www.M5Stack.com

M5STACK

A BEGINNERS GUIDE TO M5GO

Program The ESP32 With Visual Programming

This Course Is Intended For Junior High School Students
Let's Build Our First Iot Application In 60 Seconds

Shenzhen Mingzhan Information Technology Co.,Ltd
RM106, Block A, MingLiang Technology Park, Zhuguangbei Road, Nanshan, Shenzhen, China.
E-mail : Support@m5stack.com

M5Stack is a development kit for all ages and abilities.M5GO Hardware is a LEGO® compatible modular stackable system which pairs with UIflow which is an online and offline visual block coding environment. M5Stack is suitable for IoT applications, Robotics and STEM Education.

M5Stack

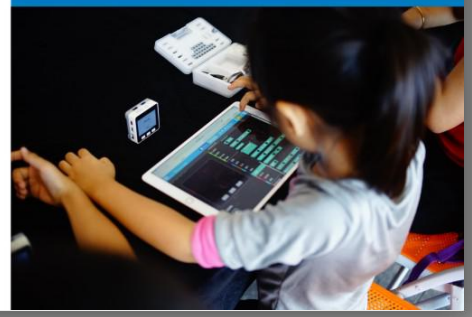
M5STACK

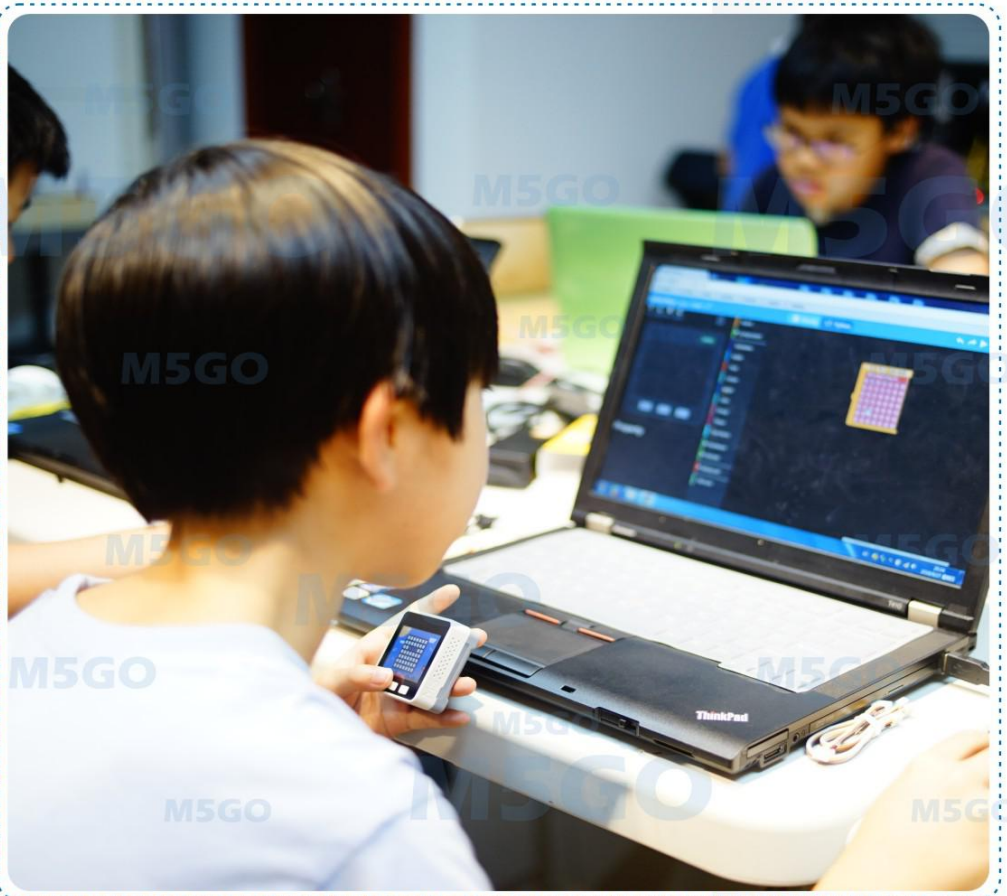
From beginner to expert
M5Stack

Inspiring a passion for coding
M5Stack

Easy for everyone
M5Stack

For the future creators





Contents

BEFORE WE START:
HARDWARE
CONSTRUCTION
UNIT INTRODUCTION
BEFORE WE START BEFORE WE
BRING IN TO USE

「BEFORE WE START」

IDEAS FAST TRACKED

Before We Start

Hardware Construction

Getting started

Unit Introduction

Introduction

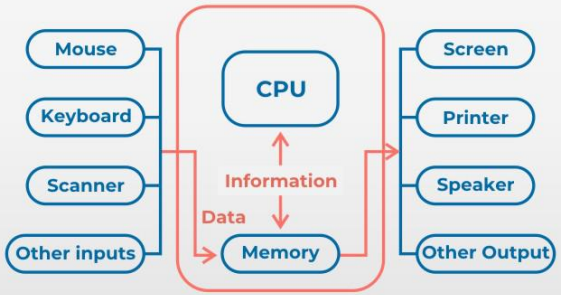
一、What is M5GO?

M5GO is a programmable microcontroller which uses the esp32. A chip chat comes with wifi and bluetooth built in. It also contains a 9-axis accelerometer, Speaker, RGB Bar and more. The M5GO is packaged in a sleek case, with 3 face buttons and 3 input/output ports. No matter weather you want to blink an LED or program more advanced functions. M5GO is designed to fit your needs.

二、M5GO is like a mini programmable computer

三、What is a computer?

4 Main elements of a computer



A computers basic elements

CPU

Inside your computer there are many chips, the Main one is called a CPU (Central Processing Unit.) It does all the calculations and communications required to make your computer run. We could liken it to the brain.

Memory

There are two kinds of memory in your computer.

1、Ram: Ram is a volatile form of memory, meaning it will be cleared when the computer is turned off. It can be read from and written too very fast and so is used for storing data related to the tasks being processed at the time.

2、Flash (ROM) Flash is used as permanent storage on a computer, and it's contents will remain unchanged even if the computer is turned off.

Inputs

These are what your computer uses to receive interaction from the physical world. Mouse, keyboard, touch screen and micro python are some examples.

Outputs

These are what the computer uses to output data to the physical world, such as a speaker, screen or printer.

四、Similarities between M5GO and a PC

CPU	✓	✓
Memory	✓	✓
Inputs	Keyboard, Mouse	Buttons, Accelerometer, Mic
Outputs	Screen, Speaker	RGB Bar, Lcd screen, Speaker

Hardware Introduction

PORT B
Port B is a standard GPIO Port which supports both analog and digital.

PORT C
Port C can handle basic input/output but is also capable of UART communication.

LCD Screen
The display of the M5GO is a 320 x 240 pixel full color screen.

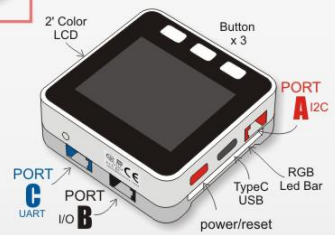
Physical Buttons
The M5GO has 3 push buttons which can be programmed to control all sorts of actions.

PORT A
Can handle basic input/output but is also capable of I2C communicatio.

Type-c USB port
This port is for charging the device and uploading codes.

Power/reset button
Single press - On/Reset

Double press - Off



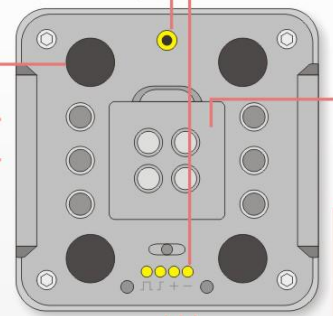
GROVE PORT	CABLE			
A I2C	GND	5V	SDA	SCL
B I/O	GND	5V	G26 In/Out	G36 Input
C UART	GND	5V	TXD	RXD

LED Bar
Both sides of the M5GO have an LED bar which consist of 5 individually addressable RGB LEDs each side.

4x Magnets
The magnets in the M5GO case allow the device to be affixed to metal surfaces, such as a fridge.

M3 Screw mounting hole
This mounting hole can be used to wall mount the device or set it on a tripod or similar device.

POGO-PIN
A pogo Pin is a spring loaded connector which is used to connect the M5GO to its charging base.



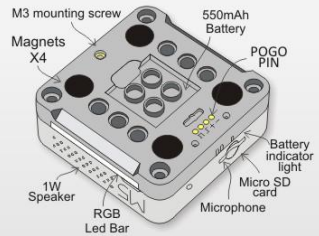
550 mah li-po battery
The powerful battery. allows you to make your projects portable.

Battery indicator light
Displays the charging status of the battery.

Microphone
The microphone can pick up sound close to the device and input it as a digital signal.

Speaker
The speaker can output sound its volume and tone can easily be programmed.

SD Card slot
When extra storage is needed we can use the SD to store pictures and sounds.



Getting Started

1、A single press of the red button turns on the device. While the M5GO is running, pressing the button again will reset the device. A swift double press will switch off the device.
2、After turning on the device you will be greeted with the main menu page. Here we have 3 options assigned to the three buttons.

A single press turns on the device

A swift double press turns the device off

Programming Mode

If you choose upload , the device will enter a page with a QR code and your API Key. You will need this in order to connect the device to the Uiflow website.

App List

Here you can find some example programs and programs that you download from Uiflow. Use the left and right arrows to navigate the list and the centre button to select. Long pressing the button on the right will give you the option to delete a program.

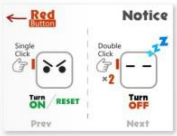
Setup

Choosing setup will allow you to configure wifi, switch to usb mode, toggle the devices startup beep or choose to connect to a previously configured wifi network.

Quick Tip

On first boot, if you don't press anything, after a few seconds the device will enter into a demo mode which will demonstrate the different capabilities of the M5GO. This may give you some ideas of what kind of projects to make with the M5GO.

3. Demo Program



Demo 1 On/Off

An explanation of how to turn the device off and on.
Single press to turn on or reset the device.
To turn off the M5GO, swiftly double press the red button.

Demo 2 Speaker

A display of the M5GO's speaker capabilities. Press the centre button to hear a sound.

Demo 3 Microphone

Try to speak into the small microphone hole on the side of the M5Stack, or tap your fingers on the table next to it. Watch the line react and display the sound waves.

Demo 4 Gyro

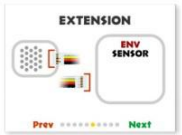
Tilt the device and notice how the ball moves around on the screen, this is possible due to the gyro or accelerometer inside.

Demo 5 RGB LEDS

See the LEDs on both sides of the M5GO glow and dim just like the lights on a police car.

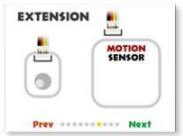
Demo 6 Port explanation

Observe the color of the different ports and their specific functions.



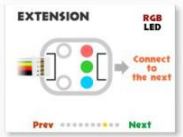
Demo 7 Environment Sensor

After connecting the ENV sensor to the correct port we can see the temperature level, humidity and air pressure.



Demo 8 PIR(Passive Infra-red)

Connect the PIR to the M5GO and notice the change of color on the circle if you hold your hand in front of the sensor or away from it.

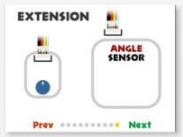


Demo 9 RGB Unit

Just like the RGB LEDs on the sides of the M5GO this unit contains programmable RGB LEDs, you can link many together to form a chain. Notice how it will only light up if we connect one of the sides of the unit. This is called polarity. Just like a battery nothing will happen unless we connect it the right way around.

Demo 10 IR Remote

Pressing the central button outputs an infrared signal. Just like your TV remote it can be used to wirelessly control things. Try pointing two M5GOs towards each other, press the button and see what happens.



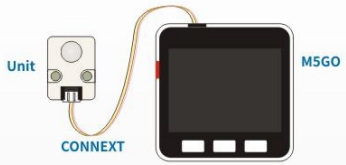
Demo 11 Angle Sensor

Twist the knob to see the RGB lights get brighter or dimmer.

Unit Introduction

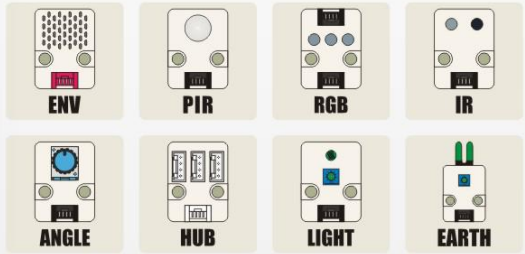
一、What is a unit?

Units are all kinds of functionality expansion modules which connect to the M5GO via connext cables. These Units can be easily programmed in Uiflow.



二、Types of Unit

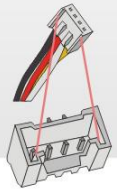
Most of the units can be separated into two distinct types. Inputs and Outputs.
Inputs are designed to take some physical stimulus such as Light, sound etc. and convert it into digital. Outputs are designed to allow the M5GO to interact with the physical world, which could be producing sound or motion via a motor and so on.



三、Connecting Units

The M5GO has 3 colour coded ports labelled a,b and c. The port of each unit is color coded to inform the user which port on the M5GO they should connect it to.

There are two ridges on the connector which help the user to plug the connector in the correct orientation.



「PREPARATORY WORK」

IDEAS FAST TRACKED

PREPARATORY WORK:
INTRODUCTION
UIFLOW PROGRAM
PROGRAMMING STRUCTURE
SAVE AND OPEN

UIFlow Program

Uiflow Introduction

Programming Structure

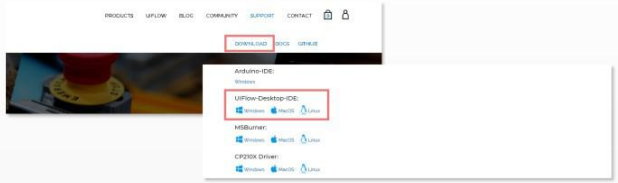
Save and Open

UIFlow IDE

UIFlow is a platform that supports blockly and micro python programming languages.

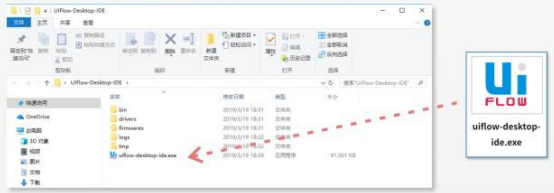
01 Download

Visit https://m5stack.com click Software > Downloads and select the correct version of the UIFlow desktop IDE for your operating system.



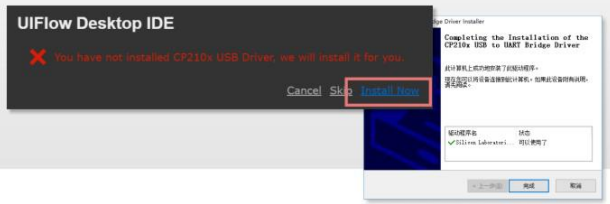
02 Unzip and Run

Extract the zip and launch the app (steps may differ depending on your operating on your OS)



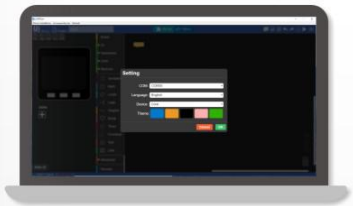
03 Install the CP210X driver

Once you run the program you will be prompted to install the drivers needed to connect your M5GO to the computer. Click "Install Now" and follow the instructions to install the driver.



04 Connect your M5GO

Once the install is complete UiFlow should automatically start. Connect your M5GO and choose your device from the drop down list. On windows it should display COM followed by a number. On Mac it will be something like /dev/tty.SLAB_USBtoUART.



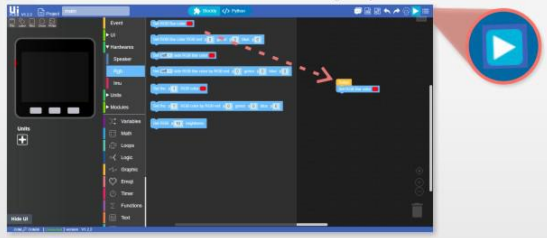
Type-C USB Cable

05 Enter USB Mode

Press the power/reset button. Once the screen lights up, press the rightmost button, scroll down to switch to usb mode with the left and right buttons and then press the centre button to select.

06 Run your first program

Select "Hardwares" from the blocks list, select RGB > set RGB bar color and drag it into the coding area and connect it up to the setup bar. Finally hit the play button to send the code to the M5GO. Both RGB Bars of the M5GO should light up.



07 Updates

M5GO has a firmware designed to work with UIFlow pre-installed. This firmware is frequently updated to add new features and fix bugs. It is important to keep the Uiflow software and firmware up to date.

UIFlow Layout

Project Title

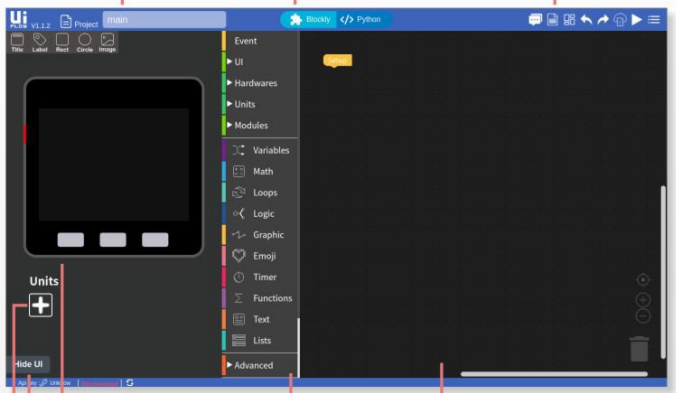
Enter your project name here. Whenever you download a program to your computer or M5GO it will retain this name.

Blockly</>Python

This button allows you to see the Python code that your blocks have generated and edit that code.

Menu Tab

From this bar you can access the forum, documentation, examples undo and redo actions upload files to the M5GO, run your code on the device and alter the settings.



UI Preview

Drag text and visual elements on to the M5GO screen to create a UI. Python code will be generated and blocks will appear in the UI panel to allow you to manipulate these elements.

Code blocks menu

Here we can find all the blocks we need to create programs.

Coding Area

The coding area is where we drag blocks to, in order to build our program.

Units

Here we can pick which Units we want to use in our project and which port we plan to connect them to.

Hide UI

Hide the Ui Manager panel to free up more space in the coding area.

Code Blocks Menu

The code block menu contains all the blocks you need to create a vast amount of programs.From Hardware blocks which directly interface with the hardware of the M5GO to maths for calculations and Logic to tie it all together. Work slowly through the basic blocks first before you move on to anything too advanced.

