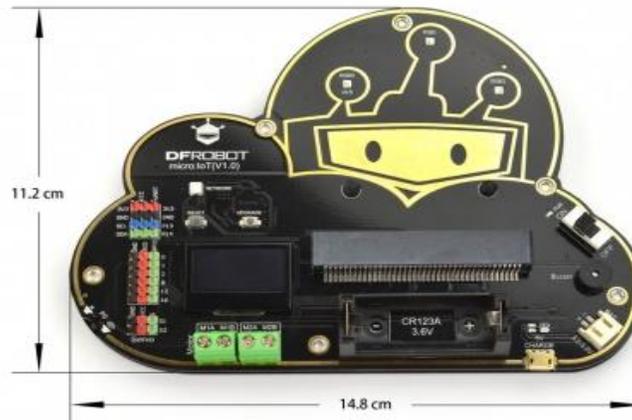


# micro: IoT - micro:bit IoT Expansion Board

SKU:MBT0012



## Introduction

With the development of IoT, there are so many mature IoT platforms emerging in both domestic and overseas, but most of them are mainly designed for the professional, which could be very hard for the non-experts to get started. Therefore, we specifically developed this Micro: IoT, a micro:bit-based IoT expansion board. Use the board together with DFRobot Easy IoT platform, lower the barrier of using IoT.

This micro: IoT board is pretty cute and delicate, on which we integrated Wi-Fi, OLED, 2-way motor drive, 6-way IO port, 2-way IIC, 1-way serial port, 1-way buzzer, 3-way RGB, 2-way servo, Li-ion battery holder, charging circuit, emergency power interface etc. So many resources for you to programme! Even without complicated background knowledge, you can build your IoT applications in few steps!

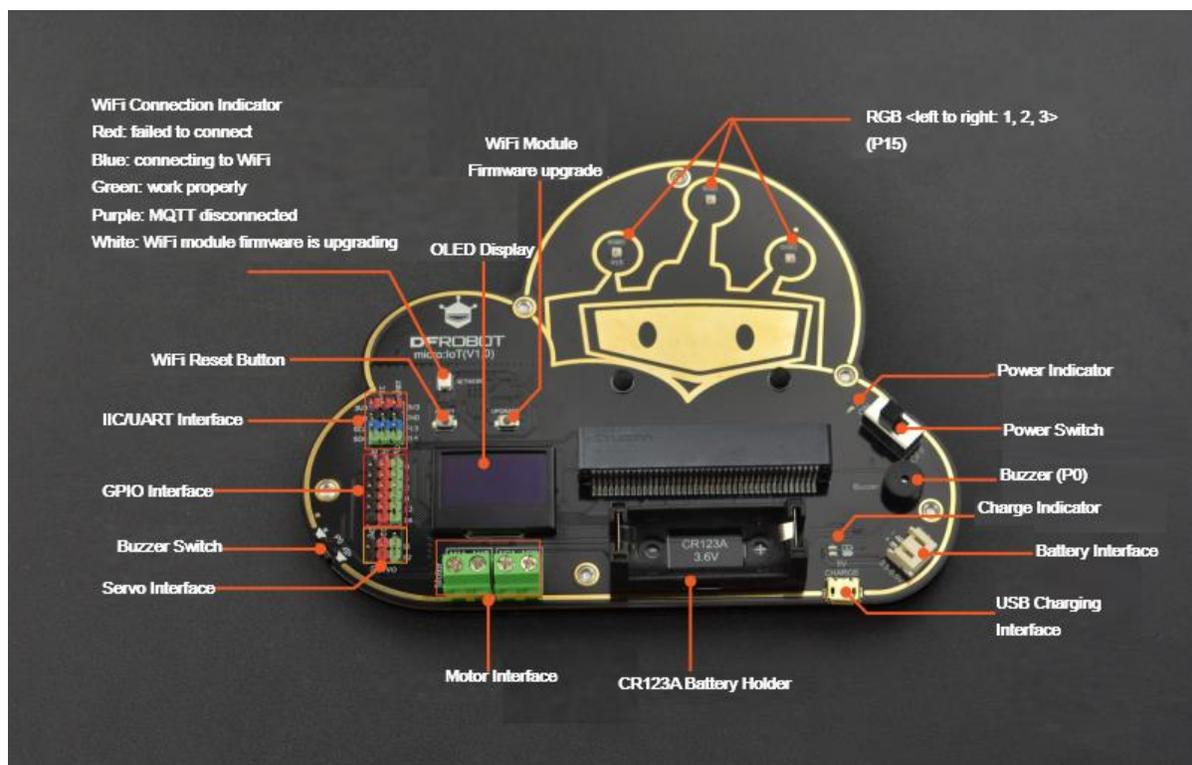
We designed the board as cloud shape to make it more suitable for the theme of IoT. As for power supply, we selected CR123A 3.6V lithium rechargeable battery of 1000mA. This kind of battery features small size, high capacity and high safety. What's more, we add short circuit and reverse connection protection for the product to further improve the safety in using Li-ion Battery.

