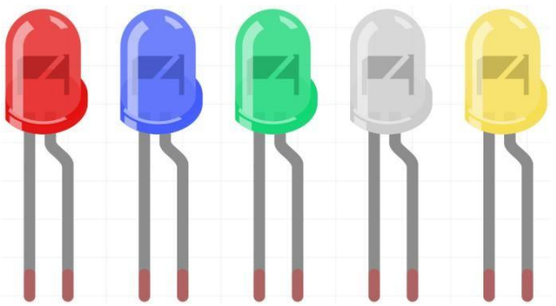
Note:

Area ① and area ② are not connected, and area ① and area ② can be connected through a jumper.

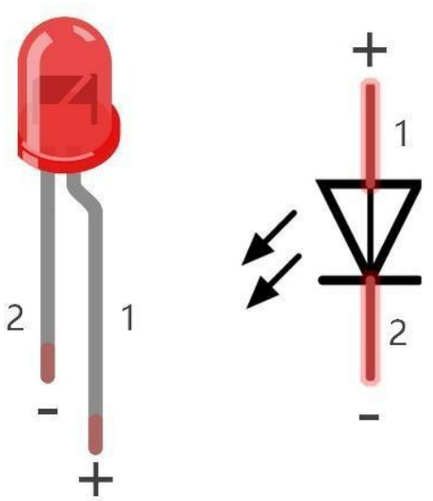


If you want to know more about breadboard, refer to: How to Use a Breadboard - Science Buddies

➢ LED



An LED is a type of diode. All diodes only work if current is flowing in the correct direction and have two Poles.An LED will only work (light up) if the longer pin (+) of LED is connected to the positive output from a power source and the shorter pin is connected to the negative (-). Negative output is also referred to as Ground (GND). This type of component is known as “Polar” (think One-Way Street).All common 2 lead diodes are the same in this respect. Diodes work only if the voltage of its positive electrode is higher than its negative electrode and there is a narrow range of operating voltage for most all common diodes of 1.9 and 3.4V. If you use much more than 3.3V the LED will be damaged and burn out.



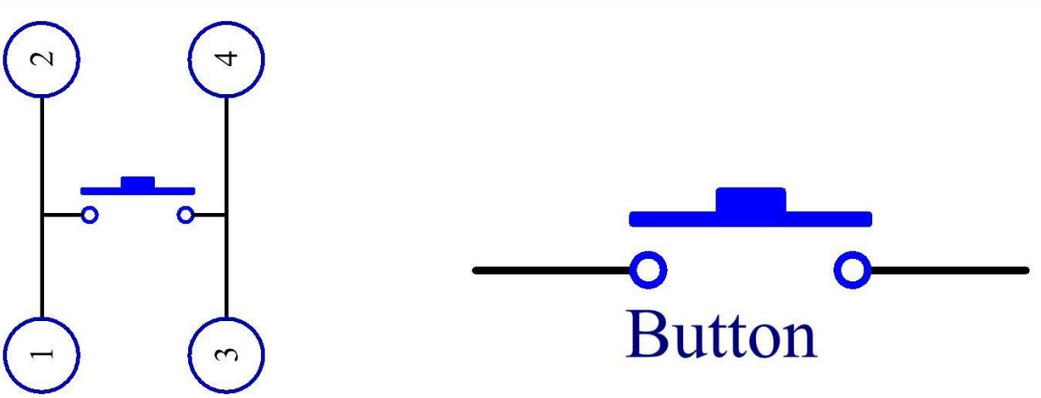
Note:

LEDs cannot be directly connected to a power supply, which usually ends in a damaged component. A
resistor with a specified resistance value must be connected in series to the LED you plan to use.

➢ Button

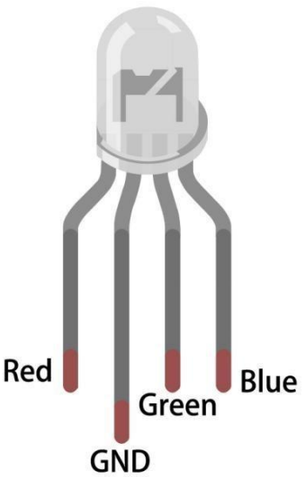
Buttons are a common component used to control electronic devices. They are usually used as switches to connect or break circuits. Although buttons come in a variety of sizes and shapes, the one used here is a 6mm mini-button as shown in the following pictures. Pin 1 is connected to pin 2 and pin 3 to pin 4. So you just need to connect either of pin 1 and pin 2 to pin 3 or pin 4.

The following is the internal structure of a button. The symbol on the right below is usually used to represent a button in circuits.

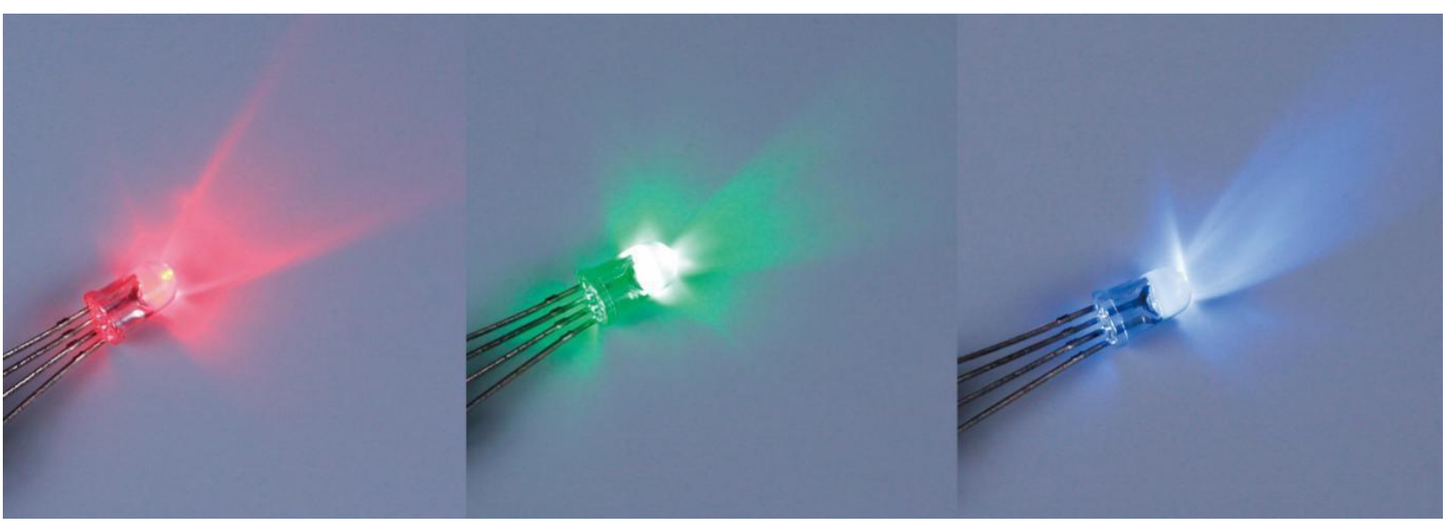


Since the pin 1 is connected to pin 2, and pin 3 to pin 4, when the button is pressed, the 4 pins are connected, thus closing the circuit.