Event

Here you can find Loop and button press event blocks.

Hardwares

Program the internal peripherals of the M5GO such as the RGB Bar, Speaker, Accelerometer and Power management.

Units

Whenever you add a unit, it will appear hear along with all the code blocks related to it.

Math

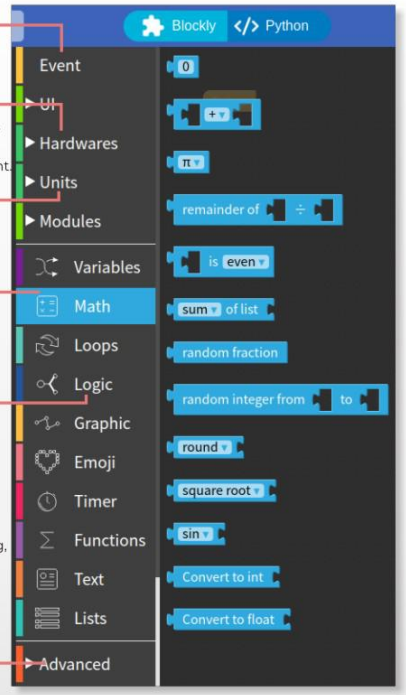
Maths is essential in programming. Here you will find all the blocks necessary to make both simple and complex calculations.

Logic

Every program needs logic to decide which action to take when an event occurs.

Advanced

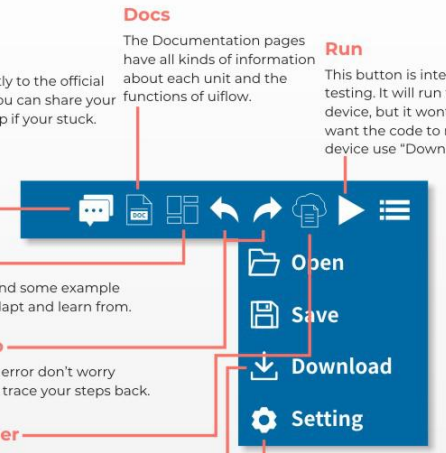
Advanced blocks for experienced coders. You'll find blocks for networking, digital/ analog pin control and more here.



The Menu Tab

A closer look at the menu tab

This Main navigation bar gives you access to the forum, documentation pages, example programs undo and redo actions, file uploader, run code on the device and alter the settings.



Forum

This links directly to the official forum where you can share your code or get help if your stuck.

Docs

The Documentation pages have all kinds of information about each unit and the functions of uiflow.

Run

This button is intended for quick testing. It will run the code on your device, but it wont save it. If you want the code to remain on the device use "Download"

Example

Here you can find some example programs to adapt and learn from.

Undo/Redo

If you made an error don't worry you can always trace your steps back.

File Manager

It's possible to upload .m5f M5GO program files and pictures to the M5GO. The manager helps us to do this.

Download

Want to download your program to the device so you can show your friends. Press downloads and then look for your program in the app list on boot.

Settings

Here you can setup the devices connection, change your language, Device and color scheme.

Coding Blocks Explained

Setup

The setup block is essential for any program to run. It defines the first thing that will happen when the code is uploaded or the device is switched on. It will only run once.

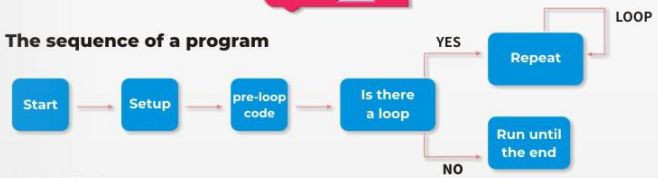
Loop

The loop block will run any code placed inside it indefinitely . That means unless you turn off the device it will continue to run without stopping.

Wait

The wait block will delay your program for however many seconds you input. Sometimes this is necessary to see the result of some code that might have otherwise run so fast that you blinked and missed it.

The sequence of a program



Code block connection

UIFlow uses blockly, a block based programming language. Blockly works a little bit like a jigsaw puzzle. When blocks are moved close together they snap into place and change color. This means they are connected and you have successfully created a chunk of code. If they don't change colour that means they are not connected and will not be run as part of the program.

UIFlow Tips

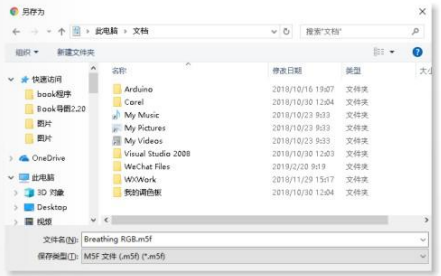
To duplicate a block, double click it with the left mouse button or right click and choose duplicate from the list.
To delete a block you can drag it back to the coding blocks menu or to the trash bin in the lower right corner.

Saving And Uploading Programs

As we move through the lessons in this course we will go on to create more complicated programs which can not be finished in a short time.Therefore we can save our programs to our computer and upload them at a later date to continue working on them. It's always a good habit to save your work in case of a crash. To do this we can click save from the drop down menu and choose where we want to save it to. When we want to upload again, simply press open and find the .m5f file that we previously downloaded. Your program will be restored from where you left off.

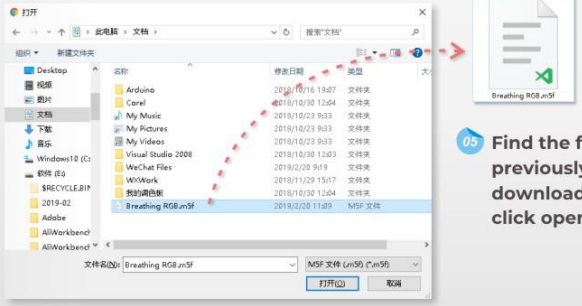
01 Open the UIFlow website. (flow.m5stack.com)

02 Once you've finished your program click "Save"



03 Your program will be saved as a .m5f file

04 When you want to open a .m5f file, simply click "Open"



05 Find the file you previously downloaded and click open

「CURRICULUM」

IDEAS FAST TRACKED

CURRICULUM: DIGITAL MUSIC BURGLAR ALARM M5GO TRAFFIC LIGHT PLANT STEWARD GAME ENGINEER

Digital Music

Burglar Alarm

Traffic Light

Game Engineer

1 Hello world - Displaying text and pictures

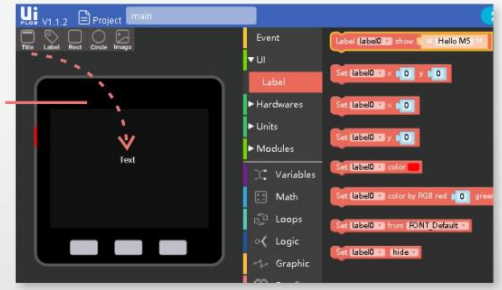
Introduction: While using UIflow you may have noticed the UI Manager on the left. This is for displaying text and graphics.
Lesson outcome: By the end of this lesson you should be familiar with the basics of creating a program and running it on the M5GO. We will display some simple text and go on to create our own personalized name card.

1-1 A programmers very first program

Hello world has long been held as the induction to programming, regardless of what language is being taught. It teaches the most basic of programming concepts and prints "Hello World" to the screen of the device. While it may sound simple, even some seasoned programmers may use this code as a test when trying a new language or device for the first time. Once we have learned the hello world program it will give you confidence to go on to learn more advanced things.

So lets get started with our "hello world" program.

01 **Start by dragging a label on to the UI Manager from the top panel.**

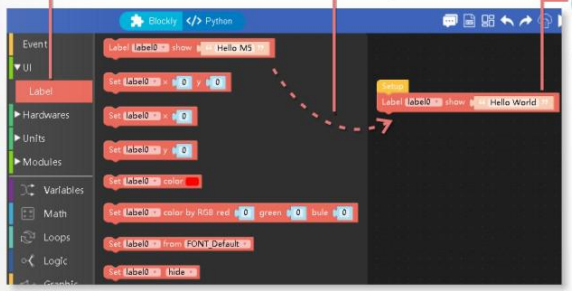


Click on the UI section of the blocks menu and you'll notice there is a sub menu entitled "label" Inside label we'll find a bunch of different blocks which can be used to program the text label. We will choose the first in the list for now "Label (label_name) show [text] Drag this block into the coding area and connect it up to the setup block. Now we can enter "hello world" inside the text field, or any other message for that matter of fact.

02 **Click on the label blocks - select label show block**

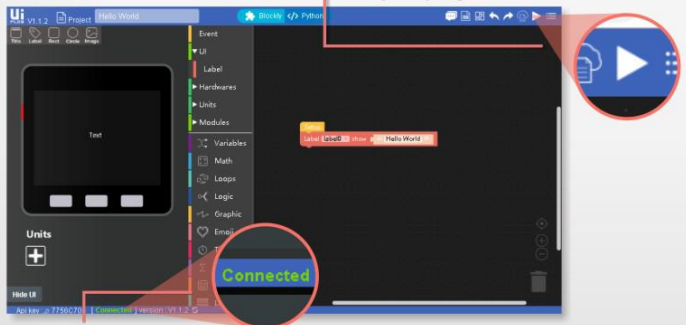
03 **Drag it into the coding area and connect it to the setup block**

04 **Change the text to "Hello world"**



Now you've created your very first program. That was easy wasn't it. Now in order for it to run on the M5GO we need to make sure that your USB cable is connected and that you have entered into USB mode as instructed in the previous chapter.

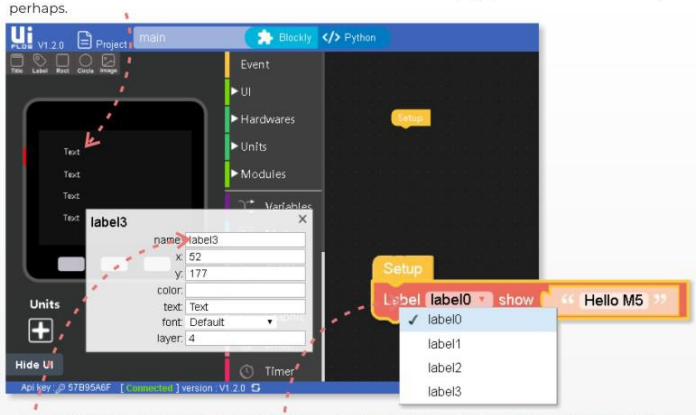
06 **Click on the play button at the top right hand corner of the window to run your program**



05 **If your M5GO Device is connected properly you should notice The word "connected" in green. If you don't see it click on the refresh Icon next to it to try to reconnect.**

1-2 Make your own digital name card

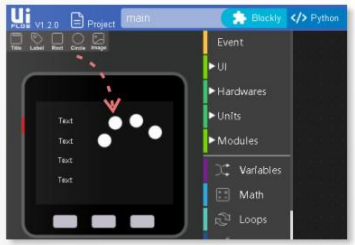
Now how about trying to add more text labels and get them to display different messages. If we want to make a name card what information will we need? Name, Age , Address and even hobby perhaps.



Notice: Each time we add a new text label it will automatically be assigned the name "label" with a higher number than the last. We can change this in properties by clicking on the text label and altering the text in the name field. In order to program the right block we need to choose this name from the drop down list of the label show block.

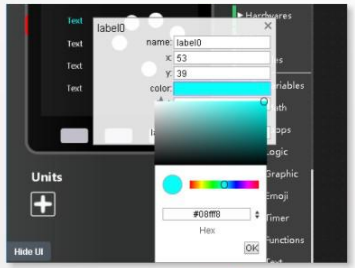
2. We can duplicate blocks by double clicking them or right clicking and selecting "duplicate"

3. Make sure to choose the correct label in the drop down menu and change the text to suit your needs.



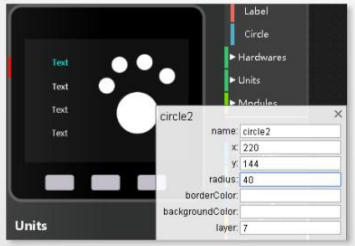
4. Adding Graphics

It may look a little boring with only text on our name card. We can soon change that. Add graphics to your name card by dragging the shape elements from the UI Manager tab on to the screen. The size and color of these elements can be changed by clicking on them and altering their properties.



5. Add a little color to your name card

We can add some color to both our text and graphics. In the properties window, simply click on the color tab. This will open the color picker window. We can select a color by sliding the color bar and then selecting the specific hue from the color swatch above. Once your happy with your color selection click ok.



6.Altering the properties of UI Elements

Theres much more we change about the appearance of our graphic elements in the properties window. Alter the size and position of text and graphics. The fill color and the border color of shapes can be customized and they can be set to different layers. Text can be beautified with different fonts and even rotated.

7. Naming your project

It's always important to name your work. You can do this by clicking in the text box above the UI Manager and entering a name for your project. Such as "My business card".

8.Downloading your programs to the M5GO

When you press the play button you are simply testing the program on the device. If you switch of the device the program will be lost. In order to store your programs on the M5GO you can press "download" from the drop down menu in the top right hand corner. An uploading message will appear on the screen of the M5GO and when it's complete it will automatically reset.

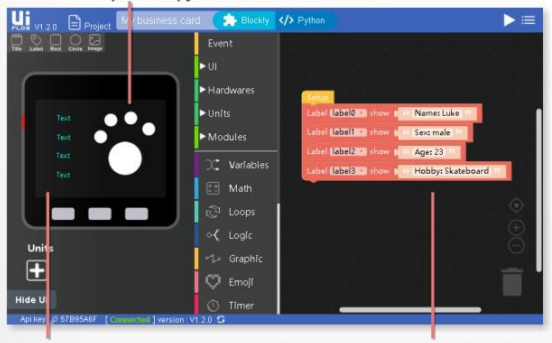
9. Stored programs

To access the programs we have downloaded you can enter the app list from the menu screen. Reset the device and press the center button to enter this list.

A glance at python

Inside each of these blocks we have an equivalent python code. Python is a popular programming language which we can view by clicking on the python | blockly switch at the top of the page. Let's compare the blocks to python.

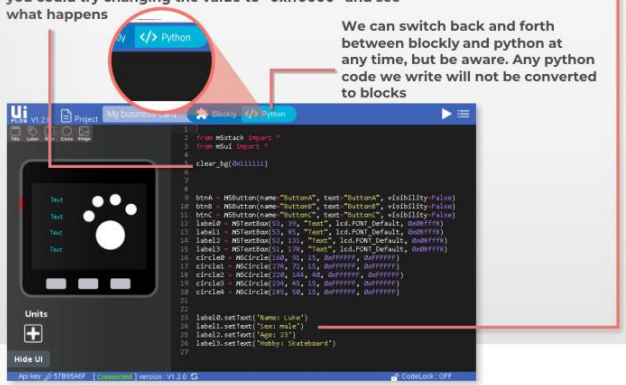
"circle" These are the circle shaped graphic elements that we dragged on to the screen. Each time we drag text or graphics the equivalent python code will be created.



The background color can be changed by clicking on it an editing in properties. In python The background color is set at the start of the program with the hex value "0x111111" We'll talk about hex later, but for now you could try changing the value to "0xff0000" and see what happens

"Label" is the text box we added to the screen. In python we can view and edit its position, font and color

We can switch back and forth between blockly and python at any time, but be aware. Any python code we write will not be converted to blocks



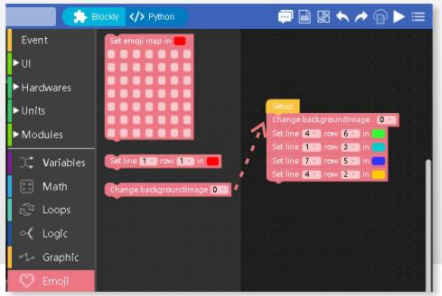
2 Animations and games with the emoji blocks

Introduction: Emoji blocks can be used to create pictures, animations and even games.
Lesson outcome: Learn to display emoji's, make animations with them and trigger them with the a,b and c buttons. Learn how the wait block is used to delay events.

1-1 Make an emoji animation

What is an emoji? usually we may think of them as smiley faces in a chat app. In this context an emoji is a grid of dots that we can use as a canvas to make such smiley faces and other pictures. In the following steps we will learn to display an emoji on the screen and link multiple emojis together to create an animation. Lets grab an emoji grid from the emoji section of the blocks menu and drag it over to the setup block. Now we can click the squares in the grid to create a picture. We can also change the color of the squares with the color selector in the top right of the block. Click the run button to see your result.

If you want to have multiple colors on the same emoji you can use the 2nd block in the emoji blocks menu. This allows you to set a single square of the grid to a specific color. The 3rd block allows you to select a background from a bunch of presets.



Now we're familiar with all of the emoji blocks, we can focus on creating an animation. What exactly is animation? An animation is essentially a series of drawings or pictures played one after another at high speed which creates the illusion of movement. The smoother the movement appears the more pictures or frames per second the animation has.

In order to create a basic animation we need at least 2 frames. Drag another emoji below the first one we created and click the run button. What happened? did you see the pictures change from the first to the second? It seems that only the second image was displayed.

1. Troubleshooting

Even though we only saw the second image, the first image was in fact also displayed. It just happened so fast that we didn't see it. This is because the M5GO is capable of running many tasks within the blink of an eye.

2. Solving the issue

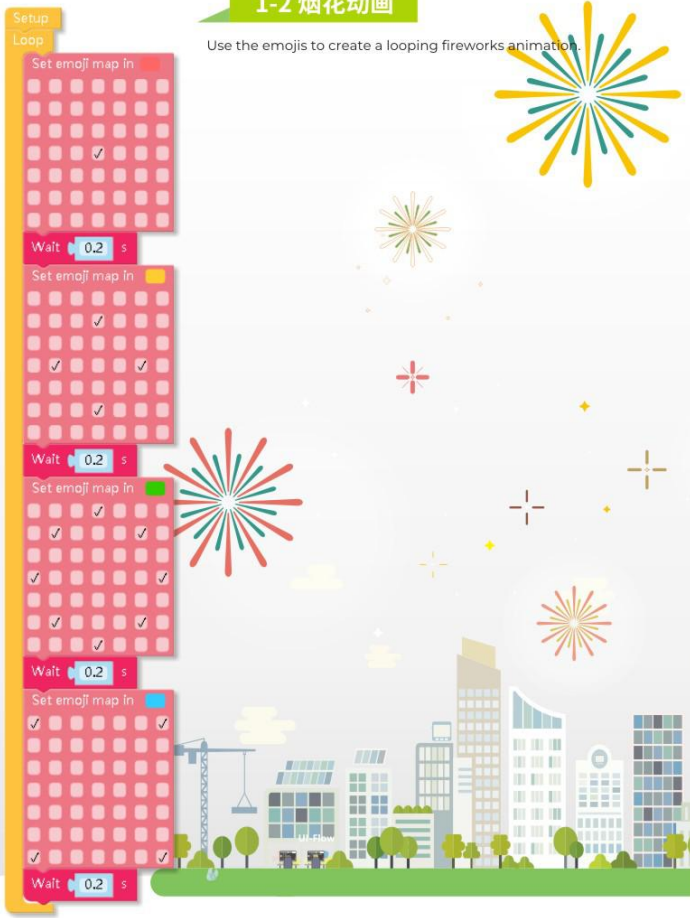
To solve this issue all we need to do is place a wait block in between the emojis to delay the change from one to the other. We can find the wait blocks in the timer section of the blocks menu. There is a wait fors (seconds) and a wait for ms (milliseconds) 1000 ms = 1 s.