Supports makecode and Mind+ graphical programming platforms.

## Specification

- Battery Type: CR123A 3.6V Rechargeable Li-ion Battery (Note: do not use 3V CR123A battery and non-rechargeable. Recharging the non-rechargeable battery is dangerous.)
- Onboard charging circuit
- Interface: 2-way IIC, 1-way serial
- Micro:bit Interface: P0 P1 P2 P8 P12 P16
- Other Interface: two DC motor ports, two servo ports
- Standard Gravity Pins
- On-board 3-way RGB LED
- On-board buzzer and Switch
- On-board OLED display
- Dimension: 148×112mm/5.83×4.41"
- Weight: 104g
- Programming Platforms: makecode, Mind+

## Overview



Color	Indication
Red	failed to connect
Blue	connecting to WiFi
Green	Work properly
Purple	MQTT disconnected
White	WiFi module firmware upgrading

**Note:** when the indicator turns purple, it means that MQTT is disconnected, and please check if the id account is correct.

## Makecode Online Graphical Programming

---

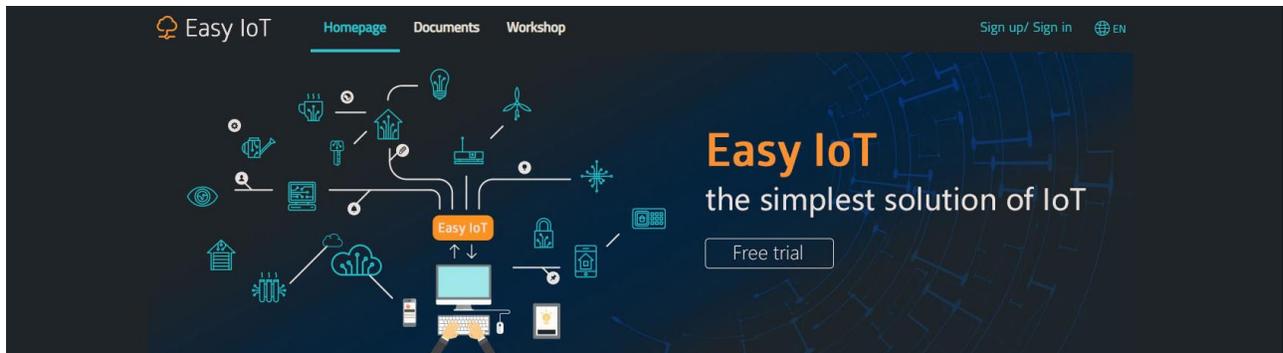
Micro:IoT library address: <https://github.com/DFRobot/pxt-microIoT>. [How to install an library?](#)

## Easy IoT Platform

- Easy IoT from DFRobot is dedicated to simple solutions for Internet of Things.
- Features Included:
  - Get started in 10 minutes, plug and operate
  - OBLOQ supported, serial port transmission
  - Real time monitoring, data analysis

## Sign in and Setting Easy IoT

- Open the easy IoT website : <http://iot.dfrobot.com.cn/>



- With a micro:bit board, a micro:bit IoT expansion board and Easy IoT, a whole IoT system will be ready.
- **Sign up and Sign In**
  - Click the Sign up/Sign in at the upper right corner of the webpage to register a account.
  - Fill in the black with your personal information, you can register via phone or email.
  - Sign in the Easy IoT, then it will enter into Workshop interface. You need to add a new device to match your micro:bit.



## Register an account

Have an account? Immediately [Login](#)

 Easy IoTLogout 

[Homepage](#) [Documents](#) [Workshop](#)

lot\_id(user)

lot\_pwd(password)

Allocated: 0/10000

+

 Easy IoTFree access to 10,000 messages storage.Free Signup

- **Easy IoT Setting**

- When added a new device, a device topic Topic: eIJA7JHZg will be generated randomly, and you can click the device name to change it.
- Click "send message", then the webpage of message receiving and transmitting will appear. You can find the details of the received data.
- Get to know other information in this page- ID account interface

0/1000

## New Device

Topic: sNxeKupZg

Send msg View details

Topic: one topic can correspond to one or more devices for storing interactive data.

Move your mouse onto the "New Device", a pencil icon will appear. Click it to rename.