➢ RGB LED

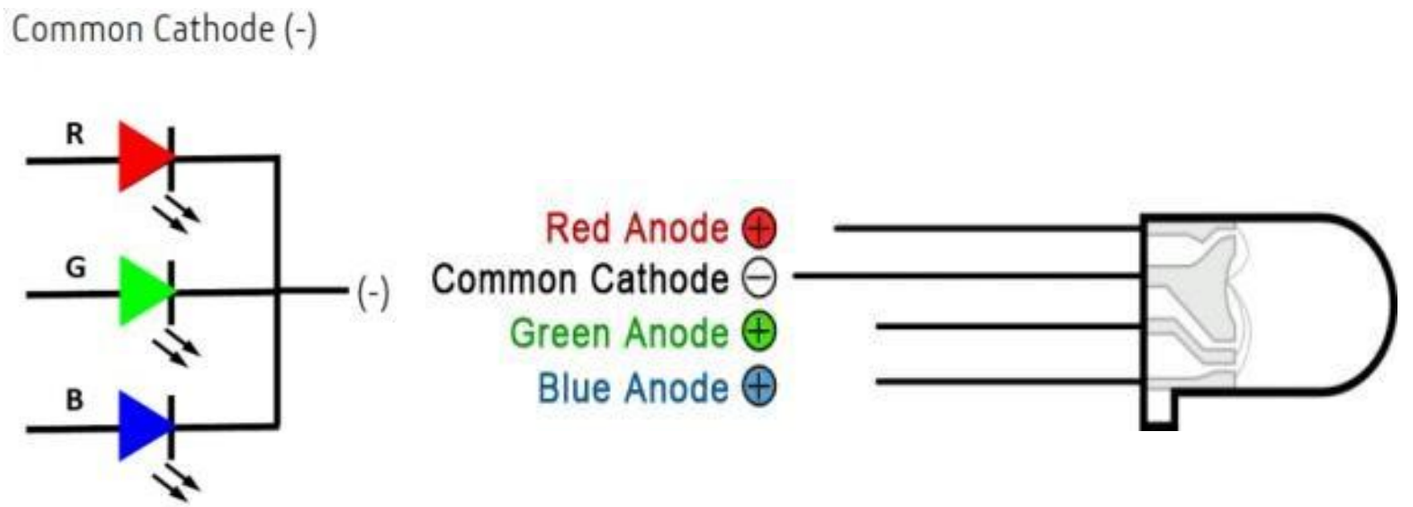


RGB LEDs emit light in various colors. An RGB LED packages three LEDs of red, green, and blue into a transparent or semitransparent plastic shell. It can display various colors by changing the input voltage of the three pins and superimpose them, which, according to statistics, can create 16,777,216 different colors.



RGB LEDs can be categorized into common anode and common cathode ones. In this kit, the latter is used. The common cathode, or CC, means to connect the cathodes of the three LEDs. After you connect it with GND and plug in the three pins, the LED will flash the corresponding color.

Its circuit symbol is shown as figure.

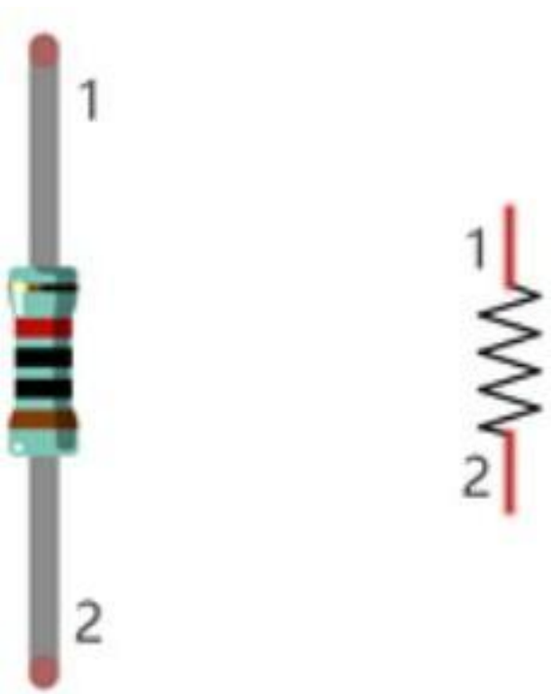


An RGB LED has 4 pins: the longest one is GND; the others are Red, Green and Blue. Touch its plastic shell and you will find a cut. The pin closest to the cut is the first pin, marked as Red, then GND, Green and Blue in turn.

➢ Resistor

Resistors use Ohms (Ω) as the unit of measurement of their resistance (R). 1MΩ=1000kΩ, 1kΩ=1000Ω.

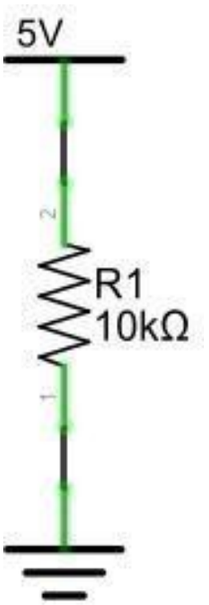
A resistor is a passive electrical component that limits or regulates the flow of current in an electronic circuit.On the left, we see a physical representation of a resistor, and the right is the symbol used to represent the presence of a resistor in a circuit diagram or schematic.



The bands of color on a resistor is a shorthand code used to identify its resistance value. For more details of resistor color codes, please refer to the appendix of this tutorial.

With a fixed voltage, there will be less current output with greater resistance added to the circuit. The relationship between Current, Voltage and Resistance can be expressed by this formula: I=V/R known as Ohm’ s Law where I = Current, V = Voltage and R = Resistance. Knowing the values of any two of these allows you to solve the value of the third.

In the following diagram, the current through R1 is: I=U/R=5V/10kΩ=0.0005A=0.5mA.



WARNING:

Never connect the two poles of a power supply with anything of low resistance value (i.e. a metal object or bare wire) this is a Short and results in high current that may damage the power supply and electronic components.

Note:

Unlike LEDs and Diodes, Resistors have no poles and re non-polar (it does not matter which direction you insert them into a circuit, it will work the same)

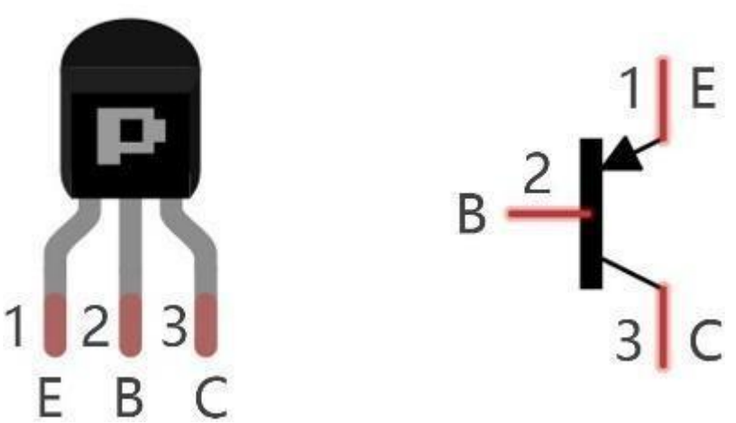
➢ Transistor

Because the buzzer requires such large current that GP of Raspberry Pi Pico output capability cannot meet the requirement, a transistor of NPN type is needed here to amplify the current.