Our animation only changes once which makes it very short. Often in animation the artists will loop sequences to save time, we can also do the same with the loop block. We can find it in the "Event" section. It has a bit of a different shape to other blocks that we have used. When blocks have this "C" shape it means we are meant to put code inside of them. Drag your emoji code inside the loop and run it to see what happens.

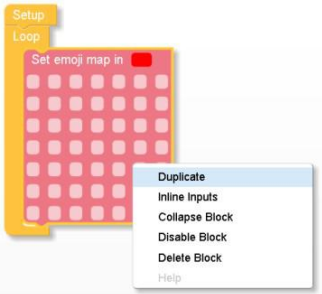
Now our animation loops over and over again, but theres an abrupt jump back to the start. To avoid this we need to place a wait block at the end of the program.

1-2 烟花动画

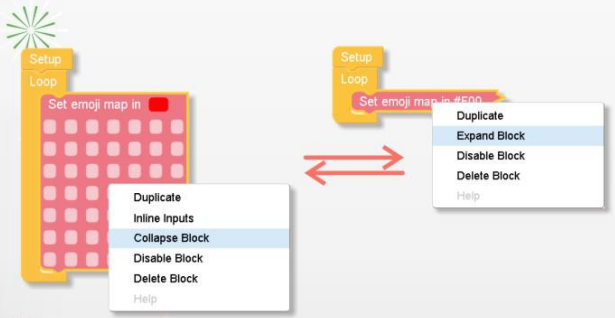
Use the emojis to create a looping fireworks animation.



Sometimes we may only want to make a slight modification to the emoji block in this case we can simply duplicate instead of drawing from scratch. Gradually make the sparks of the firework spread out.



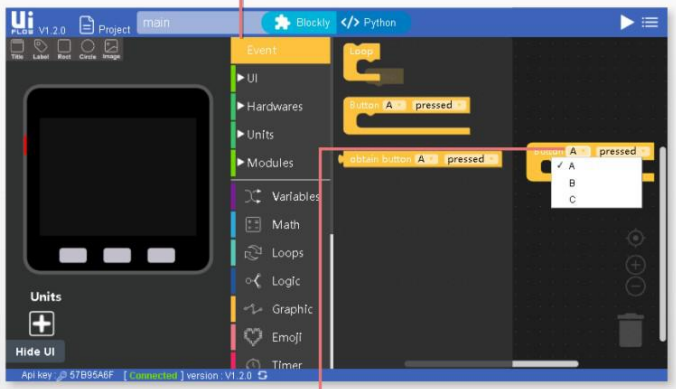
Occasionally we may find that our code takes up too much of the page and its difficult to navigate it. We can reduce the length by right clicking and selecting “Collapse block”. The code is still all there and when we need to edit it further, simply right click again and select “Expand block”.



1-3 Rock, Paper ,Scissors

We hope you had fun with the emojis and tried to make longer and more complex animation. Other than creating animations we can use the emoji blocks for other things. Such as making games. We could use the A, B and C buttons to display a different emoji when pressed, and using this method create a rock, paper, scissors game.

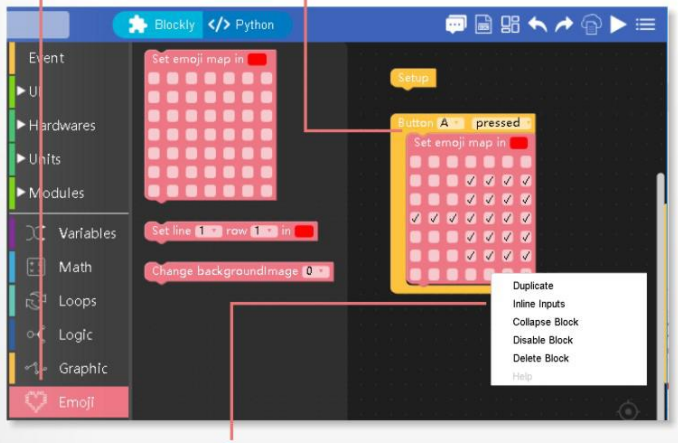
01 Click on the "Event" Section and drag out a "button A pressed" block



02 If we click on the A we can see that a drop down list appears from which we can choose A, B or C. Whenever we press the the selected button, whatever is inside this code block will run.

01 Drag an emoji grid from the emoji section

02 Click the squares to design your rock, paper and scissor blocks. You can design them however you like but make sure your friends know which one is which



03 To save time we can duplicate this block, and change the button and emoji inside it

Double check that each of the emojis are set to a different button on the M5GO.

Two can play at this game. Make sure you both have the same code on your device. 1, 2, 3 and press a button, let's see who won.

3 Digital Music

Introduction: In this chapter we will look at how to produce sound from the M5GO .
Lesson outcome: Understand the basics of music theory, Use the repeat blocks to reduce code.

1-1 Making music

Now we will start to look at how to produce sound from the M5GOs speaker, but first let's ask a question. What exactly is sound?

Sound put simply is vibration. To demonstrate this place a ruler on the edge of a table and whilst holding one side down pull the other side down and release. Did you hear the sound and feel the vibration on your fingers?

Try moving more or less of the ruler over the edge of the table. The vibration of the ruler creates distinct sounds.

In the experiment we observed the ruler moving faster or slower and producing higher or lower pitch notes. The speed of the vibrations is called frequency and is measured in Hz(Hertz) The average human ear can hear sound in the frequency range of 20 Hz to 20 kHz. If we enter any amount higher or lower than that into the speaker block, we are unlikely to hear it.

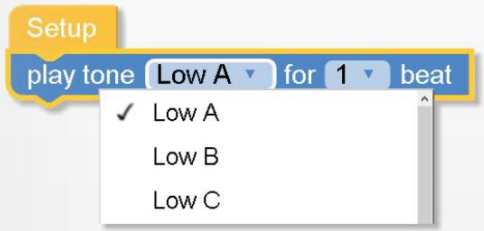
1. Tone and frequency

Now that we are familiar with a bit of the science behind sound, lets get started in using the Speaker blocks to create music. In the blocks menu, select "Hardwares" then "Speaker" then drag and connect the block up to setup. Try to alter the frequency in the first box and the beep duration in the second. Run the program and see what results you get.

Since it would be difficult to remember all the frequencies that make up a song, we invented musical notes. The letters "C-D-E-F-G-A-B" represent those different frequencies or tones. These notes are repeated up and down the piano at a higher or lower pitch. Here is a chart showing the equivalent frequency in Hz for each note.

	C	D	E	F	G	A	B
Low	131	147	165	175	196	220	247
Middle	262	294	330	349	392	440	494
High	523	587	659	698	784	880	988

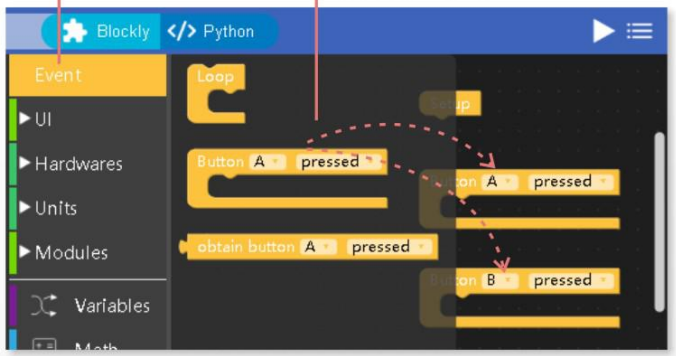
In the speaker blocks we can also find a block which allows us to create melodies from musical notes. The first drop down list allows us to select the note and the second allows us to adjust the beat or duration of the note.



Lets make a simple program to show the relation between frequency and tone.

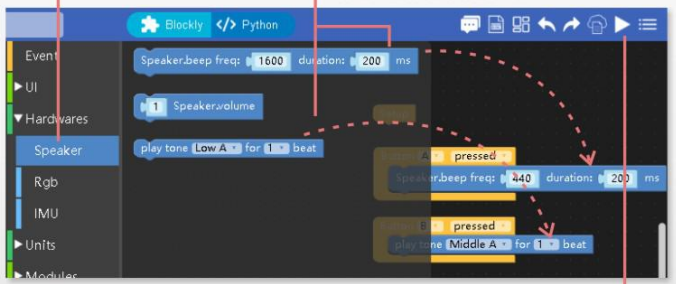
01 **Click “Event”**

02 **Drag two button pressed blocks to the coding area**



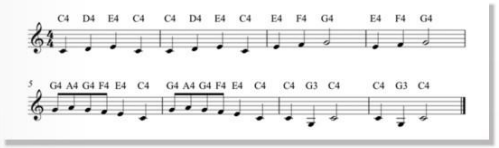
03 **Click “Speaker”**

04 **Drag a “speaker beep freq” block into the first button press block and “Play tone” into the second.**



05 **Run the program and press the A button followed by the B button. You will notice that they both produce the same sound.**

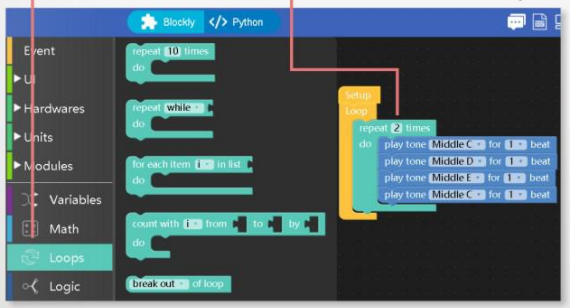
Let’s use the play tone blocks to create a simple tune.



Above we can see a piece of sheet music with the notes we need to create the song. You may have noticed theres quite a bit of repetition. In programming we want to avoid unnecessarily repeating things. Our code is bound to get longer the more complicated it gets, but in this program we can reduce its length by using the repeat blocks. A repeat block is a sort of loop that only runs for a set number of times, often referred to as an iterator. So the number we set in this block is the amount of times our code will run.

01 **Click “Loops”**

02 **Set the repeat times to 2 and add the tone blocks for the first part of the song**



Below we can compare what our code would look like with and without the repeat blocks. Both of these programs have the exact same result but adding the repeat blocks makes it neater, shorter and saves time

It's a good practice in programming to avoid typing the same things more than once. Rather we should try our best to make use of the tools we have to reduce repetition.

1-2 M5 Orchestra

Try this in your classroom or with your friends. Each person should program their M5GO to produce a different note on the press of a button. One person should be the conductor pointing to who should press their button. If you work together you could play a whole song this way.

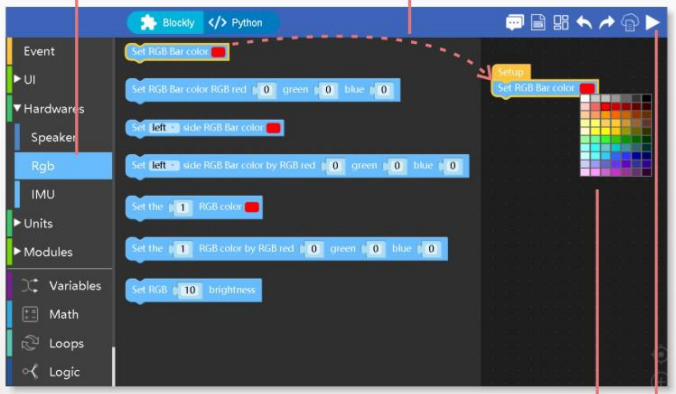
4 Amazing Light Show

Introduction: In this chapter we will focus on producing different colored lights from the RGB bars of the M5GO, and combine our knowledge to make a light show.
Lesson outcome: Learn a little about color theory, deepen your understanding of loops and delays .

1-1 Light up and turn off the RGB lights

01 Click “Hardwares” and select “RGB”

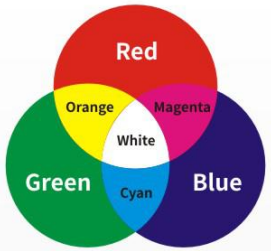
02 Drag a “set RGB bar” block out and connect it to setup



03 Click the color swatch and select your favourite color

04 Hit the run button

Apart from simply selecting a color from the color grid, we can also enter an RGB value.
What is an RGB LED and what do these values stand for
RGB Stands for “Red”, “Green” and “Blue” the additive primary colors. By combining these 3 colors in different values we can create all the colors we need.



RGB LEDs contain 3 tiny LEDS which can be set to varying intensities to represent all off the colors. They can be given a value from 0-255, 0 being unlit and 255 being the most intense level of that color. The LEDs can be turned of by setting the values to (0, 0, 0)

Apart from simultaneously turning on both RGB bars at the same time, we also have a bunch of other options. We can set all of the left side LEDs to one color or all of the right. There are 10 RGB LEDs in total, 5 on each side. We call these LED’s individually addressable as they can be switched on and off and have their color changed independent of each other we can do this with the “set the [1] RGB color” block. We can also control the brightness of the LEDs by selecting a number between 1 and 10.

1-2 Police car

In the following program we will attempt to mimic a police car with flashing lights left to right and a siren sound.

1. Simple light and sound show

Using what we have learned so far, we will start to use different RGB blocks to create a cool light and sound show.

Part 1

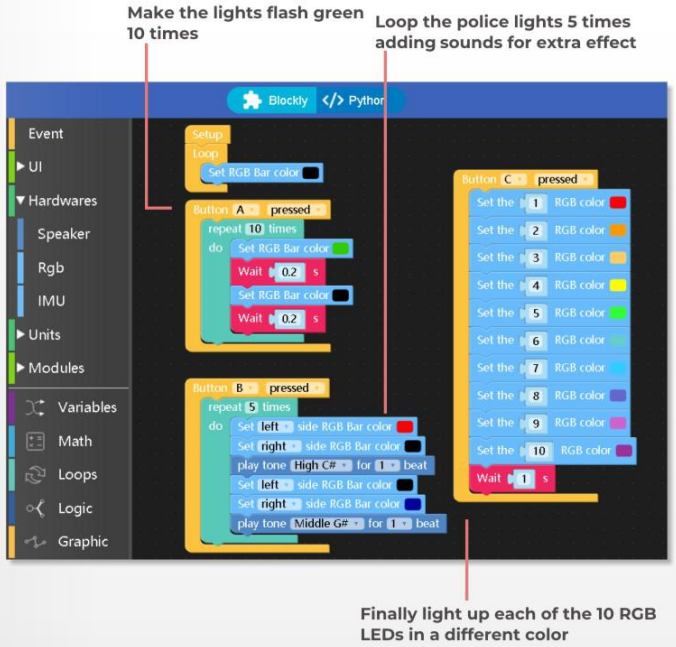
Here we turn on all of the RGB Lights at once and then quickly turn them off only using a very short delay. We use the repeat block to repeat it twice.

Part 2

In the second part we use the set left/right side RGB blocks to create an effect where the left side lights up first and then is quickly turned off and replace by the light on the right side.

2. Function expansion

Lets add some button functionlality now to control the flashing of the lights.



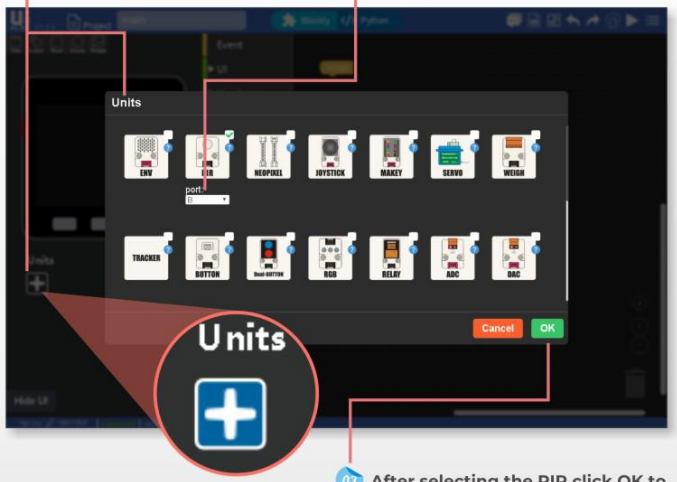
5 Burglar Alarm

Introduction: In this chapter we'll use the PIR unit to create a burglar alarm.
Lesson outcome: Learn how to connect units and what function the PIR sensor has. Try out some basic logic structures.

一、Adding Units

01 Click the plus button below Units in the UI Manager area of the screen.The Units selection window will appear

02 Select the PIR sensor by clicking on it, a green tick will appear in its corner. Below you can select which port you want to attach it to.

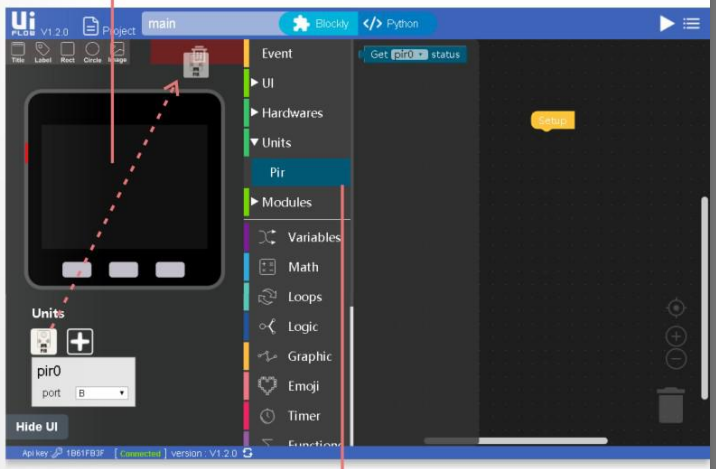


03 After selecting the PIR click OK to confirm the selection and add the unit to your program.

Each unit is different and has different parameters. Feel free to explore the other units and see what options they have.