Click to check the details of this topic.

Easy IoT [Homepage](#) [Documents](#) [Workshop](#) [France](#) [Logout](#) [EN](#)

[←Go back to my workshop](#)

**micro:bit device** Device name

**Send a new message** Send a new message

Adding a -> prefix to the message means that the message is a pure command message and will not be stored in the database. For instance "->off"

Please enter the message. Input limit is 128 bits. **Send**

**View message** Check messages

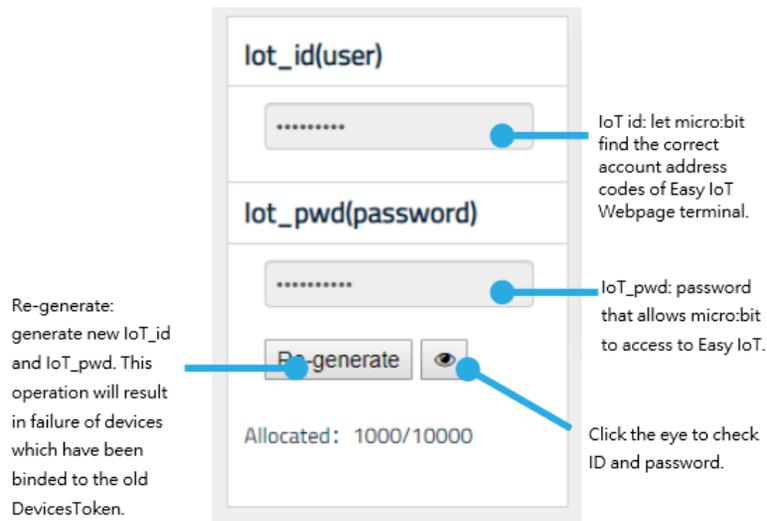
Date range  -

**Check**

**Latest news** List of the latest messages

Time	Message

[Previous](#) [Next](#)



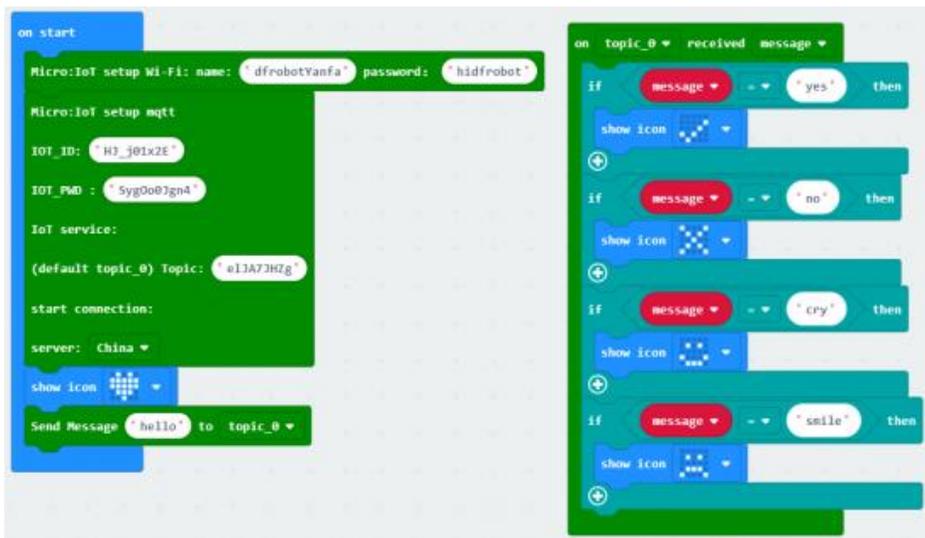
## Easy IoT Basic Tutorial

### Smart Emojis

Makecode Program: [smart emojis](#)

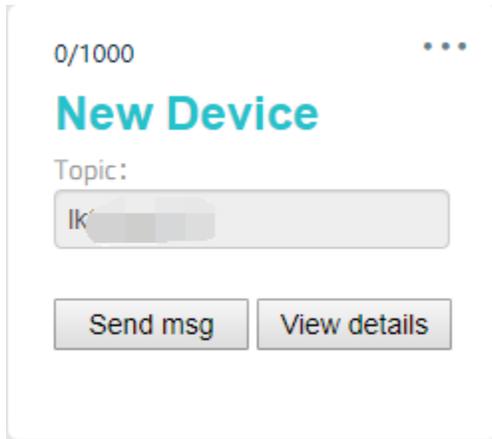
Function: display your current feeling. For example, if you are not in the mood, you can send a message to micro:bit through Easy IoT to let micro:bit board display the set pattern.