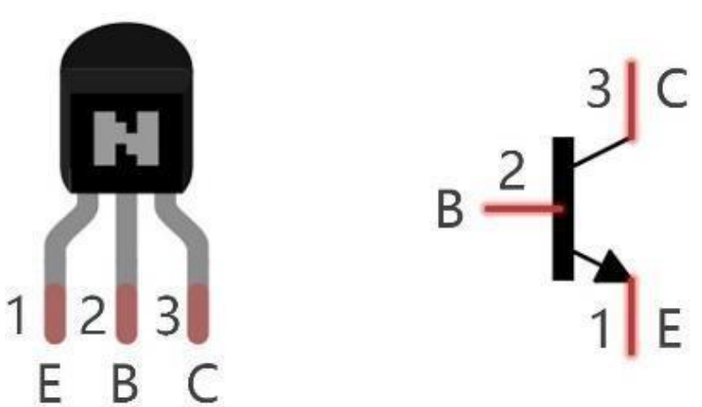
Transistor, the full name: semiconductor transistor, is a semiconductor device that controls current. Transistor

can be used to amplify weak signal, or works as a switch. It has three electrodes(PINs): base (b), collector (c) and emitter (e). When there is current passing between "be", "ce" will allow several-fold current (• t between "be" exceeds a certain value, "ce" will not allow current to increase any longer, at this point, transistor works in the saturation area. Transistor has two types as shown below: PNP and NPN.

PNP transistor



NPN transistor

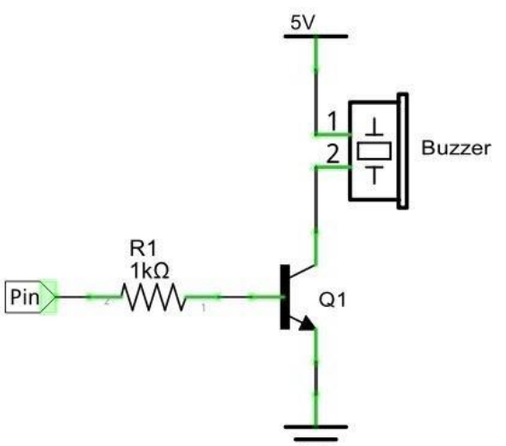


In our kit, There are two NPN transistor is marked with 8050.

Based on the transistor's characteristics, it is often used as a switch in digital circuits. As micro-controller's capacity to output current is very weak, we will use transistor to amplify current and drive large-current components.

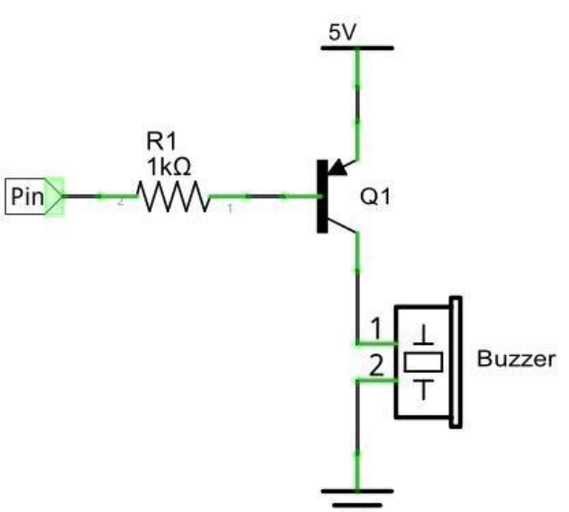
- When using NPN transistor to drive buzzer, we often adopt the following method. If GP outputs high level,current will flow through R1, the transistor will get conducted, and the buzzer will sound. If GP outputs low level, no current flows through R1, the transistor will not be conducted, and buzzer will not sound.

NPN transistor to drive buzzer



- When using PNP transistor to drive buzzer, we often adopt the following method. If GP outputs low level,current will flow through R1, the transistor will get conducted, and the buzzer will sound. If GP outputs high level, no current flows through R1, the transistor will not be conducted, and buzzer will not sound.

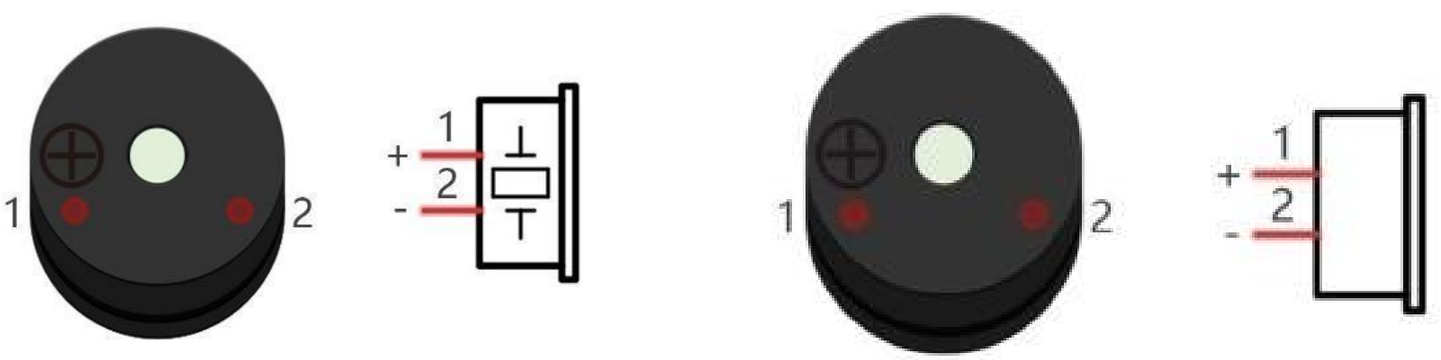
PNP transistor to drive buzzer



➢ Buzzer

Buzzer is a sounding component, which is widely used in electronic devices such as calculator, electronic warning clock and alarm. Buzzer has two types: active and passive. Active buzzer has oscillator inside, which will sound as long as it is supplied with power. Passive buzzer requires external oscillator signal (generally use PWM with different frequency) to make a sound.

Active buzzer **Passive buzzer**



Active buzzer is easy to use. Generally, it can only make a specific frequency of sound. Passive buzzer requires an external circuit to make a sound, but it can be controlled to make a sound with different frequency. The resonant frequency of the passive buzzer is 2kHz, which means the passive buzzer is loudest when its resonant frequency is 2kHz.

How to identify active and passive buzzer?

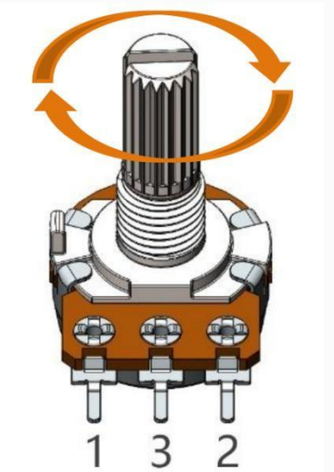
1. Usually, there is a label on the surface of active buzzer covering the vocal hole, but this is not an absolute judgment method.
2. Active buzzers are more complex than passive buzzers in their manufacture. There are many circuits and crystal oscillator elements inside active buzzers; all of this is usually protected with a waterproof coating(and a housing) exposing only its pins from the underside. On the other hand, passive buzzers do not have protective coatings on their

underside. From the pin holes viewing of a passive buzzer, you can see the circuit board, coils, and a permanent magnet (all or any combination of these components depending on the model.