2. Removing Units

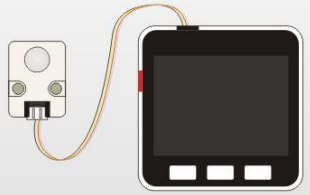
If we accidentally added a unit we didn't need or don't want to use a unit any more we can click on its picture and a trash bin will appear above the Ui Manager. We can drag the unwanted unit into the trash bin to get rid of it.



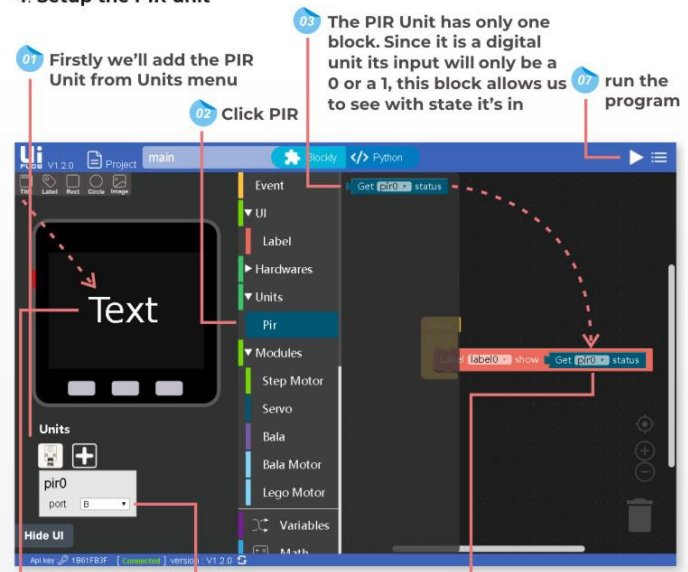
3. Using Units

Once we have added a unit we will notice that clicking on the "Units" tab in the blocks menu will reveal the PIR unit. Only once we have added a unit will it appear here. Now once we click on PIR we can see the blocks associated to it. The blocks for each specific unit will vary quite a lot.

When using a unit, we need to make sure we connect it to the same port as we chose in the unit selection window. After programming the functionality of the unit as long as there are no connection problems we will see the desired result.



4. Setup the PIR unit



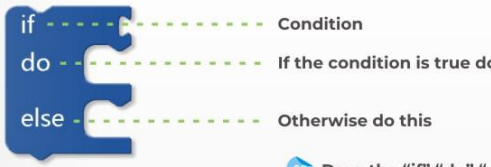
1. Testing the PIR unit

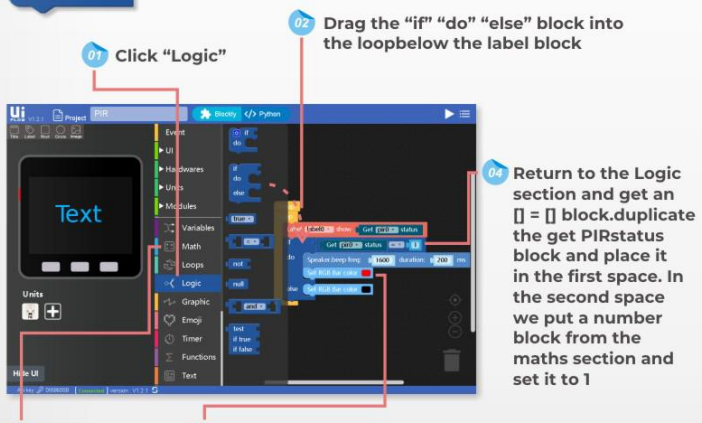
Once setup, unless there is a person or other living thing in front of the PIR the screen will display 0. If we wave our hand in front of the PIR it will change to 1. The PIR has a slight delay before it resets back to 0.

How does the PIR Unit sense there is somebody in front of it?
Security lights outside your house also use a PIR sensor and will switch on if a person or cat walks by. This is because our bodies emit infra-red light and the PIR can sense this.

1-1 Burglar Alarm

Now that we have established how the PIR sensor works, why don't we put it to good use. Since it can sense if people are near wouldn't it make a good burglar alarm. Now all we need to do is make a system to alert us. We have already programmed the lights and speaker of the M5GO so why don't we use those. All we need to do is to combine these elements together with the "if" blocks. This block allows us to steer the program in different directions depending on what input we receive.





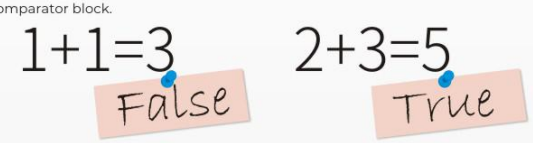
After connecting the PIR unit to the M5GO, click Run and see the result.

6 Plant Monitoring System

Introduction: In this chapter we will use the Earth unit, a soil moisture sensor to check the water level.
Lesson outcome: Get a better understanding of logic, comparators and booleans.

1. Booleans

What is a boolean? In the previous chapter we used the "if block" and the equals to block. We checked that if the PIR value was equal to 1 then do something. With booleans we can do just the same but boolean only has two options "true" and "false" However what equals true can be defined by the programmer. For example 2+3 = 5 is True and any other response the program receives is False. We can use this logic to steer the outcome of the program which is referred to as "control flow" In the logic section we can find blocks for true and false, and they can be attached to the if condition or in a comparator block.



2. Boolean Expressions

Building on this boolean concept, sometimes we may want to check multiple conditions and if one is true then allow the program to flow in a different direction. Boolean expressions consists of "and", "or" and "not". Let's have a look below at how we might use them.

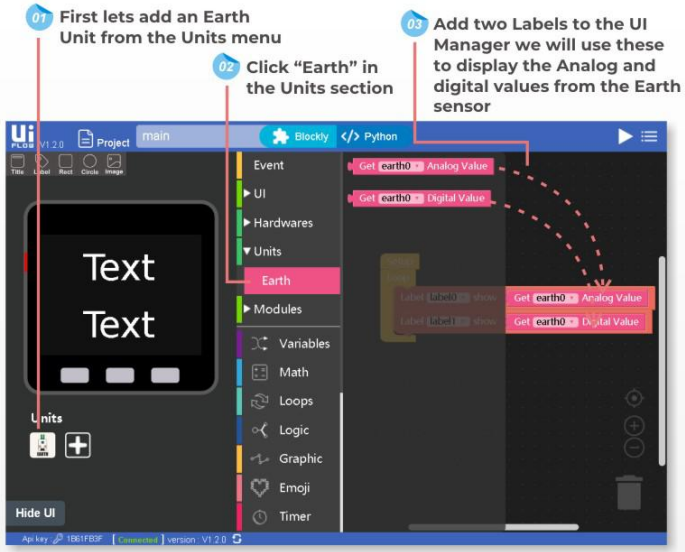
Imagine you are in a restaurant and ask for coke with ice (If "coke" and "ice" received = True) If the waiter brought you only coke with no ice then the condition is False because both conditions have to be fulfilled in order for the program to return "True"

Now lets imagine your not too bothered what you are going to drink, and you ask the waiter for coke or juice if coke isn't available (If "coke" or "juice" received = True) The waiter only has to bring you one of these two options for the condition to return "True" if the waiter brings you neither then it would return "False".

Finally imagine your allergic to coke and you ask the waiter to bring you anything other than coke (if not received "coke" = True) In this case you may be offered many options and they will all be "True" the condition will only return "False" if you receive the coke.

3. Testing the Earth sensor

The Earth Unit is a sensor which can check the level of moisture in the soil.



We need to connect the Earth sensor to port B and select Port B in the units menu.

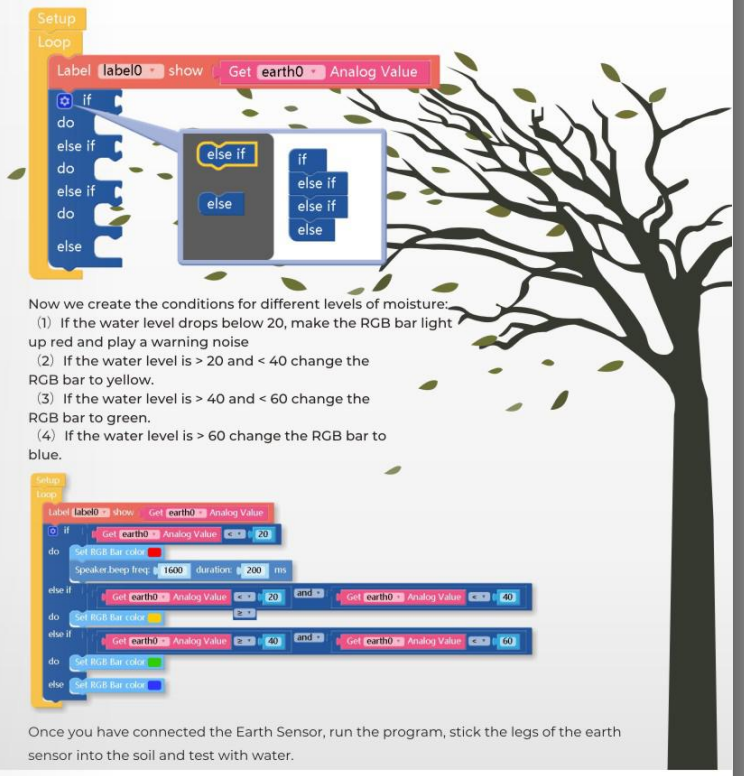
After we have added the Earth Unit we can see two blocks "Get Digital Value" we talked about digital values when dealing with the PIR sensor, and we know it can only input 0 or 1.
The following block "Get Analog Value" however gets the Analog Input. Analog is different from digital in that we get a range of values rather than two states. Analog will input a number in the range of 0 - 1023. We could liken it to a thermometer which will give a different reading based on the current temperature of an object.

How does the Earth Unit check the moisture level?

As you may know water is a conductor of electricity. The more water in the soil the more conductive it will be and vice versa. The two legs of the Earth Unit measure the electrical resistance between themselves and input it as a number.

1-1 Plant Monitoring System

In order to represent different levels of soil moisture we may need to make a test first, using our Earth unit and the simple program we just made and then noting down the values when the plant is dry, has a moderate amount of water a sufficient amount or too much. We can then use these values and the "else if" conditions to create four different outcomes.



Now we create the conditions for different levels of moisture:
(1) If the water level drops below 20, make the RGB bar light up red and play a warning noise
(2) If the water level is > 20 and < 40 change the RGB bar to yellow.
(3) If the water level is > 40 and < 60 change the RGB bar to green.
(4) If the water level is > 60 change the RGB bar to blue.

Once you have connected the Earth Sensor, run the program, stick the legs of the earth sensor into the soil and test with water.

7 Weather Station

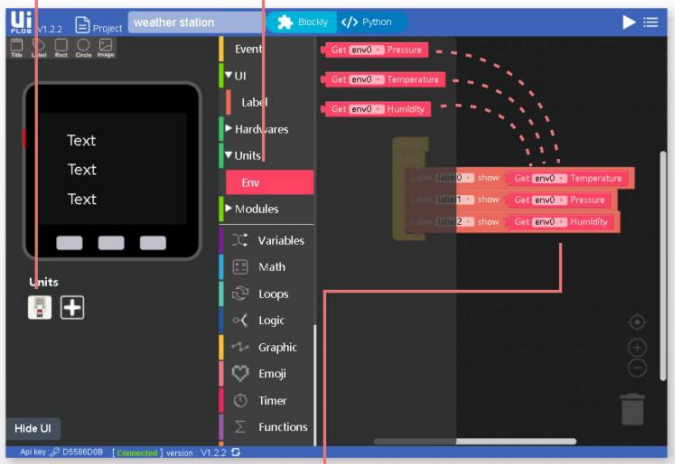
Introduction: In this chapter we will use the ENV unit to get the temperature, moisture and air pressure, display them and use them to control various outputs.
Lesson outcome: Learn what variables are and what we use them for, and how we can use more complex logic in our programs.

1. Testing the ENV Unit

The ENV can sense temperature, moisture and air pressure.

01 Add an Earth Unit from the Units menu

02 Select ENV from the blocks menu



03 Add 3 labels and add all three blocks from the ENV section to their show section

2. Temperature, Moisture, Air Pressure

If we test the program we just made we will see 3 values on the screen, which are the sensors input for the current temperature, moisture and air pressure.

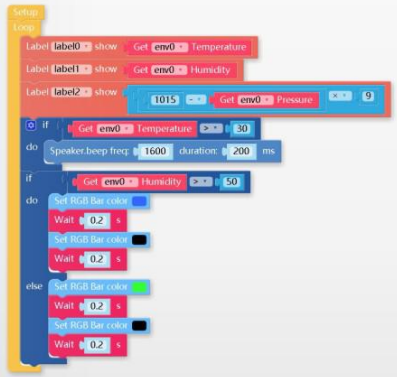
You may be familiar with temperature and moisture but what is air pressure?
it is the force exerted on a surface by the air above it as gravity pulls it to Earth. It is usually measured in pa(pascal) or psi (pressure per square inch)

The ENV unit gives us a readout of hpa(Hectopascal 100 hPa = 1 Pa) Based on the fact that for every 9 Metres higher than sea level we go the air pressure will drop by 1hPA Keeping the same point of reference we can roughly estimate our altitude.

Suppose we test that the current ground pressure is 1015hpa. Based on this value we only need to deduct the pressure value from the sensor and then multiply the result by 9 to get our current altitude.

1-1 Weather Station

(1) If the temperature is greater than 30 degrees make the speaker beep.
(2) If the humidity is greater than 50 make the RGB bar flash blue, else make it flash green
(3) Use the labels to display the current temperature, moisture and altitude
(4) Calculate the altitude by deducting the air pressure value from the estimated pressure x9



Once you have connected the ENV sensor and uploaded the program to the M5GO. You can test it by blowing on the ENV and seeing if the values change. You could always climb a hill with the M5GO and see if the pressure changes.

8 DIY Watch

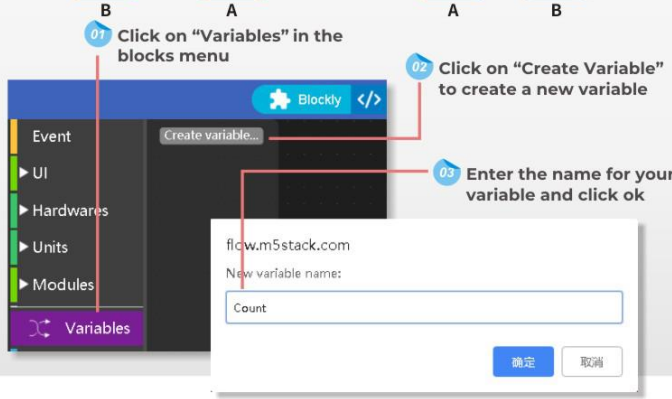
Introduction: In this chapter we will combine all we previously learnt about variables and logic to create a watch program.
Lesson outcome: Gain a deeper understanding of variables and how nested conditions work.

1-1 Understanding Variables

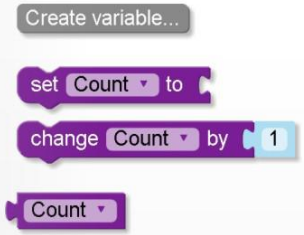
(1) We can describe variables in a simple way by using the example of a jar. First we give the variable a name e.g "A" and then we can either store a number or a bunch of letters in that variable.

(2) When we assign a new value to the variable the original content will be overwritten.

(3) We can also take the content of one variable and assign it to another variable, which would just be like taking the contents of the first bottle, making a copy of it and then placing it in the second bottle.



After creating a variable a block representing it will appear in the variables section.



1. Counter Program

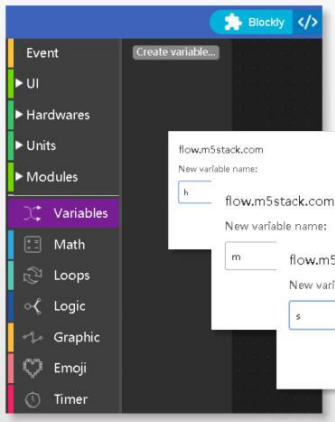
Here we'll create a variable called count, assign 0 to it and connect it to the setup block. Then we'll display that variables contents on the screen with a label block, and add 1 to it for every second passed.

Now we have a counter that goes up every second. In order to create the watch we need to remember that 60 seconds = 1 minute, 60 minutes = 1 hour and 24 hours = a day.

1-2 DIY Watch

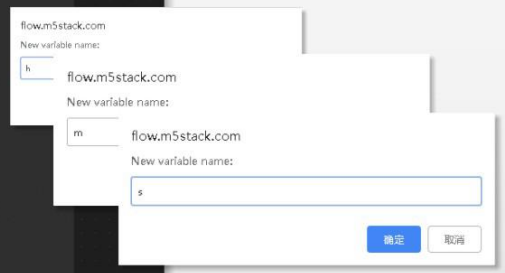
1. Creating the display

First we need to add 3 labels for the seconds, minutes and hours.
We can first add 0's to the labels as a place holder and then adjust the font and color until we are happy with the result. Change the label names to h, m and s for hour minute and second. This will make sure we don't get confused.Now add 2 more labels between the numbers and enter colons these will separate the numbers.

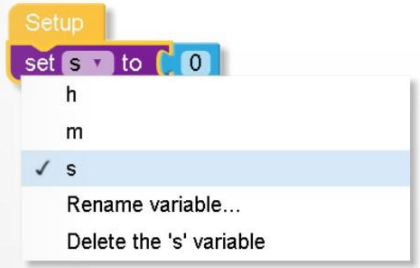


2. Start to code

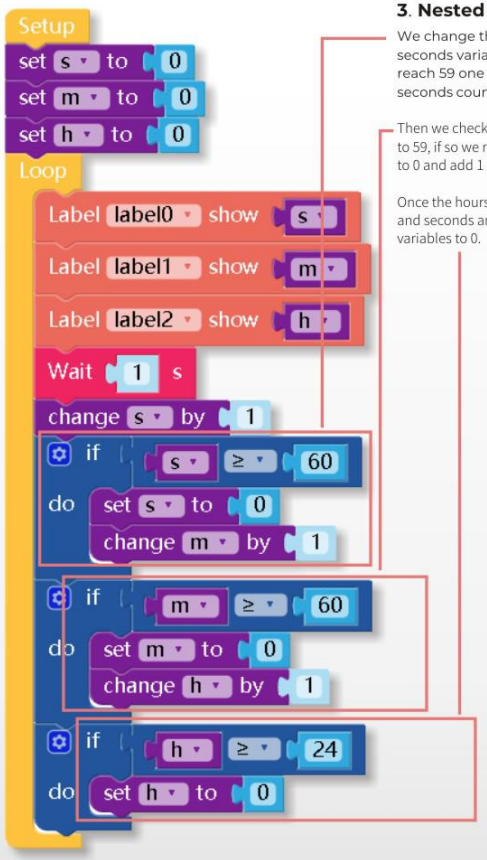
Create 3 variables called h, m and s to store the hours, minutes and seconds.