### Programs for reference



## Program Effects:

- Enter Easy IoT webpage to send message to micro:bit and find the related Topic, click "send message".



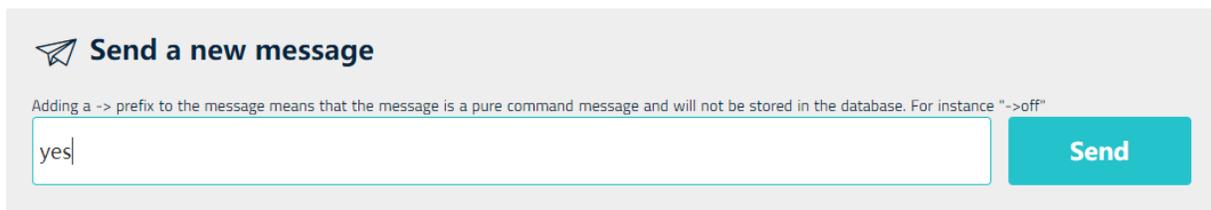
0/1000

### New Device

Topic:

Send msg View details

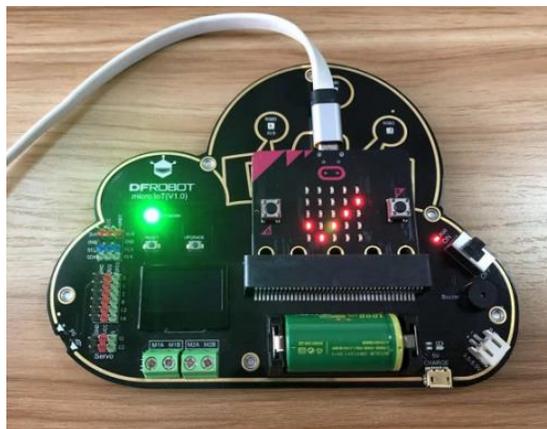
- Input your command into the blank, and send it out. Send the message "yes", micro:bit displays "√"; "no", "×"; "cry", "crying face pattern"; "smile", "smiley face".



 **Send a new message**

Adding a --> prefix to the message means that the message is a pure command message and will not be stored in the database. For instance "-->off"

Send



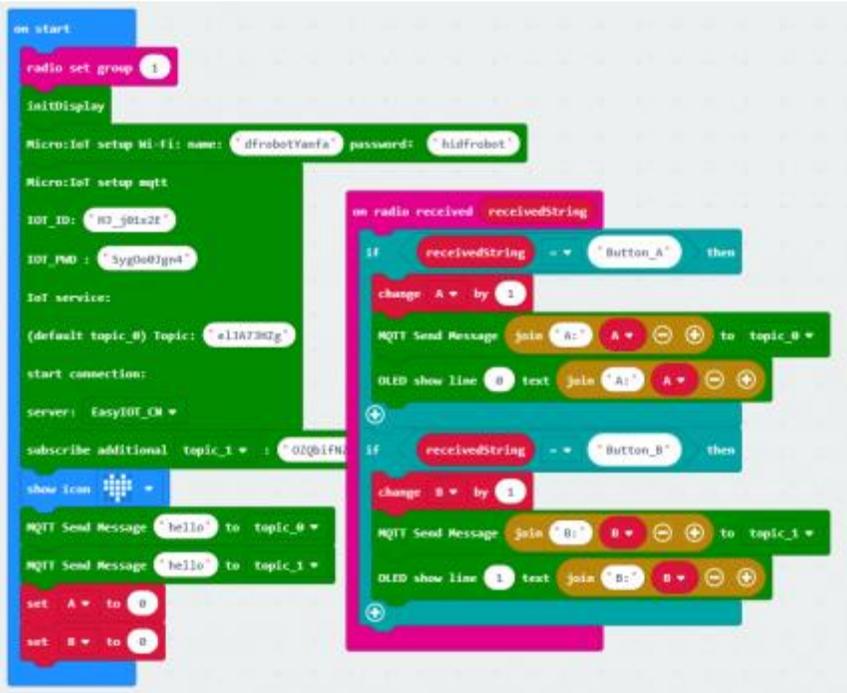
## Voting Machine

Makecode programs: [voting machine-hosting](#)

Makecode programs: [voting machine-voting](#)