Active buzzer

Passive buzzer

➢ Potentiometer

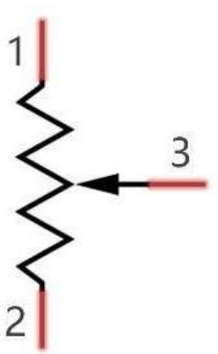


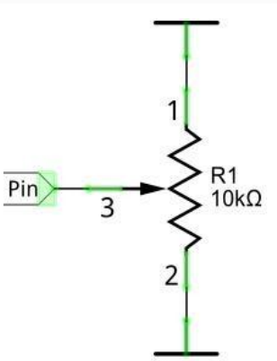
Potentiometer is also a resistance component with 3 terminals and its resistance value can be adjusted according to some regular variation.

Potentiometers come in various shapes, sizes, and values, but they all have the following things in common:

- They have three terminals (or connection points).
- They have a knob, screw, or slider that can be moved to vary the resistance between the middle terminal and either one of the outer terminals.
- The resistance between the middle terminal and either one of the outer terminals varies from 0 Ω to the maximum resistance of the pot as the knob, screw, or slider is moved.

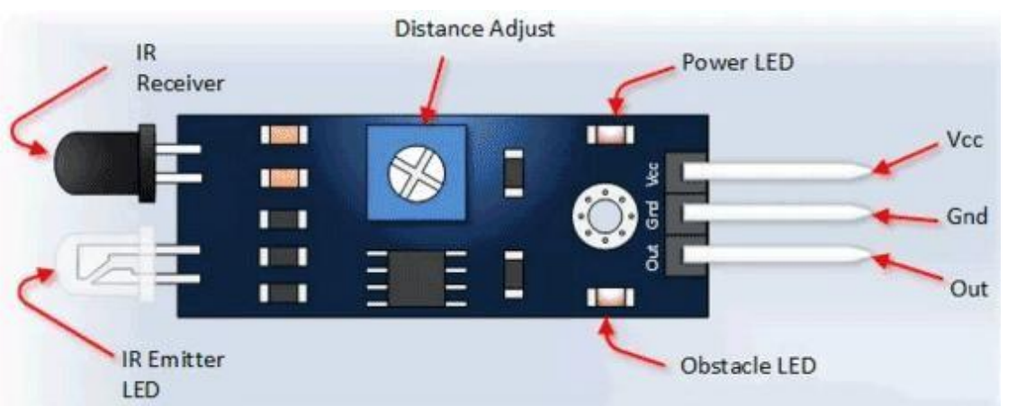
Here is the circuit symbol of potentiometer.10K ohm potentiometer is included in the kit





➢ IR Proximity Sensor Module

Proximity Sensor are used to detect objects and obstacles in front of sensor. Sensor keeps transmitting infrared light and when any object comes near, it is detected by the sensor by monitoring the reflected light from the object. It can be used in robots for obstacle avoidance, for automatic doors, for parking aid devices or for security alarm systems, or contact less tachometer by measuring RPM of rotation objects like fan blades.

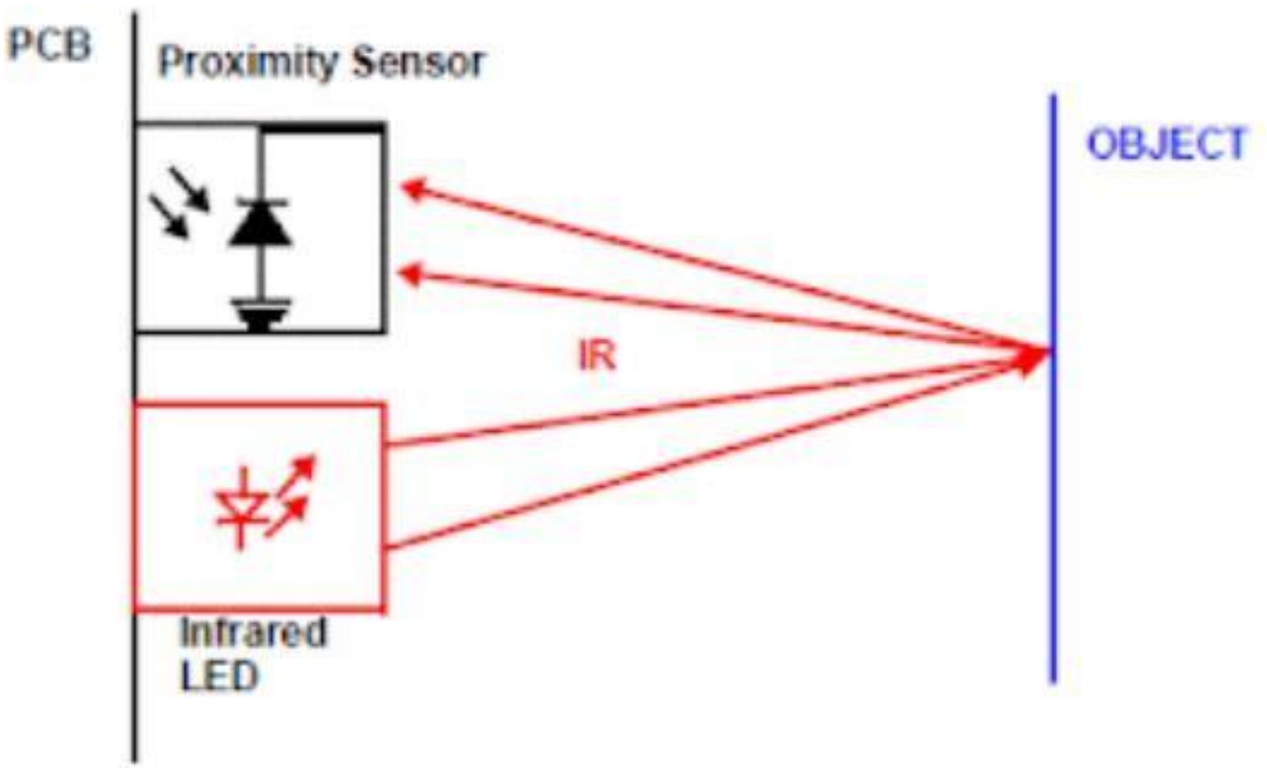


Pin, Control Indicator	Description
Vcc	3.3 to 5 Vdc Supply Input
Gnd	Ground Input
Out	Output that goes low when obstacle is in range
Power LED	Illuminates when power is applied
Obstacle LED	Illuminates when obstacle is detected
Distance Adjust	Adjust detection distance. CCW decreases distance. CW increases distance.
IR Emitter	Infrared emitter LED
IR Receiver	Infrared receiver that receives signal transmitted by Infrared emitter.

Digital OUTPUT

Digital **LOW** output on detecting objects in front.

Digital **HIGH** output on detecting no objects in front.