Drag a loop in and a set variable block and connect it to setup, choose s from the drop down list and enter 0 as the value. We can repeat this process for the minutes and hours.



(1) Put all of the set value blocks in the loop block and add another loop block below them make sure you set all of their values to 0.
(2) Add 3 Label show blocks and set each of the h, m and s variables to the corresponding label.
(3) Add a change s by block set it's value to 1 and add a wait for 1 s block below it.
(4) Since the loop already has an unseen delay value it won't be entirely accurate.



3. Nested Logic

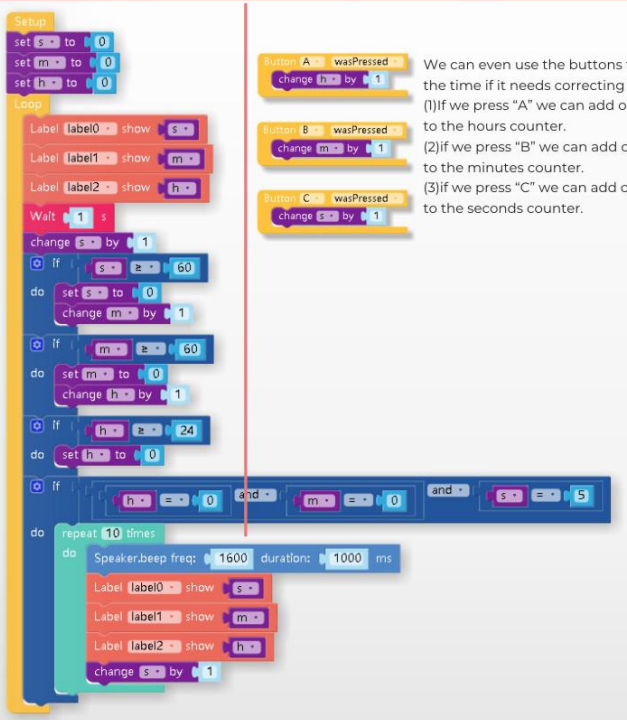
We change the minutes based on the seconds variable. Once the seconds reach 59 one minute is added and the seconds counter reset to 0.

Then we check if the minutes are equal to 59, if so we reset the minute counter to 0 and add 1 to the hours variable.

Once the hours reach 24 and the minutes and seconds are all at 59 we reset all variables to 0.

4. Function expansion

We can expand the function of this program by adding an alarm clock function.
We can add an extra condition in the program to make a noise when the time reaches one specified
We can use a repeat loop to make the speaker beep for 1000 milliseconds or 1 second for 10 times. Of course we don't want to upset the normal function of our program so we continue to change the seconds for each loop. This way the time keeps getting updated and will display properly even though we are outside of the main part of the watch program.



We can even use the buttons to adjust the time if it needs correcting
(1)If we press "A" we can add one unit to the hours counter.
(2)if we press "B" we can add one unit to the minutes counter.
(3)if we press "C" we can add one unit to the seconds counter.

9 Amazing accelerometers

Introduction: In this chapter we will take a look at how to use the accelerometer or gyro to create a dice game and a step counter.
Lesson outcome: Understand how the accelerometer works, learn a little about random numbers and probability.

1. What is an accelerometer?

An accelerometer measures an increase in acceleration of an object along a fixed line within a given time. Usually the time in which the measurements are taken is a short duration, in the scale of milli and microseconds.

For example a car accelerating along a straight road may go from 20 mph to 30 mph in the space of a second. Therefore we would say that there had been an increase of 10mph within that second.

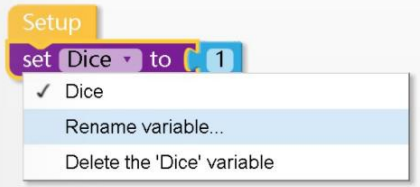
Quick question: 2 cars are sat motionless on a road, then they both accelerate to 40. One car only takes 6 seconds to reach that speed, while the other takes 10 seconds. Was their acceleration the same?

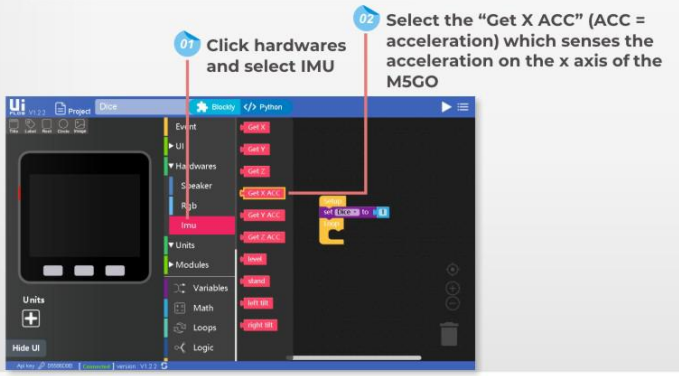
Of course not, the car that took the least amount of time to reach 40mph clearly has a greater acceleration than the car that took longer to reach the target.

Acceleration is present in all aspects of our lives, even down to a small movement of our hand from one position to another. It's acceleration changed during the course of that movement. The following program will demonstrate this in the form of a simple dice game.

Let's make a magic dice with the use of the accelerometer.

(1) Create a variable called dice to store the current value of the dice.
(2) Add a set variable to block. If we wish to change the name of the variable we can always do so by right clicking
(3) On setup we need to set the dice variables value to "1"



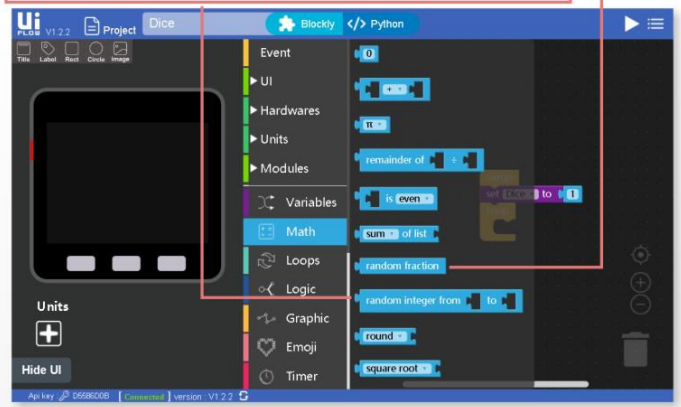


2. Random Numbers

In the maths section there are two random number blocks.

"random fraction" produces a random number with a decimal point.

"random integer from a to b" produces a whole number from the range specified.

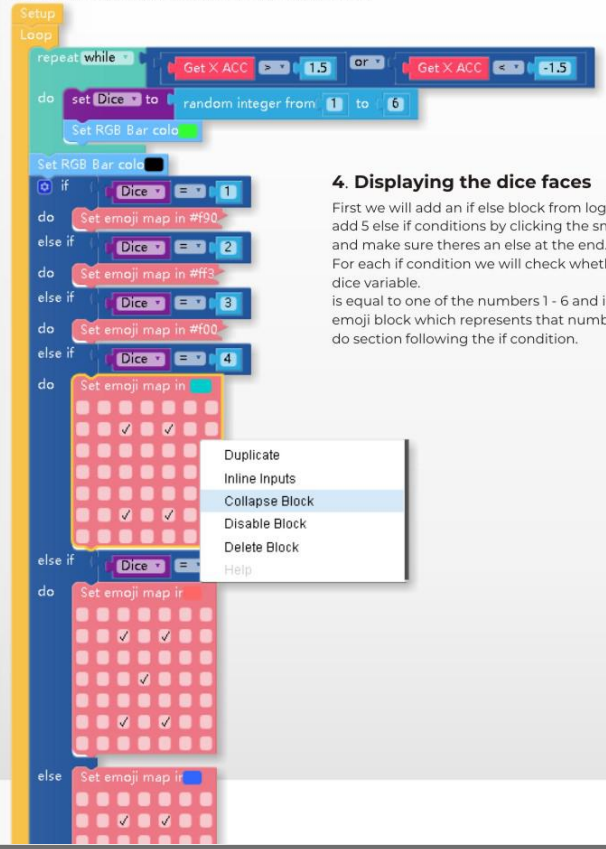


Drag a "set variable to" block and connect it to a random integer block setting the values from 1 to 6, the amount of sides a dice has.

3. While loops

In the loops section we can find the "while" loop. A while loop will repeat anything placed inside it while the condition is true, if the condition is not true this loop will be ignored.

We will use the while loop to randomly change the value of the dice variable whilst the dice is being shaken. We can use the boolean expression "or" to check whether the M5GO is being shaken by checking if the acceleration on the X axis is greater than 1.5 or less than -1.5. We can also light up the RGB bars for a nice effect. If the dice is not being shaken the dice variable will stay the same number it was when it last came to a halt. Now we need to create some if conditions based on what number is produced to show the result on the screen.



4. Displaying the dice faces

First we will add an if else block from logic, then add 5 else if conditions by clicking the small gear and make sure theres an else at the end.

For each if condition we will check whether the dice variable.

is equal to one of the numbers 1 - 6 and insert an emoji block which represents that number in the do section following the if condition.

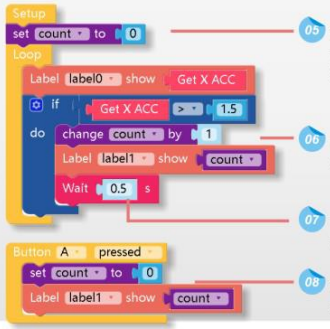
Step counter

01 Add 2 labels in the UI Manager, one to show the acceleration and another to show the amount of steps

02 Create a count variable set it to 0 and connect it to the setup block

03 The acceleration value will constantly change so we need to create a program that will keep updating the latest value

04 We must add a "label show" block and place the "Get X Acc" block inside to constantly display the ever changing acceleration value



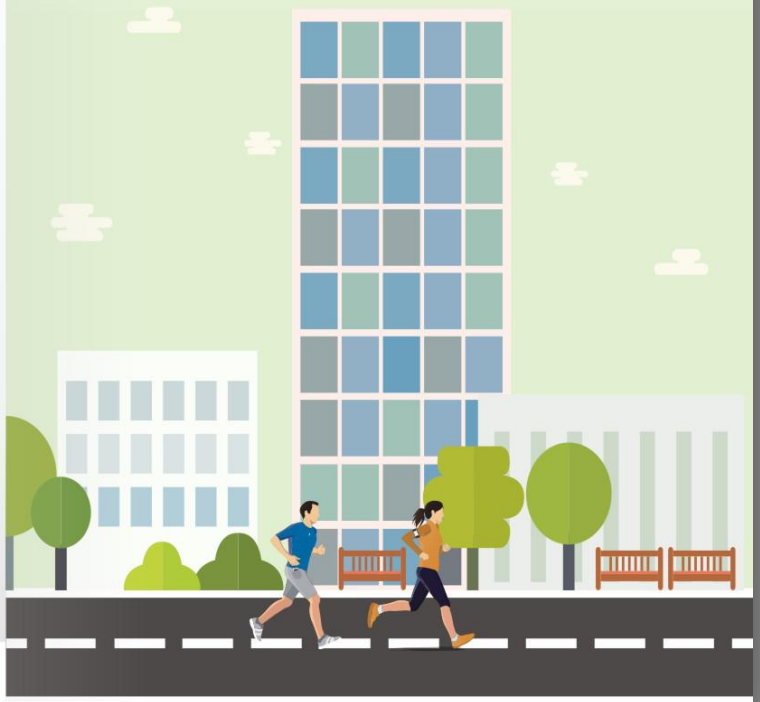
05 We need to check if the acceleration value goes above 1.5 and then change the step count by 1

06 Then we need to make sure the label shows the latest step count

07 To avoid registering too many steps we can add a 0.5 second delay

08 We can also add an A button block to reset the step count

It was necessary for us to add the delay value since our program will be executed very quickly and may have already been through several loops before we have even taken one step. If your program is registering too many step counts or not registering you can play with the delay value and the acceleration threshold value to see if you get better results.



10 Potentiometer Control

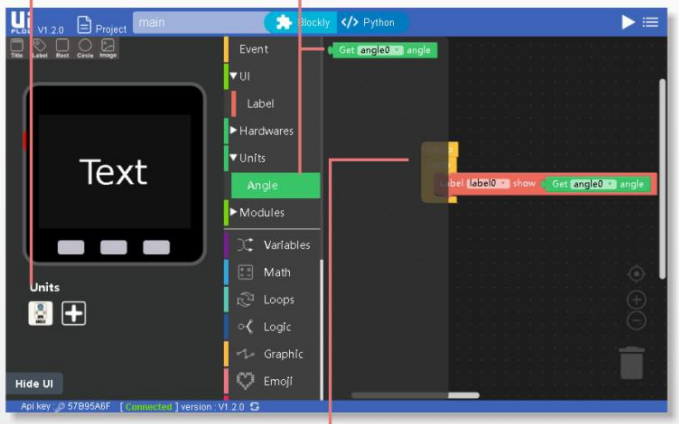
Introduction: In this chapter we will use the angle sensor (Potentiometer) to control the brightness of the RGB Bar.
Lesson outcome: Discover what a potentiometer is and what its uses are, and How we can map one value to another

1. Testing the angle sensor

The Angle sensor is able to sense the angle of the knob and input it as an analog value.

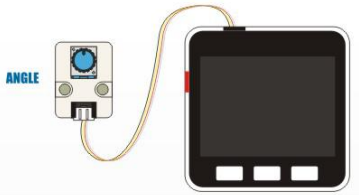
01 First we add an “angle sensor” Unit from the Units menu

02 Open the angle section within units and drag out the “Get Angle” block



03 Add a “label show” block to the loop and put the “Get Angle” block inside it

Once we have connected the “Angle” unit up to port B of the M5GO and have run the program, we should be able to see the values change between 0 - 1023 when we twist the knob.

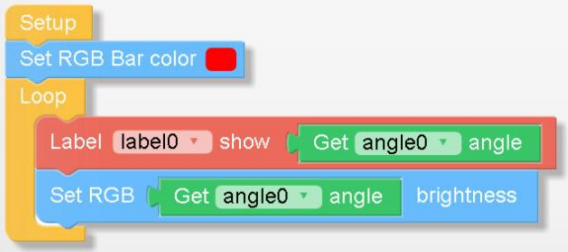


How does the angle sensor know it’s being turned?

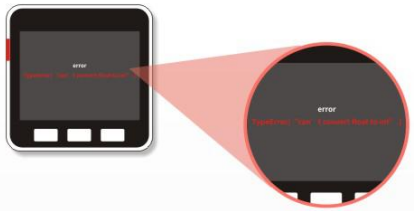
The angle sensor is also known as a “potentiometer” or a “variable resistor” because we are able to vary the resistance in the circuit. Inside it has a contact which scrapes along a track which has a range of resistance. It is supplied with 5 volts and in its left most position that voltage goes down to 0 volts. The range from 0 - 5 is read into the M5GO as a number ranging between 0- 1023 .

2. Adjustable Light

Now that we understand a little about the workings of the angle unit, we can try to use it to control the brightness of the light with the following program.



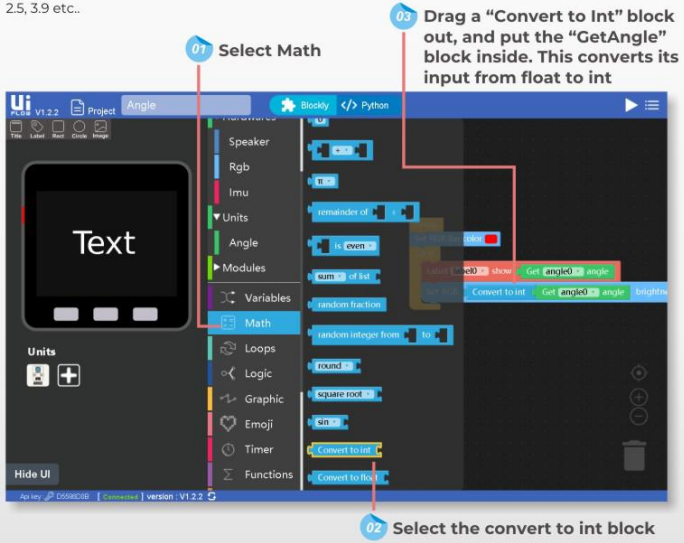
When we run the program and twist the knob, we will notice that an error occurs.



This error happens as a result of a different kind of input than expected, the RGB Bar expects an int (integer) value but the angle sensor inputs a float value.

3. Ints and floats

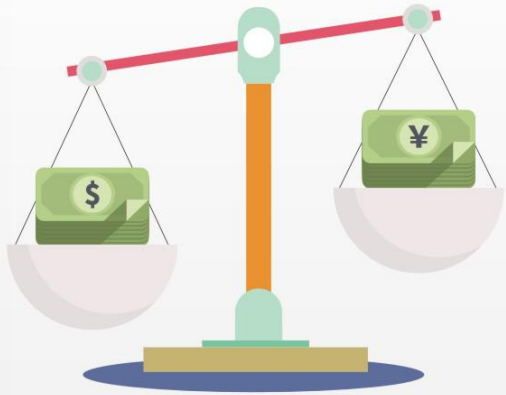
Int and float are two common data types. An Int can only represent whole numbers such as 0, 2, -4 and so on. While a float is used to represent numbers with a decimal point or floating point e.g. -1.1, 2.5, 3.9 etc..



This time if we run the program we will notice that the program works without error, but the RGB bar can only accept a value as high as 255 and we learnt that the angle unit inputs numbers as high as 1024. So even before we have twisted the knob to the extent of its range the highest value has already passed. Is there any way to solve this so we can have better control over the brightness?

4. What is mapping?

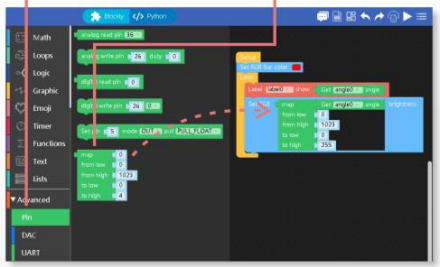
We can use the map blocks to solve this problem. the map blocks take one set of values and transpose it to another. For example if we change one currency into another such as dollars to pounds, the exchange rate dictates that the value of the money is not the same as the original value. Currency values are always changing but we can input a set value and change it to whatever value we want.



We need to add an RGB brightness block and insert the map block from the Easy IO section. The get angle block should be put in the map position. The first two values "from low" and "from high" are already set correctly as we mentioned the analog input will give us a value between 0-1023. Now the next two values we need to set to 0 and 255 in order to match with the max and min values that the RGB blocks accept. We don't need to use the convert to int block as the map will take care of everything for us.

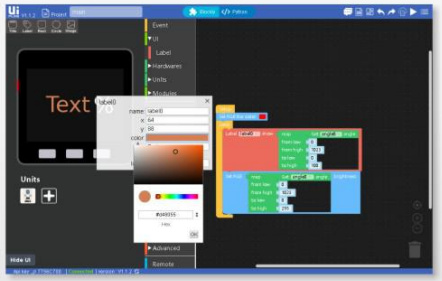
01 Click Advanced and select Easy IO

02 Drag out a mapping block into The RGB Brightness block



6. Setting up the display

In order to display the current value of brightness from 0 to 100% all we need to do is add another map block to the label show block, add the get angle block and then set the to low, to high values to 0 and 100.



Once you run the program you can now adjust the light correctly with the angle sensor.

11 Traffic Light

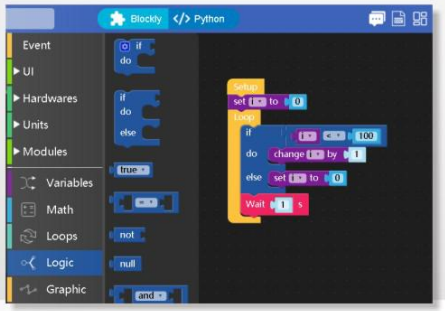
Introduction: In this chapter we will use the RGB Unit to create a traffic light program.
Lesson Outcome: Learn more about variables and iterators.

1. Understanding Iteration

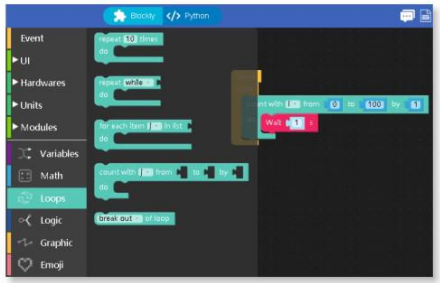
The count block is an iterator and it automatically assigns a variable if none is assigned and increments that variable until the specified range is met , the number the variable is incremented by can also be adjusted. To give an example, imagine there is an ATM with only £100 remaining and a queue of 5 people take £20 each. By the time the final person draws out the money there will be no money left. Likewise the program will stop running or the loop will be exited as the range has been met.

Quick Question: Why do we need to use an iterator and what benefit does it bring to the program?
We can demonstrate with a simple example.

Lets have a comparison of two ways to iterate numbers and see which one is more efficient. The goal is to count up from 0 - 100 in increments of 1.



Method 1
While "I" is less than 100 change "I" by 1 once "I" is more than 100 reset the value back to 0. Add a 0.1 - 1 second delay in order to see the change in the numbers. Place all of this code inside a loop.



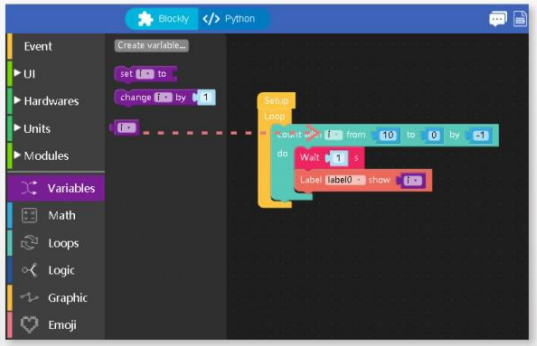
Method 2

Using the "count with i from" block we can set "i" to 0 , its max range to 100 and its increment value to 1. We will add a 1 second delay inside the count loop, so the program will run for 100 seconds. If we use a label show block and insert the "i" variable into it we will see the numbers increment by 1 up to 100 and then reset back to 0

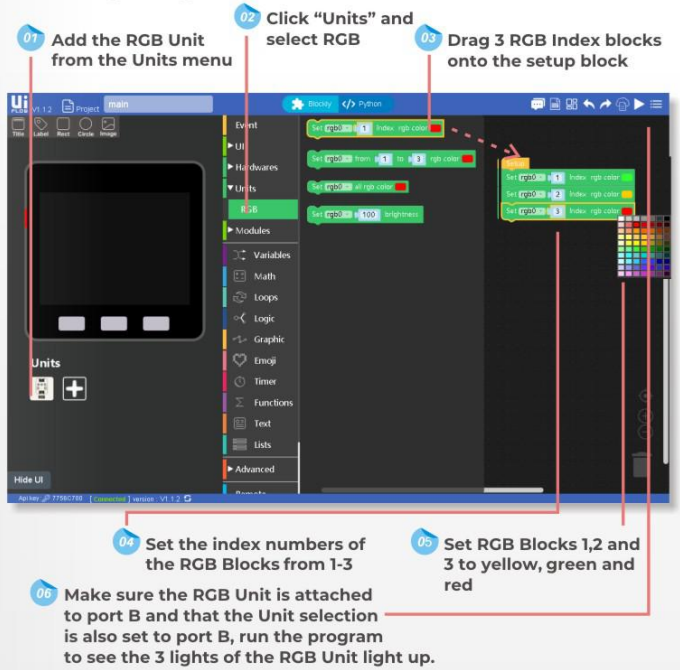
We can see that the second method using the count block has simplified our program.
It has many more uses than just simplifying code though as we will go on to see.

Supplementary information

In UI flow whenever we add a count block the variable "i" will be automatically generated, but we can also create other variables and assign them to the count block from the drop down list.



2. Traffic Light Program

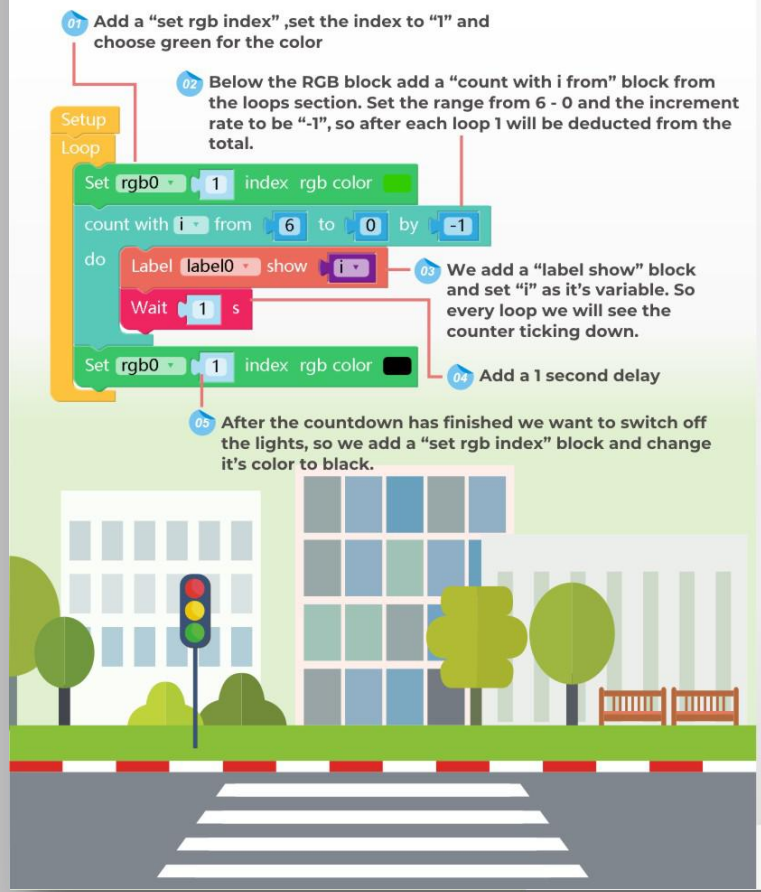


3. Traffic light display setup

Firstly we need to add 2 labels to the UI Manager. Click on the label to open the properties window and change the font size to a bigger size. Drag the other label to the right side of the first label and change its text to "s" for seconds.

4. The Traffic Light Program

We will separate the code into 3 parts. Control of the green light, yellow light and red light.



Yellow light

Usually the duration of the yellow light is much shorter and only stays lit for a short time. We can duplicate the first code and set this counter to only 3 seconds and the delay to 0.5, we also need to change the RGB Index to “2” then once the yellow light has turned on we turn it off again and add a delay of 0.5 seconds.

Red Light

Firstly we need to set the index to 3 and the RGB color to red. We can copy the count loop from the the green light code as it's exactly the same. Then the final thing we need to do is set the RGB led at index 3 to black.

The final RGB block we set to black so the light switches off before returning to the start of the loop.

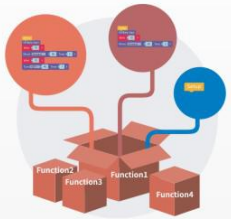
Once we run the code we can see the traffic light running as expected.

12 Light sensor and functions introduction

Introduction: In this chapter we will learn how to use the light sensor unit and start to use functions.
Lesson outcome: Understand the basics of functions and how we use them, get a deeper understanding of logic, and use the light sensor to control the rgb bar.

1. Understanding the basics of functions

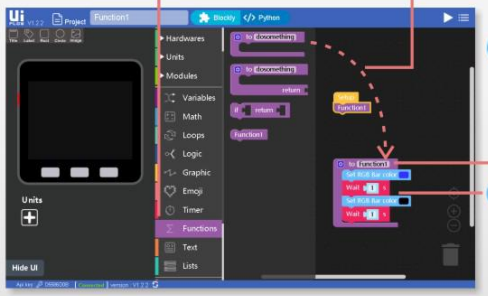
Functions are just like a package, we can name a function and put a bunch of code inside it, and when we want to use that code we can just call its name and it will run all the code inside it.



2. Creating a function

01 Click on the "Functions" tab in the blocks menu

02 Drag the "to do something" block into the coding area



03 By clicking in the text field of the function block we can change the functions name

04 We can add all of the code that we want to run inside of the function block this is called defining the function

3. Defining a function

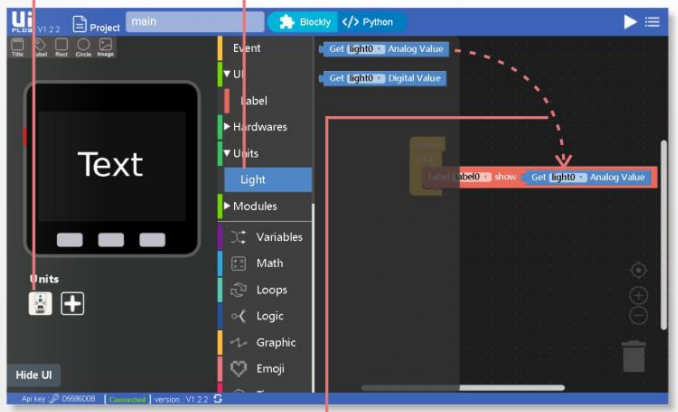
Once we have created a function, the function name will automatically appear in the functions section as a call block placing this block anywhere in your code will execute the code that you placed inside the c shaped defining block.

4. Testing the light sensor

The light sensor will input a range of numbers based on the light level.

01 Add the light unit from the units menu

02 Select "Light" from the units section of the block menu and drag the "get light analog value" block into a loop



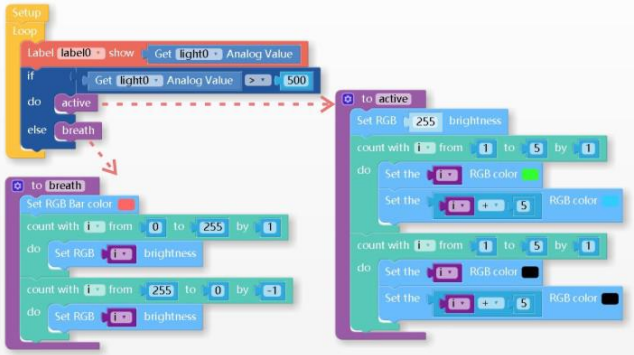
03 Add a label block and insert the "get light analog value" inside it

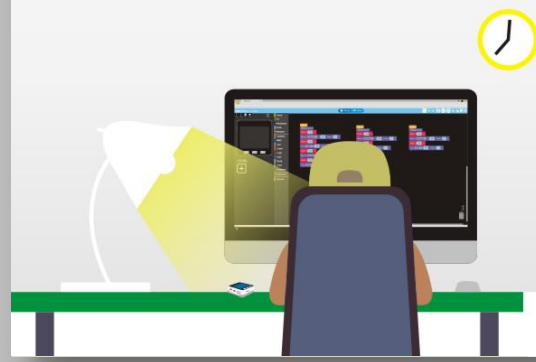
Once the light unit is connected to the M5GO and you have run the program.
You can try cupping your hand over the sensor and moving it closer and further away.
You'll see the values change on the screen between 0 - 1023.

Quick Question: How does the light sensor sense the light level and convert it into an analog value?
The light sensor is also known as a photoresistor or an LDR (Light Dependent Resistor) Unlike the potentiometer we don't need to manually adjust the LDR since depending on the level of light that falls on the LDR the voltage will increase or decrease in the circuit.

5. Light sensor function

In this section we will make different functions for different lighting effects such as brightness control. During the daylight hours or while the light value is more than 500 the function "active" will run. This function cycles the lights on and off. When daylight starts to fade and the light sensor value drops below 500 the function "breath" will run. This function gradually increases the led bar brightness and then decreases it once the led bar has reached maximum brightness.





13 M5GO Address book

Introduction: In this chapter we will learn about lists and use them to create a simple address book.
Lesson outcome: Understand the concept of a list, how to create one and how they are used.

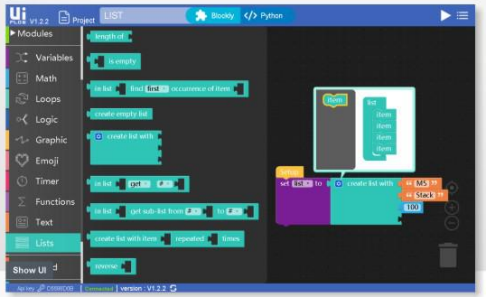
1. Understanding Lists

What is a list? We could say that a list essentially gathers together a bunch of variables in one easy to manage set. We previously described the variable as being like a bottle, we can then liken the list to a storage container for those bottles. Each bottle has its own numbered position in storage and this number can be used to get the data from that variable or change it.

When creating a list we must assign the list to a variable, then whenever we want to get data from the list we just use that variables name and state the position within the list from where we want to obtain the data.

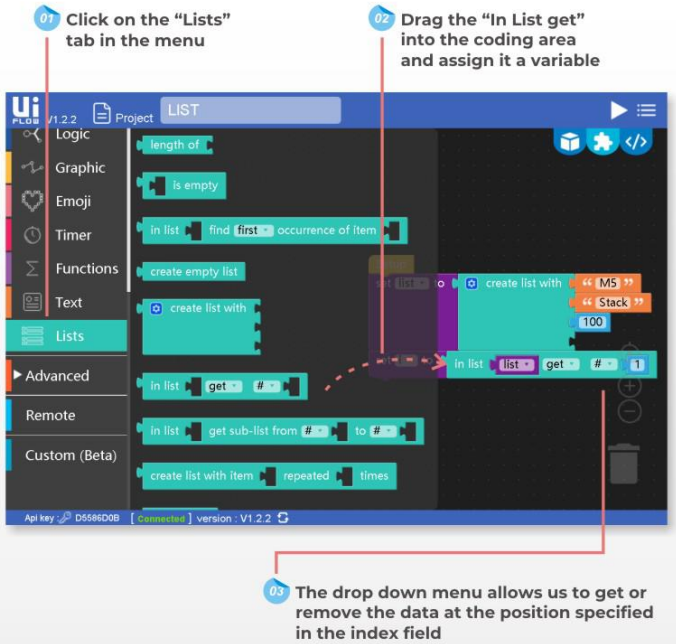
2. Creating a List

First we'll create a variable which we can call list, then we drag the "set list to" block to setup. In the blocks menu we can find a section called "lists", here we find the "create list with" block and drag it to the "set list to" block. Now to fill the list with data we can drag number blocks from the maths section, string(text) blocks from the text section or even other variables into the empty slots on the "create list with" block. Each block that we added to the list is now assigned a number starting from 1 in the first position, we can add more positions to the list block by clicking the little gear and dragging more item blocks in.



3. Using Lists

Once we have created a list and populated it with some data, we need to learn how to use this data and how to alter it according to our requirements.

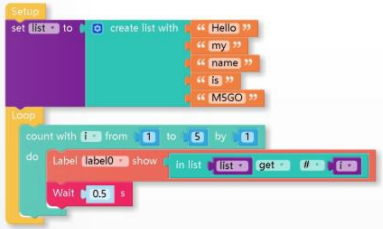


In UIFlow the index of lists starts from 1 however in Python it is 0. If we want to get the text M5 from the list we should use the index number "1".

Quick Question: Can we use a variable to cycle through the positions of the list?
Definitely, this is one of the most useful features of a list which has so many possibilities.

Quick test

We can use the count block to cycle through the positions in our list. Try to make this program.



M5GO Address Book

The program we are going to make will display a persons name and number on the screen from the list, and then we can use the A and C buttons to cycle through the list.

1. UI Setup

First we need to add 3 labels and set their text to display name and number headers and the third one we can leave blank as we will be using it to display the information we get from the list.

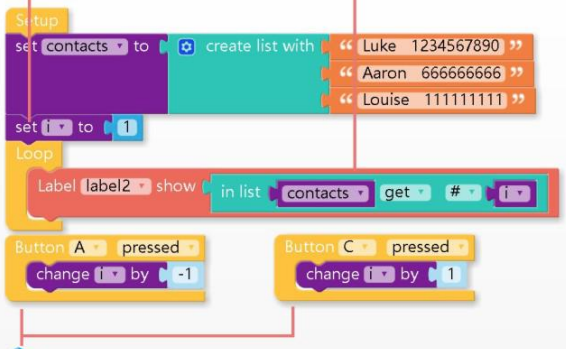
2. Adding the data to the list

Create a list and assign it to a variable called "contacts". In the blocks menu go to the "Text" tab and drag an empty string(text)block on to the list slots, duplicate the block and add the data for as many contact name and numbers as you like.