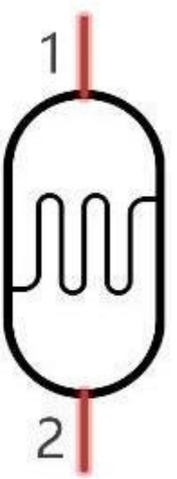


Distance Adjust

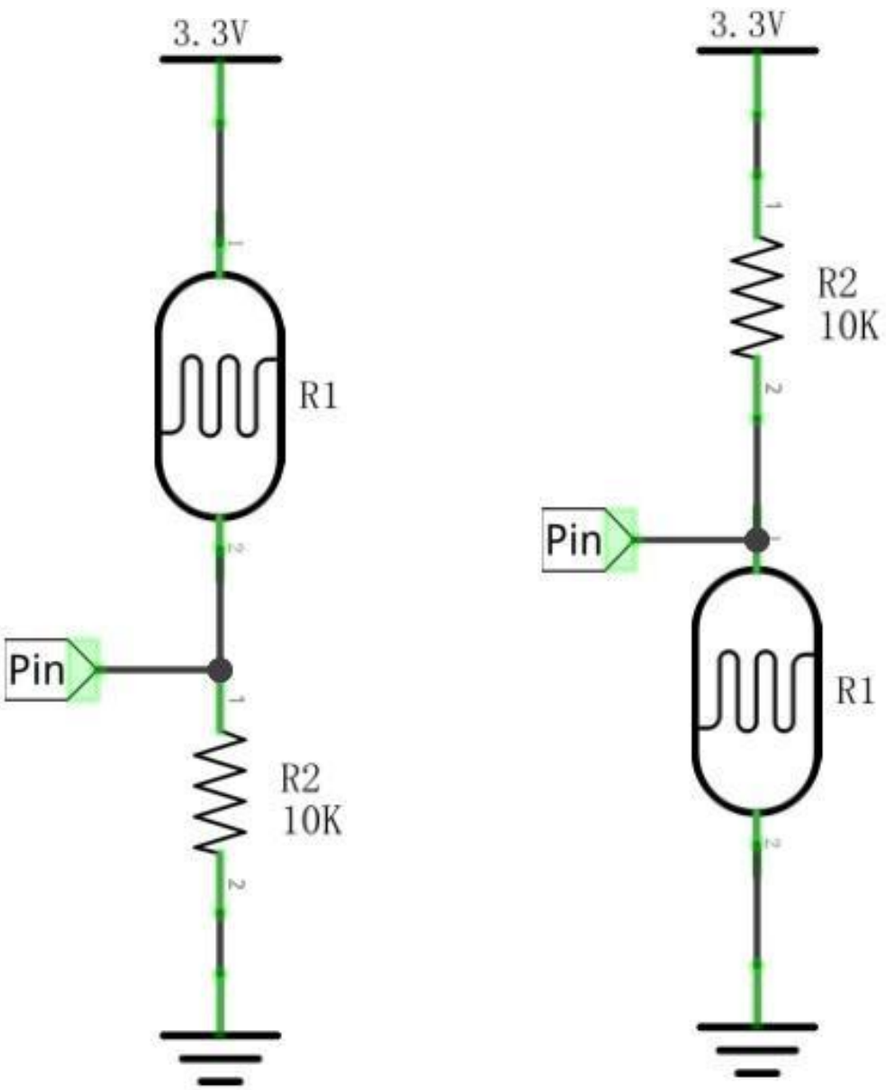
IR sensors are highly susceptible to ambient light and the IR sensor on this sensor is suitably covered to reduce effect of ambient light on the sensor. To For maximum, range the on board potentiometer should be used to calibrate the sensor. To set the potentiometer, use a screw driver and turn the potentiometer till the output LED just turns off.

➢ Photoresistor

Photoresistor is simply a light sensitive resistor. It is an active component that decreases resistance with respect to receiving luminosity (light) on the component's light sensitive surface. Photoresistor’s resistance value will change in proportion to the ambient light detected. With this characteristic, we can use a Photoresistor to detect light intensity. The Photoresistor and its electronic symbol are as follows.



The circuit below is used to detect the change of a Photoresistor’s resistance value:



In the above circuit, when a Photoresistor's resistance value changes due to a change in light intensity, the voltage between the Photoresistor and Resistor R1 will also change. Therefore, the intensity of the light can be obtained by measuring this voltage.

➢ Thermistor

A Thermistor is a temperature sensitive resistor. When it senses a change in temperature, the resistance of the Thermistor will change. We can take advantage of this characteristic by using a Thermistor to detect temperature intensity. A Thermistor and its electronic symbol are shown below.

➢ Thermistor - Wikipedia

The relationship between resistance value and temperature of a thermistor is:

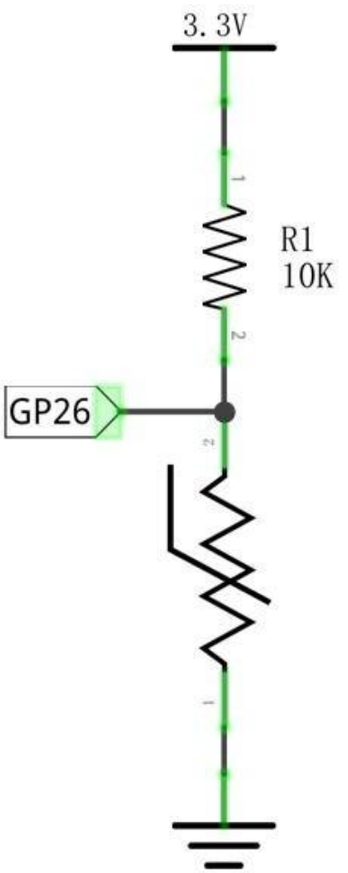
$$Rt = R * EXP[\ B * (\frac{1}{T2} - \frac{1}{T1})\ ]$$

Where:

- **Rt** is the thermistor resistance under T2 temperature;
- **R** is the nominal resistance of thermistor under T1 temperature;
- **EXP[n]** is nth power of e;
- **B** is for thermal index;
- **T1**, **T2** is Kelvin temperature (absolute temperature). Kelvin temperature=273.15 + Celsius temperature.

For the parameters of the Thermistor, we use: B=3950, R=10kΩ, T1=25℃.

The circuit connection method of the Thermistor is similar to photoresistor, as the following:



We can use the value measured by the ADC converter to obtain the resistance value of Thermistor, and then we can use the formula to obtain the temperature value.