3. Using Lists

01 **Click on the "Lists" tab in the menu**

02 **Drag the "In List get" into the coding area and assign it a variable**



03 **Add 2 "Button pressed" blocks to the coding area. Set one to Button C and add a "change by" block to each, remembering to change the variable to "i" from the drop down list. Set the A button code to reduce "i" by -1 and the C button code to increase "i" by 1. Now we can change the position in the list by reducing and increasing the number of the "i"variable.**

4. Testing

Run the program and press the buttons to test. You may notice if you pressed the c button first that it works for a while. However a problem occurs when we press the button too many times, we get an error. Why does this happen?

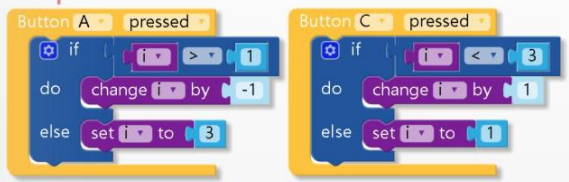
5. Index Error

When we press the a or c button continuously we reduce the "i" variables value to the point where its out of the range of the list. Remember we only created 3 positions in the list so if the "i" variable = 4 theres no fourth position in the list and thus we get an error.

6. Fixing the Error

Using the if blocks we can prevent this. We need to change the code within the A and C blocks.

After A is pressed check if "i" is more than 1 if so change by -1 if not set it to 3 which is the highest value in our list



After C is pressed check if "i" is less than 3 if so change by 1 if not set it to 1 which is the lowest value in our list

14 IR Remote

Introduction: In this chapter we will attempt to remote control one M5GO with another using the IR Unit.
Lesson Outcome: Learn how to prepare an image and display it on the screen of the M5GO. Understand What IR is and how we can use it.

1. What is infrared?

IR(Infrared) is a type of ultraviolet light that is invisible to the human eye. Infrared radiation was discovered by British scientist William Herschel in the 1800s.Herschel used a prism to split sunlight into its spectrum and in each color band he placed a thermometer to test the temperature of each color. Doing so he discovered that past the red light band was the area that was the hottest and concluded that in the light spectrum there must be an invisible light. This is what we call infrared. Apart from being an invisible light it can be used to transmit signals. IR has many applications in our lives such as turning on a TV or air conditioner.

In this lesson we will learn how to use the M5GO to make a remote just like the one you use with your TV. We need to use 2 M5GO devices and 2 IR Units, and upload two different programs to each.

2. IR Remote

01 Add an IR Unit from the Units Menu and 2 button pressed blocks

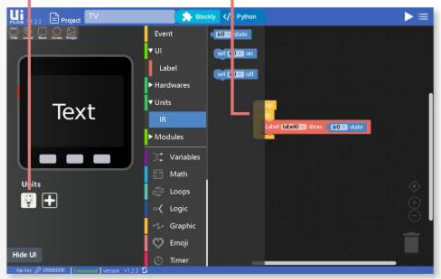
02 In the A pressed block we use the “set IR on” block and set a label to display IR on. This will cause the IR Unit to emit an IR signal

03 Set the second A button block to released, and use the “set IR OFF” block to stop transmitting

3. Display control program

01 Add the IR Unit

02 Add a loop, label show and the IR state blocks. Place the IR state block into the label show block.



Code Breakdown

Once we have uploaded both parts of the code to two separate M5GO devices and connected both IR Units, The IR Units should be pointed at each other and the A button pressed to send the signal.

(1)Once the A button is pressed on the remote device the infrared signal will be transmitted.
(2)When the display device receives the Infrared signal, The text on the screen will change from “0” to “1”.

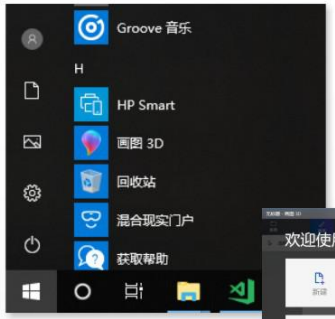
Now that we have established that we can transmit and receive signals, we should change the program to do something more exciting.

Image preparation

Now that we have established that we can transmit and receive signals, we should change the program to do something more exciting.

4. Picture requirements

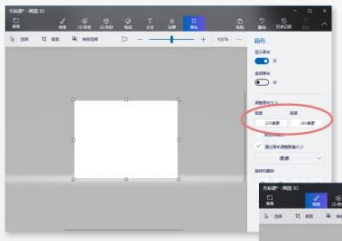
(1)The resolution or size of the picture should be no more than 320x240 pixels the size of the M5GO screen.
(2)The only file formats accepted are “bmp” and “jpg”.
(3)The file size must be within 25kb.



5. Prepare the image in an image editing software

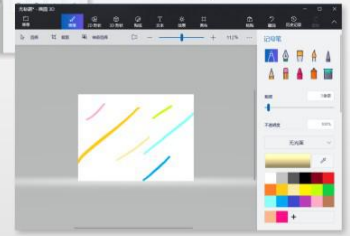
In most operating systems there will be some form of image editing software. We can use this software to prepare our images so they can be displayed on the M5GO. Open the image editing software onyour computer. In this tutorial we will be using Windows 10 default image editing software Paint 3D.

2. Open a blank canvas



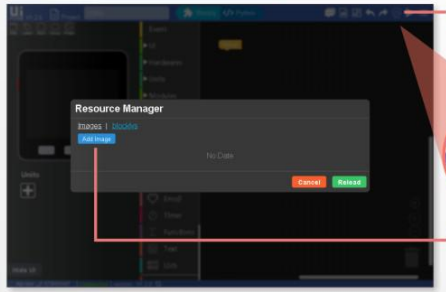
3. Set the canvas size to 320x240 this setting is usually in a menu called image on most software packages.

4. Choose the paintbrush or pencil tool and start to create your artwork.



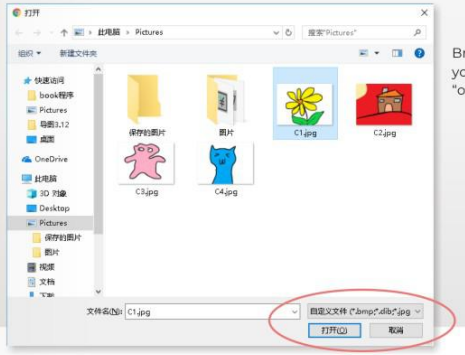
5. We can try to create multiple images with different content, perhaps scenes of a story. Once you've created your images make sure to save them, setting the compression level so that the image is smaller than 25kb and saving them with appropriate names such as image1 - 4.

Upload the images

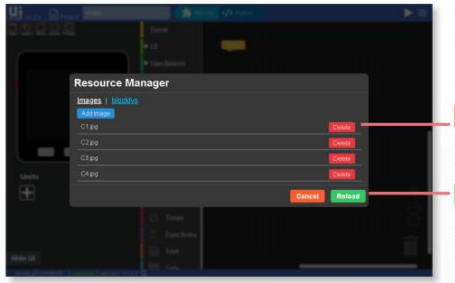


01 Make sure your M5GO is connected and click the Resource Manager Icon.

02 Click “Add Image”



Browse your computer to where you saved the images and click “open” to upload.



After the files have successfully been uploaded to your M5GO, they will appear in the list which means now we can program them.

Delete

Click "Delete" to remove any unwanted images.

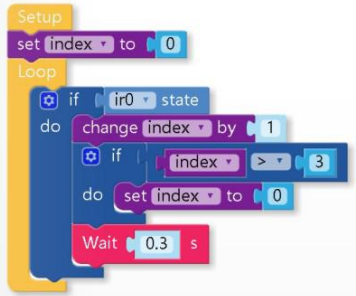
Reload

Clicking "Reload" will refresh the list to show all of the images stored in the image folder.

1. Programming pictures

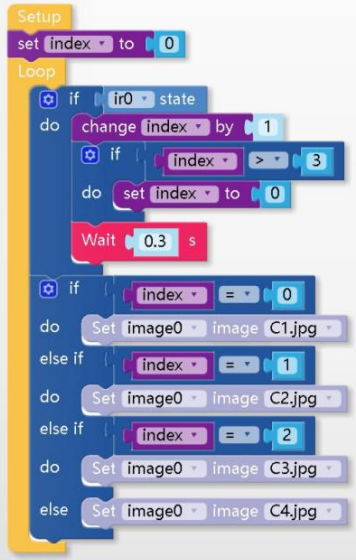
The method for displaying a picture in the UI Manager is similar to the way we've displayed shapes. Drag the image icon down to the screen and click it to open the properties, here we can select the image from the drop down "ImagePath" list, and set the position.

After adding an image to the UI Manager the related code blocks will appear in the UI section of the code blocks menu.



2. Variable and logic setup

Create a variable called index to represent the different images that we will display. Use an " if " condition and add an "IR state" block to it. Now we use a "change by" block so that when the IR signal is received the index variable will increase. Then we add an if condition to check whether the max number has been reached and reset the variable back to 0 if it has. This allows us to jump back to the first image after we have cycled through all of the images.



3. Image Cycle Code

Now that we have setup the variables and the logic all we need to do is assign each of the images to an index value. To do this we just need to add an extra if condition with enough else if conditions for every image. Then in each condition we check whether index is equal to a number between 0-3 or the number of pictures that you have created. Then if the index is equal to that number we use the set image block to display the image at each index.

To consider: In our daily life we use remote controls with many buttons but theres only one transmitter so how does it make different things happen on the press of each button? IR Remotes emit infrared in a series of timed pulses which correspond to 0's and 1's. Each button is programmed to send a code made up of 0's and 1's which will be linked to a specific function.

15 Game designer

Introduction: In this chapter we will use the buttons to control the movement of UI elements on the screen.
Lesson outcome: Use variables to control the x/y position of a UI element and get a deeper understanding of logic.

Ticking Time Bomb

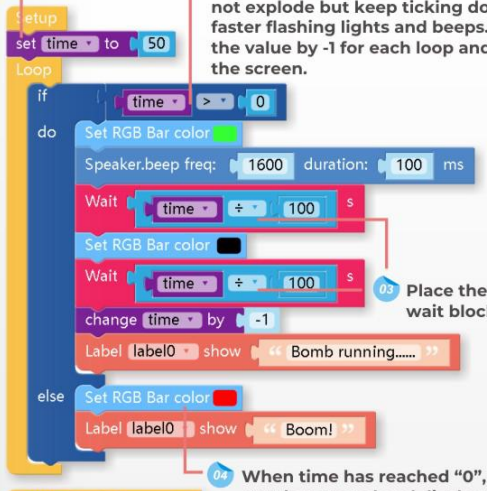
We will create a simple demo with a ticking time bomb. The beeps increase in frequency as the timer ticks down. Once the time is up the RGB bar will light up RED and the screen will display "Boom!".

In order to gradually decrease the time and make the lights flash and beeps get more frequent, we need to adjust the wait block to reduce the delay each time. How can we do that?
We can define a time limit for the bomb and then use logic blocks to check while the timer has not run out reduce the time limit for each loop.

Add a text label to the UI Manager to display the state of the bomb

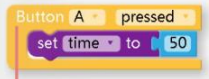
01 Create a variable called "time" and set its value to 50

02 Use the logic blocks to check if "time" is more than "0". While the value is higher than "0" the bomb will not explode but keep ticking down with progressively faster flashing lights and beeps. We will also decrease the value by -1 for each loop and display a message on the screen.



03 Place the time variable in the wait block and divide it by 100

04 When time has reached "0", we set the RGB bars to red and display "Boom!" on the screen

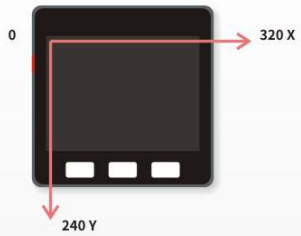


05 Add a button to reset the "time" variable back to 50, so we can run the program again

Dodgeball Game

1. Understanding the UI Manager Coordinates

The UI manager shows a representation of our screen which is 320 x 240 pixels wide. Objects are placed on the screen by giving them an x and y coordinate. Each pixel of the screen has its own coordinate. Some shapes are drawn from the pixel at there corner and some are drawn from the pixel at there centre.

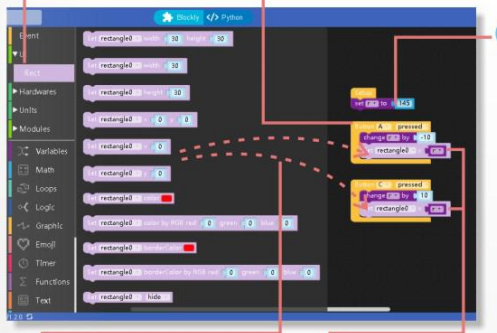


2. Create the player object

Add a rect element to the screen in the UI manager. Once added click on the square to bring up the properties Set the x to 145 and the y to 200.

03 Click UI and then click the “Rect” Tab

02 Drag two button blocks and set one to “C” add 2 “change by” blocks to the button loops and set the one in button A to decrease by -10 and the one in C to increase by 10



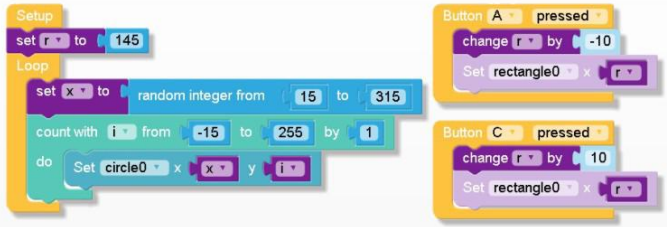
01 Create a variable named “r” and set its value to 145. Connect it up to the setup block

04 Take a “set rect x” block and add the “r” variable to control its position

05 Add a “change by” block inside each button block and set the one in A to change by -10 and the one in C to change by 10. This way when ever we press the A or C buttons the square will move back and forth across the X axis

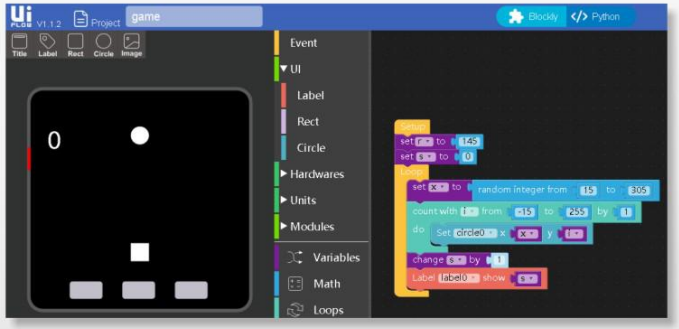
3. Randomly Falling Balls

Drag a circle onto the top of the UI Manager. In the following code we will make this ball appear in a random position at the top of the screen and fall down.
To begin with we need to create the variable "x" which will contain a randomly generated number between 15-315 the extremities of the screen. We then add a count block and set its range from -15 to 255 incremented by 1. Then we add a "circle x y" block from the UI section and set the x variable to x and the i variable to y.



4. Scoring code

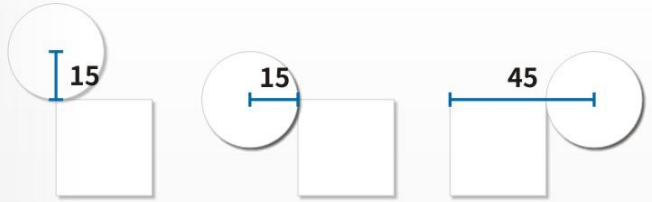
Create a variable called "s" and set its value to 0, this will hold the score for our game. Every time the ball passes our player, we change s by 1 and use the label block to display the score on the screen. Next we need to implement the code that checks whether the ball hit our player and if so, inform our player that they lost and it's game over.



5. Collision Detection

How do we detect the collision between the ball and player objects?
Using the logic blocks we can check whether the shapes have intersected.

When the shapes are on the same horizontal plane we can check if the radius of the circle which is 15 overlaps the width or height of the square which is 30. We can think about the possibilities as shown in the following diagrams.

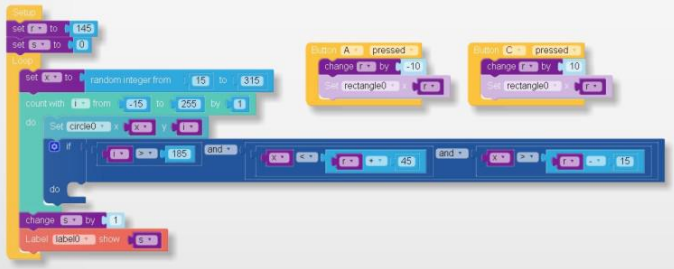


In the if block the first condition we check is if "i" is more than 185. This checks if the ball has fallen in range of the player object as there would be no chance of them colliding if it hadn't.

Next we check if x > r -15. This means that if the circle falls to the left hand side of the player object, with a distance of 15 it doesn't count as a collision.

When x < r - 45 means that if the circle falls to the right hand side of the player object, with a distance of 15 it doesn't count as a collision.

Once all conditions have been met, we establish that the ball has collided. Now all we need to do is program what happens after this occurs.



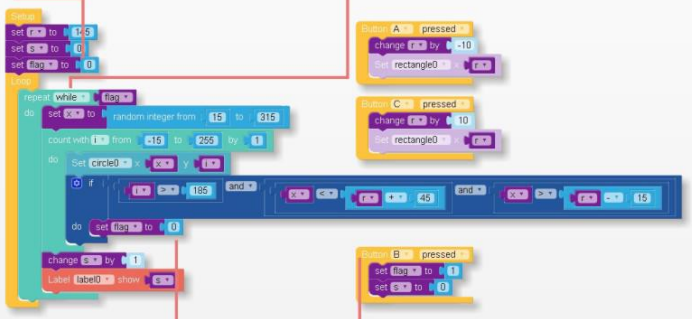
Why should we put the collision detection code inside the count block?
If we put it in another area can we still sense the collision? During the count loop the circle is falling from top to bottom from a random point. We need to check every time the ball falls whether it has collided. If we put the if condition in another place outside the count loop, it will miss the ball falling and produce an incorrect result.

6. Running the game

We will discover that after running the program the game will begin automatically, we did not program a way to manually start the game. During the game, after the ball has collided it means game over, but we have to design a way to make the game end and be able to restart. and stop the ball from falling. We should use a game state to check whether the game has ended, or is still in the state of play. Let's have a look how to do this below.

01 Create a variable called flag and set its value to 0 within setup

02 Use a repeat while block with the variable flag to switch between game over state and playing state



03 We need to cause the game state to change to 0 once the ball has collided in order to exit the game loop

04 In order to reset the game we can use the B button to set the variable flag back to 1 and at the same time reset the score to 0

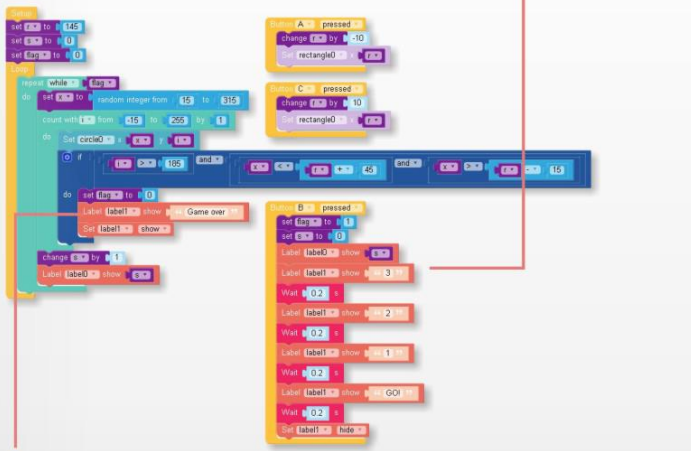
When the program starts the variable flag will be set to 0, so when we run the the program the game won't automatically start. The program will loop around indefinitely until the state is changed by pressing button b and setting flag to 1. Therefore every time we want to reset the game, we just set flag back to 1.

7. Program optimization

We've already tested that we can move the player, the ball falls from a random position each time, and on collision we enter game over. However there is no sign to tell the player it's game over or when the game is about to start. The game starts a little abruptly without giving time for the player to prepare. Lets see how we can modify the code to do so.

Lets think about it. Usually how do games notify the player that they are about to start?
Perhaps with a countdown. We can use a label to prompt the player to "press button b"to start the game and then display a countdown and a game over message if the player loses.

01 In the Button B pressed block, we can add a series of label blocks and wait blocks to create the countdown.. Once the countdown has finished we can use a label hide block to make it disappear from the screen, while in play mode

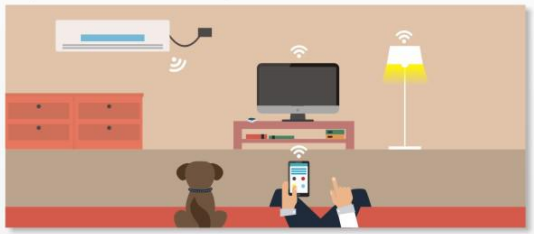


02 After the collision logic and the section where we change the game state, we can add a label to display "Game Over" and use a label show block to make the label we hid before reappear

16 Remote Control

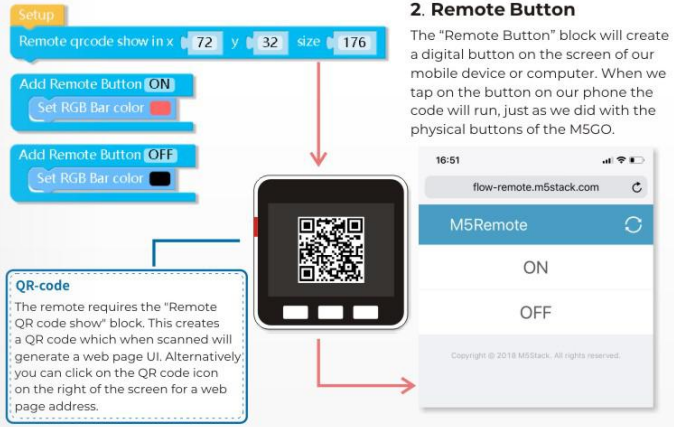
Introduction: In this chapter we use the remote blocks to control the light, and display some information remotely.
Lesson Outcome: Learn how to use the remote blocks to create on off switches and sliders on the remote interface.
Reminder: The remote blocks require the M5GO to be connected to the internet. In order to connect to the internet check the guide on page 105.

What is remote control, why should we learn about it, and what can it be used for?
We previously used IR as a remote control method. Remote control means to control something from a distance usually without wires, using IR, Wifi or some other means. We can use remote control to conveniently switch on or off a light, TV or some other electrical appliance in our home, without even needing to leave the sofa. We can do more than just turn them on and off as well, we can even monitor their status, for example monitoring the current temperature of our AC. All these things can make our life more convenient and efficient.



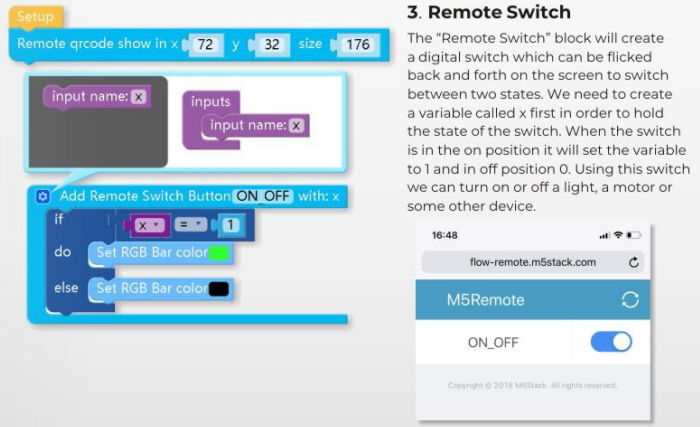
1. Remote

In the remote section at the bottom of the blocks menu we can find a whole bunch of options to create our mobile remote interface. On the next page we will see how we can use these to create a remote control program.



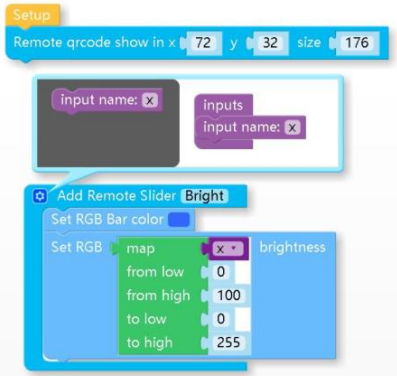
2. Remote Button

The "Remote Button" block will create a digital button on the screen of our mobile device or computer. When we tap on the button on our phone the code will run, just as we did with the physical buttons of the M5GO.



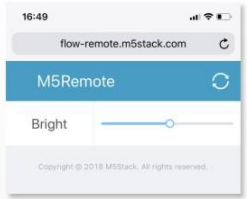
3. Remote Switch

The "Remote Switch" block will create a digital switch which can be flicked back and forth on the screen to switch between two states. We need to create a variable called x first in order to hold the state of the switch. When the switch is in the on position it will set the variable to 1 and in off position 0. Using this switch we can turn on or off a light, a motor or some other device.



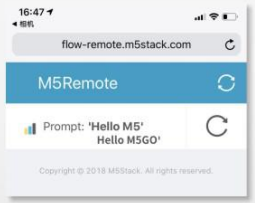
4. Remote Slider

The remote slider is,a draggable button on a line, which can be used to control the value of a variable. Before we add a remote slider we should configure a variable to signify its value. When the slider is dragged it can change the range of the variable between 0 and 100.



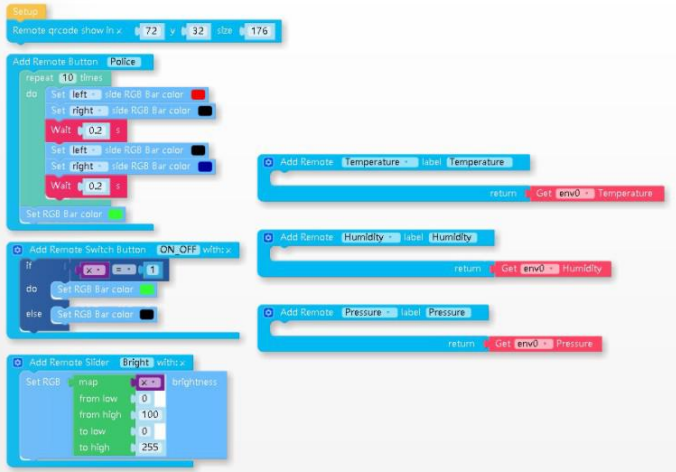
5. Remote Label

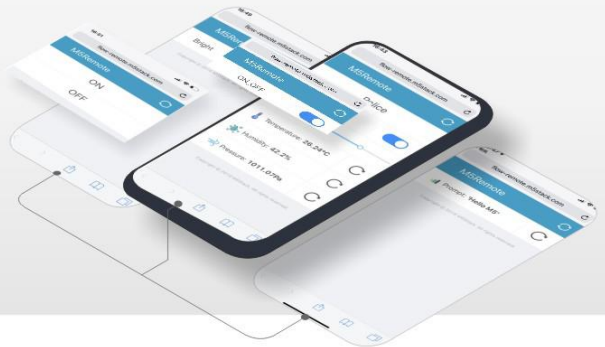
The remote label is used to display information such as sensor data, some static text or the value of a variable. We could use it to display the temperature from the ENV Sensor.



Overview

Using the remote blocks allows us to turn the RGB bar on and off, adjust the brightness level of the RGB bar and control some effects and also display sensor data from Units like the ENV to your mobile phone screen.





「APPENDIX」

IDEAS FAST TRACKED

Firmware Upgrade

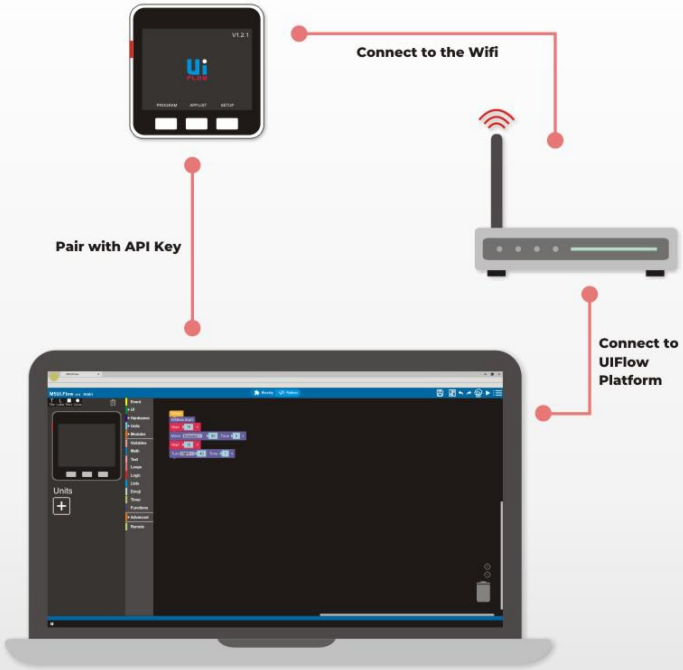
WIFI Configuration

Learn more

Appendix

Setup Wifi

Other than supporting programming over USB, the M5GO can also be programmed wirelessly over the internet. The web version has some extra feature related to wireless communications and IoT. Although the web IDE and offline IDE look almost the same, the main difference is we must connect our device to wifi before we can use it. We also need to use the devices API Key to pair it with the website.



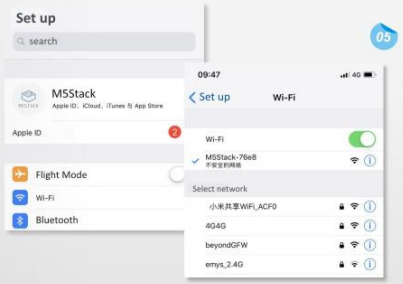
Wifi Setup

01 Press the red button on the left side of the device to power on

02 When the Menu screen appears press the far right button to enter the wifi setup page

03 Use the left and right buttons to scroll through the menu and select “Change Wifi Connect”

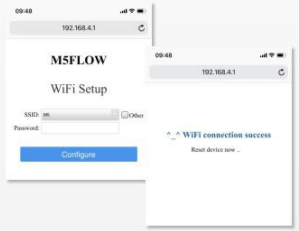
04 After selecting “change wifi connect” The screen will show the hotspot name of your device



05 Look for the hotspot name of your M5GO in the Wifi Settings list of your computer or Mobile Device

06 Once a device has successfully connected to the hotspot of the M5GO the QR code and IP Address will appear

07 Open your web browser and type 192.168.4.1 or turn on your mobile phones camera function and scan the QR Code to access the WIFI setup page



08 Select the Wifi network that you wish to connect to and enter your password. Click the connect button, and once your device is connected it will automatically reset

Linking your API Key

01 When booting the device you will see the main menu. Press the button below Program to connect to the UIFlow platform

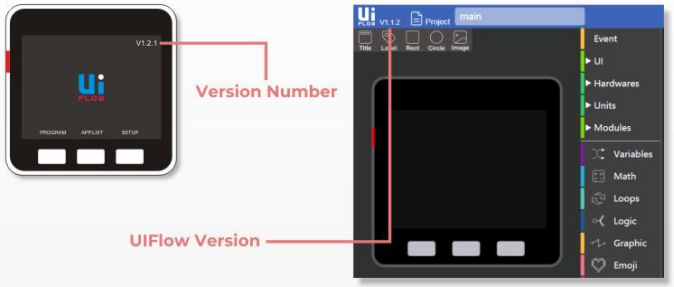
02 After entering programming mode you should see the small circle in the upper right corner change from red to green, to signify it has connected successfully

03 Enter the web address in the blue banner or scan the QR Code with your mobile device to open the UIFlow web page. Once the page has loaded it should automatically bring up the API Key setting. If not, click the small gear in the upper right hand corner

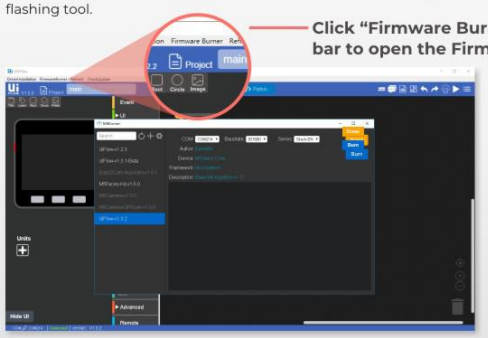
04 Input the API Key displayed on the screen of your M5GO into the API key field on the setup page. Now we can start to code

Updating Firmware

In order to keep up with all the latest features of UIFlow we will need to update the firmware on our device from time to time. The M5GO firmware version should be the same version as the UIFlow software.When booting up your M5GO device you will see the firmware version in the top right hand corner. In UIFlow you can check the firmware version in the upper left hand corner. If you notice that the UIFlow version is greater than that on your device it means the UIFlow software has been updated and you need to update the firmware on your device accordingly. The UIFlow software comes paired with a firmware flashing tool, which allows us to update the M5GO firmware.



In order to flash the firmware you must connect your M5GO device to the computer via USB. After connecting you can click on "Firmware Burner" in the upper right hand corner, to open the firmware flashing tool.



1. COM

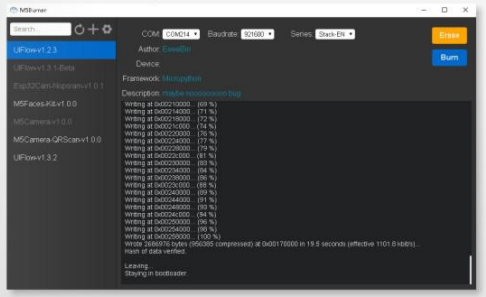
When your M5GO device is connected to the computer it will be assigned a COM port number. We must select this from the drop down list in order to flash the firmware to the correct device. On widows it will be named COM with a number after it e.g. COM32. On Mac and other systems it may look like this: /dev/tty.SLAB_USBtoUART

2. Baud

The baud rate is the speed at which the firmware is flashed to the device. Generally 115200 or 921600 are the best option.

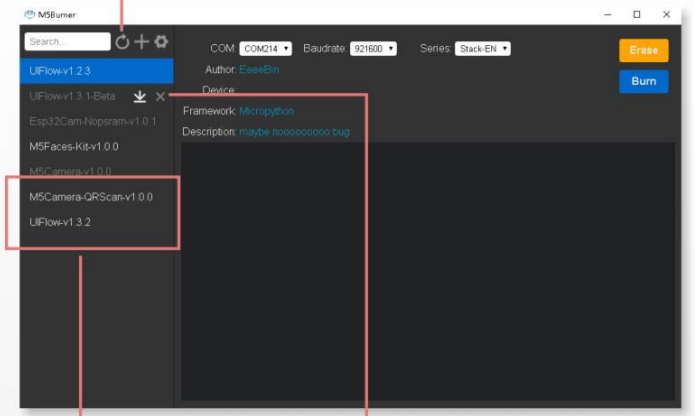
3. Firmware

Choose your desired firmware, it may be the latest or an older version depending on your needs. Different M5Stack devices, require different firmware. If your programming with UIFlow you need to flash a UIFlow compatible firmware.



4. Firmware Version

Select from more firmware versions.



01 Click the refresh button to check for other new firmware versions

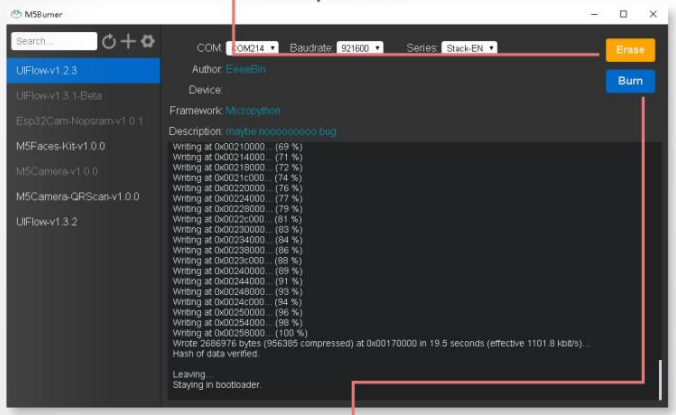
02 The firmwares highlighted in white are those that have already been downloaded and can be flashed

03 Firmwares in gray can be downloaded by clicking the white download icon next to them.Select from more firmware versions

5. Flashing the firmware

Select the firmware you want to flash.

01 Before we flash a new firmware to our device it is best to “Erase” the storage of the M5GO device. When you see the message “finished” displayed the storage has been wiped. We can choose not to fully erase if we have already setup a wifi network on the device as these details can be preserved



02 Click “Burn” to begin flashing, again the message “Finished” will be displayed once the flashing process is complete

In conclusion

Visit https://m5stack.com to find more information and resources about M5Stack.

We hope that in reading this book you were able to pick up the basic concepts of programming the M5GO in UIflow. We also hope that you had fun while learning to code and used your new programming skills to create your own programs. This is just the start of your programming journey and you still have much to learn.

There are more units and accessories that can be used with the M5GO. Why don't you give them a try, they can be purchased from the links below.

public account Aliexpress Taobao