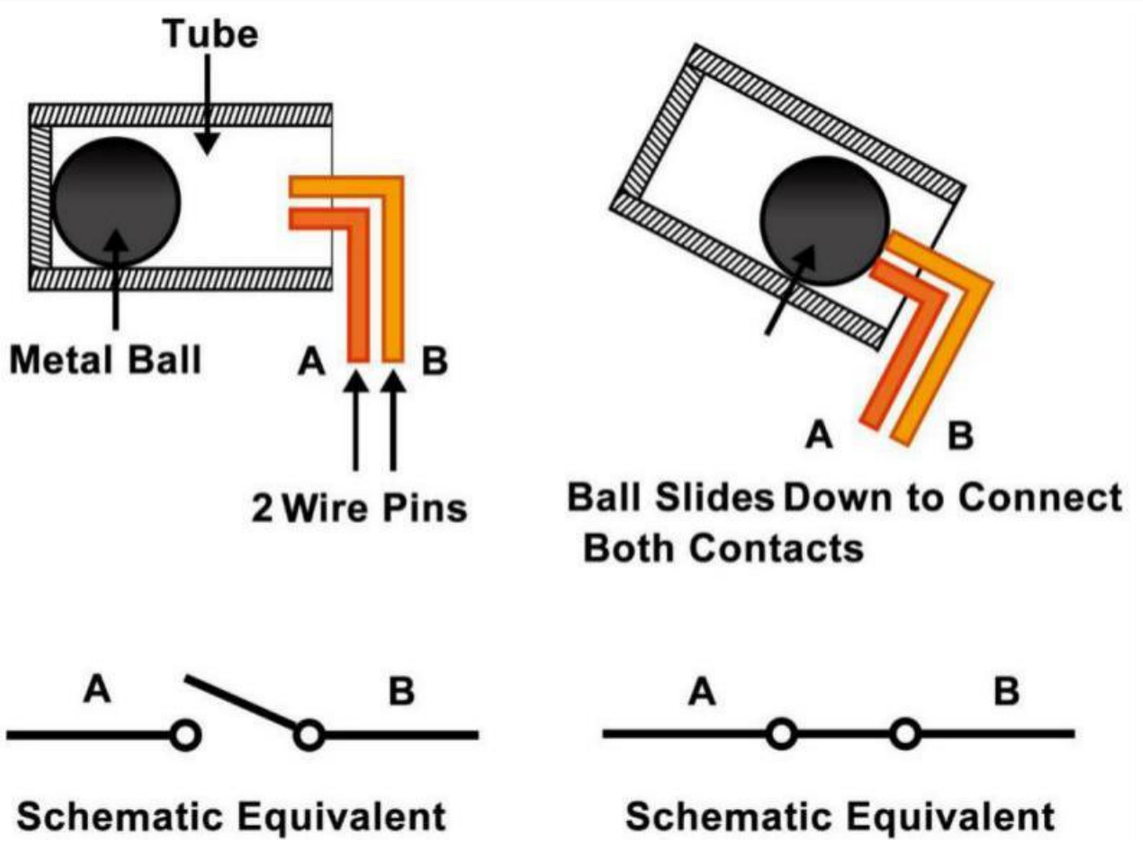
Therefore, the temperature formula can be derived as:

$$T2 = 1/(\frac{1}{T1} + \ln(\frac{Rt}{R})/B)$$

➢ Tilt Switch

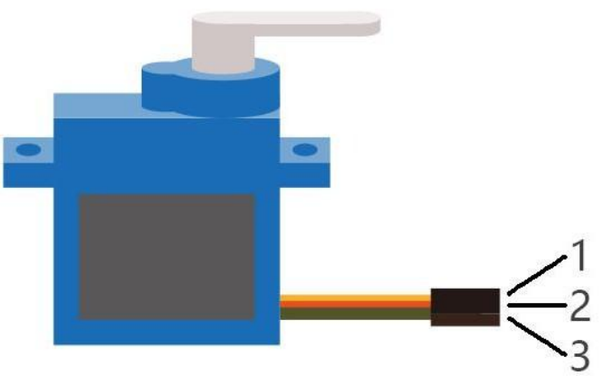
The tilt switch used here is a ball one with a metal ball inside. It is used to detect inclinations of a small angle.

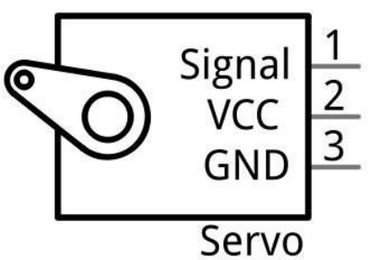
The principle is very simple. When the switch is tilted in a certain angle, the ball inside rolls down and touches the two contacts connected to the pins outside, thus triggering circuits. Otherwise the ball will stay away from the contacts, thus breaking the circuits.



➢ Servo

Servo is a compact package which consists of a DC Motor, a set of reduction gears to provide torque, a sensor and control circuit board. Most Servos only have a 180-degree range of motion via their “horn”. Servos can output higher torque than a simple DC Motor alone and they are widely used to control motion in model cars,model airplanes, robots, etc. Servos have three wire leads which usually terminate to a male or female 3-pin plug. Two leads are for electric power: Positive (2-VCC, Red wire), Negative (3-GND, Brown wire), and the signal line (1-Signal, Orange wire) as represented in the Servo provided in your Kit.





We will use a 50Hz PWM signal with a duty cycle in a certain range to drive the Servo. The time interval of 0.5ms-2.5ms of PWM single cycle high level corresponds to the Servo angle 0 degrees - 180 degrees linearly.

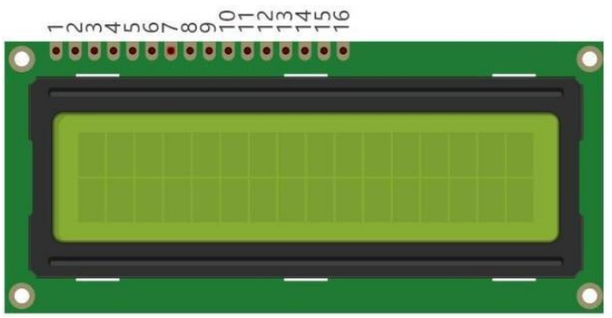
Part of the corresponding values are as follows:

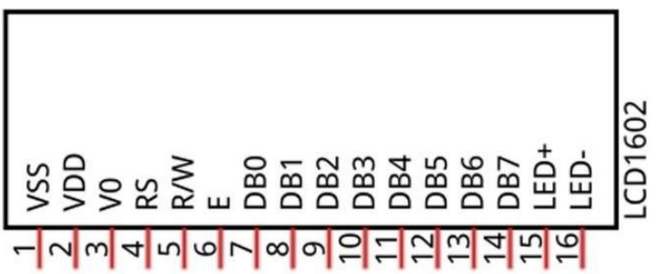
High level time	Servo angle
0.5ms	0 degree
1ms	45 degree
1.5ms	0 degree
2ms	45 degree
2.5ms	180 degree

When you change the Servo signal value, the Servo will rotate to the designated angle.

➢ I2C LCD1602

The LCD1602 Display Screen can display 2 lines of characters in 16 columns. It is capable of displaying numbers,letters, symbols, ASCII code and so on. As shown below is a monochrome LCD1602 Display Screen along with its circuit pin diagram.

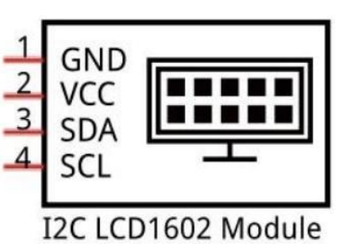




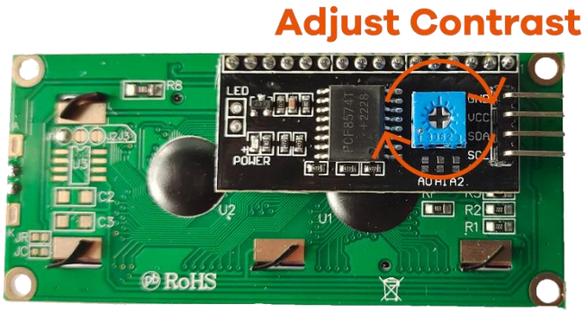
As we all know, though LCD and some other displays greatly enrich the man-machine interaction, they share a common weakness. When they are connected to a controller, multiple IOs will be occupied of the controller which has no so many outer ports. Also it restricts other functions of the controller. Therefore, LCD1602 with an I2C bus is developed to solve the problem.

➢ Inter-Integrated Circuit - Wikipedia

I2C LCD1602 Display Screen integrates an I2C interface, which connects the serial-input & parallel-output module to the LCD1602 Display Screen. This allows us to use only 4 lines to the operate the LCD1602.



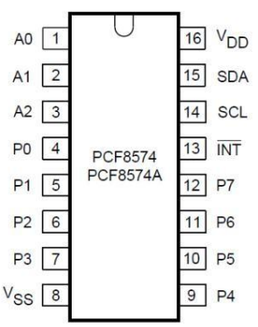
The serial-to-parallel IC chip used in this module is PCF8574T (PCF8574AT), and its default I2C address is 0x27(0x3F).



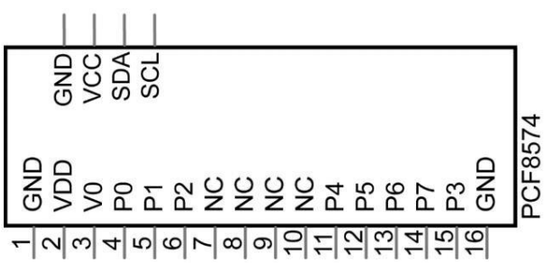
Potentiometer: It is used to adjust the contrast (the clarity of the displayed text), which is increased in the clockwise direction and decreased in the counterclockwise direction.

Below is the PCF8574 pin schematic diagram and the block pin diagram:

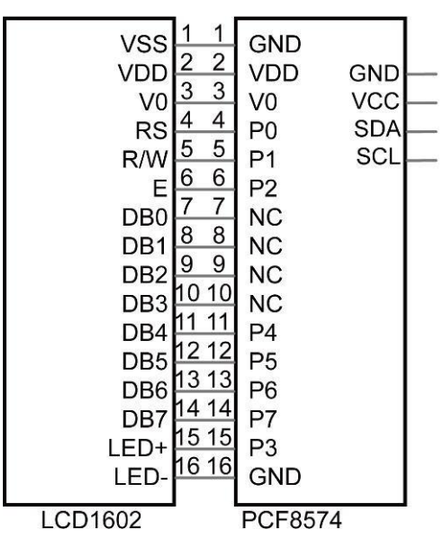
PCF8574 chip pin diagram:



PCF8574 module pin diagram



PCF8574 module pin and LCD1602 pin are corresponding to each other and connected with each other:

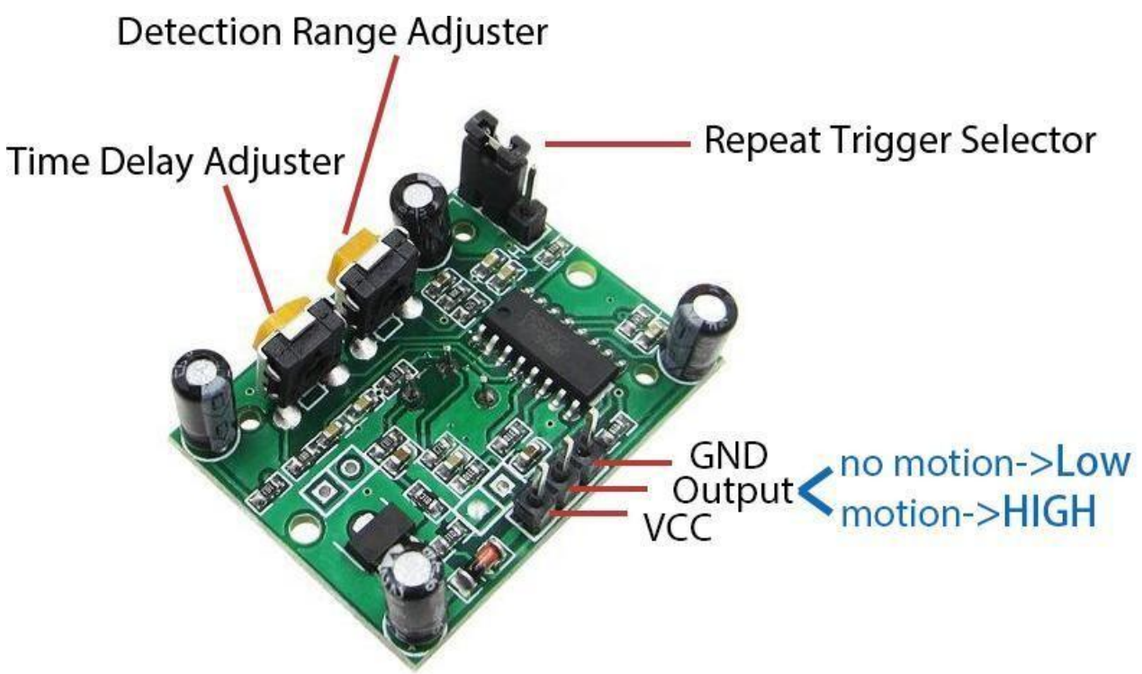


So we only need 4 pins to control the 16 pins of the LCD1602 Display Screen through the I2C interface.

we will use the I2C LCD1602 to display some static characters and dynamic variables.

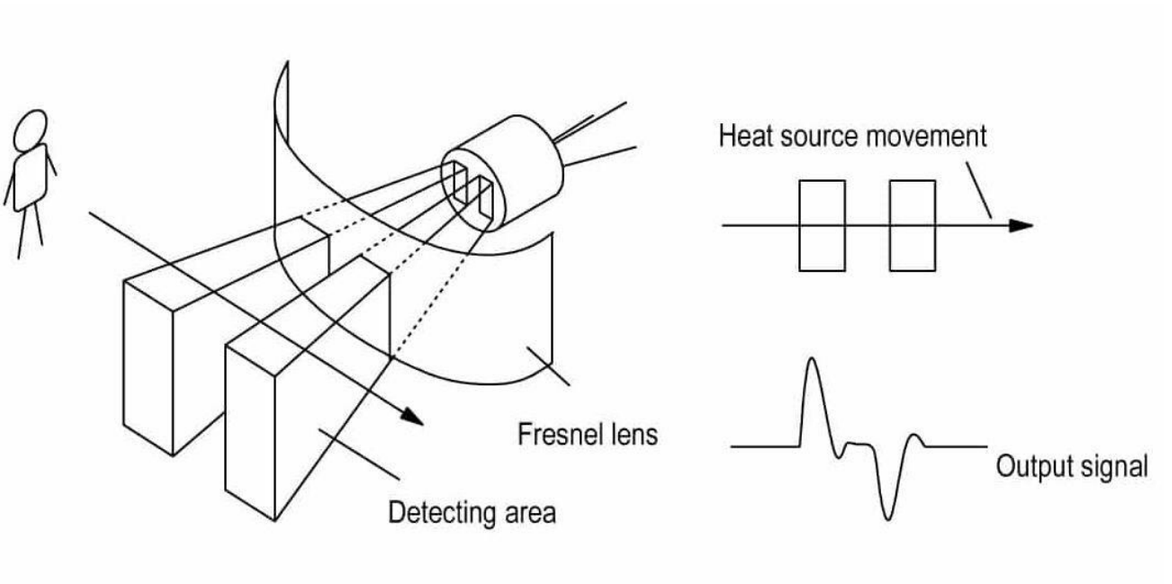
➢ PIR Motion Sensor

Passive infrared sensor (PIR sensor) is a common sensor that can measure infrared (IR) light emitted by objects in its field of view. Simply put, it will receive infrared radiation emitted from the body, thereby detecting the movement of people and other animals. More specifically, it tells the main control board that someone has entered your room.



The PIR sensor detects infrared heat radiation that can be used to detect the presence of organisms that emit infrared heat radiation.

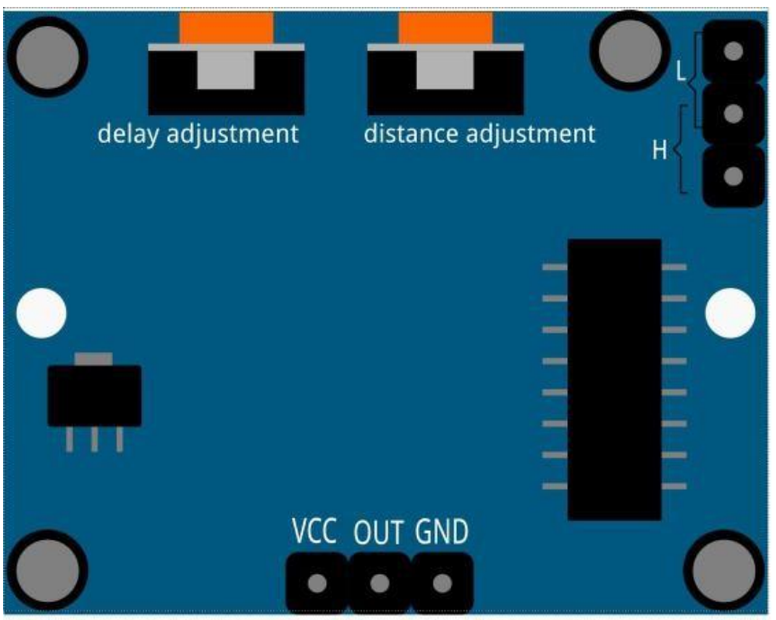
The PIR sensor is split into two slots that are connected to a differential amplifier. Whenever a stationary object is in front of the sensor, the two slots receive the same amount of radiation and the output is zero. Whenever a moving object is in front of the sensor, one of the slots receives more radiation than the other , which makes the output fluctuate high or low. This change in output voltage is a result of detection of motion.



Note:（Importance）

After the sensing module is wired, there is a one-minute initialization.After initialization,then the module will be in the standby mode.During the initialization,do not let any triggered infrared signal appear in the PIR monitoring range, including your hand. Otherwise in standby mode, it may cause false trigger detection.During the initialization, module will output for 0~3 times at intervals.This is not a real trigger result and you can ignore it until standby mode .

Please keep the interference of light source and other sources away from the surface of the module so as to avoid the misoperation caused by the interfering signal. Even you'd better use the module without too much wind, because the wind can also interfere with the sensor.



Distance Adjustment

Turning the knob of the distance adjustment potentiometer clockwise, the range of sensing distance increases, and the maximum sensing distance range is about 0-7 meters. If turn it anticlockwise, the range of sensing distance is reduced, and the minimum sensing distance range is about 0-3 meters.

Delay adjustment

Rotate the knob of the delay adjustment potentiometer clockwise, you can also see the sensing delay increasing. The maximum of the sensing delay can reach up to 300s. On the contrary, if rotate it anticlockwise, you can shorten the delay with a minimum of 1s.Usually you need to set the delay time to a minimum of 1s.

Two Trigger Modes

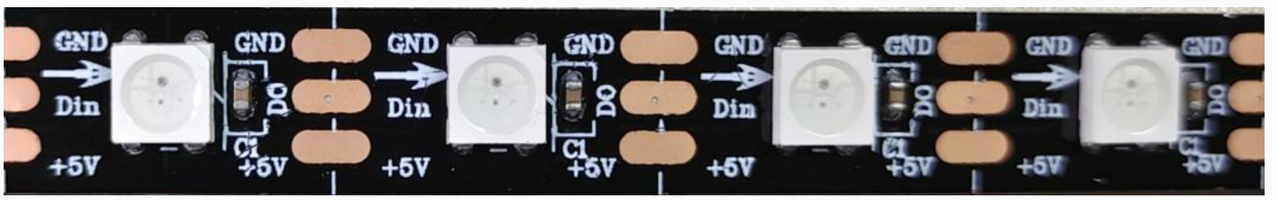
Choosing different modes by using the jumper cap.

H: Repeatable trigger mode, after sensing the human body, the module outputs high level.

During the subsequent delay period, if somebody enters the sensing range,the output will keep being the high level.

L：Non-repeatable trigger mode, outputs high level when it senses the human body. After the delay, the output will change from high level into low level automatically.

➢ WS2812 RGB 8 LEDs Strip



The WS2812 RGB 8 LEDs Strip is composed of 8 RGB LEDs. Only one pin is required to control all the LEDs. Each RGB LED has a WS2812 chip, which can be controlled independently. It can realize 256-level brightness display and complete true color display of 16,777,216 colors. At the same time, the pixel contains an intelligent digital interface data latch signal shaping amplifier drive circuit, and a signal shaping circuit is built in to effectively ensure the color height of the pixel point light Consistent.

It is flexible, can be docked, bent, and cut at will, and the back is equipped with adhesive tape, which can be fixed on the uneven surface at will, and can be installed in a narrow space.

Features

- Work Voltage: DC5V
- IC: One IC drives one RGB LED
- Consumption: 0.3w each LED
- Working Temperature: -15-50
- Color: Full color RGB
- RGB Type：5050RGB（Built-in IC WS2812B）
- Light Strip Thickness: 2mm
- Each LED can be controlled individually

WS2812B Introduction

- WS2812B Datasheet

WS2812B is a intelligent control LED light source that the control circuit and RGB chip are integrated in a package of 5050 components. It internal include intelligent digital port data latch and signal reshaping amplification drive circuit. Also include a precision internal oscillator and a 12V voltage programmable constant current control part, effectively ensuring the pixel point light color height consistent.

The data transfer protocol use single NZR communication mode. After the pixel power-on reset, the DIN port receive data from controller, the first pixel collect initial 24bit data then sent to the internal data latch, the other data which reshaping by the internal signal reshaping amplification circuit sent to the next cascade pixel through the DO port. After transmission for each pixel，the signal to reduce 24bit. pixel adopt auto reshaping transmit technology, making the pixel cascade number is not limited the signal transmission, only depend on the speed of signal transmission.

LED with low driving voltage, environmental protection and energy saving, high brightness, scattering angle is large, good consistency, low power, long life and other advantages. The control chip integrated in LED above becoming more simple circuit, small volume, convenient installation.

FAQ

Question:

Can Raspberry Pi Pico be used on Thonny, Arduino IDE and Piper Make at the same time?

Answer:

No, Raspberry Pi Pico only supports one kind of firmware installation at a time, you need to do some different operations according to the following situations.

①If you use it on Arduino IDE or Piper Make first, and now you want to use it on Thonny IDE, you need to **Burn Micropython Firmware** to your pico. Check out the "**Tutorial For MicroPython User**" folder.

②If you use it on Thonny or Piper Make first, and now you want to use it on Arduino IDE, you need to **Upload Arduino-compatible Firmware** for Pico. Check out the "**Tutorial For Arduino User**" folder.

③If you use it on Arduino IDE or Thonny first, and now you want to use it on Piper Make, you need to **Burn piper circuitpython Firmware** to your pico. Check out the "**Tutorial For Piper Make User**" folder.