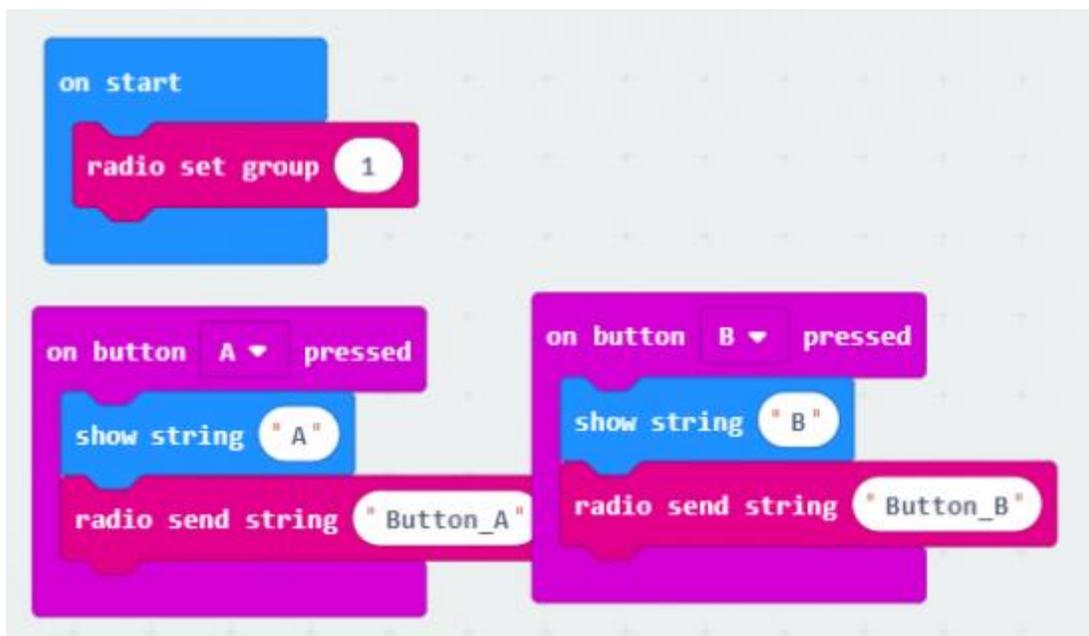
Function description: voting machine utilizes the wireless communication function. The voting machine has hosting post and voting port. The voting port can have abundant devices. When the voter presses the related button, the number of the votes will be displayed on the screen. Meanwhile, the data will be sent to IoT platform.

### Programs for reference



```
on start
  radio set group 1
  initDisplay
  Micro:IoT setup Wi-Fi: name: dfrobotYanfa password: hiffrbot
  Micro:IoT setup mqtt
  IOT_ID: 82_s8ta2E
  IOT_PID: 3yg0o0jgn4
  IoT service:
  (default topic_0) Topic: 413A7382g
  start connection:
  server: EasyIoT_CN
  subscribe additional topic_1 : 020b1FN
  show icon
  MQTT Send Message "hello" to topic_0
  MQTT Send Message "hello" to topic_1
  set A to 0
  set B to 0

on radio received receivedString
  if receivedString == "Button_A" then
    change A by 1
    MQTT Send Message join "A:" A to topic_0
    OLED show line 0 text join "A:" A
  if receivedString == "Button_B" then
    change B by 1
    MQTT Send Message join "B:" B to topic_1
    OLED show line 1 text join "B:" B
```



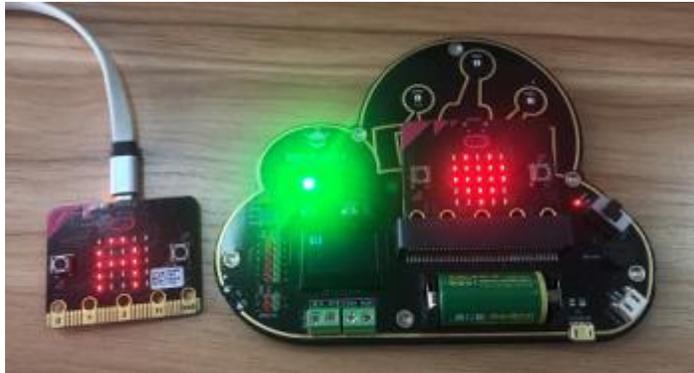
```
on start
  radio set group 1

on button A pressed
  show string "A"
  radio send string "Button_A"

on button B pressed
  show string "B"
  radio send string "Button_B"
```

## Programs Effects :

Press down button A, send a message to topic0 and change A by 1; press down B, send a message to topic1 and change B by 1.



```
on start
  radio set group 1
  initDisplay
  Micro:IoT setup Wi-Fi: name: "dfrobotYanfa" password: "hidfrobot"
  Micro:IoT setup mqtt
  IOT_ID: "H2_501a2E"
  IOT_PWD: "5y0o0jgn4"
  Iot service:
  (default topic_0) Topic: "e13A7302g"
  start connection:
  server: EasyIoT_CN
  subscribe additional topic_1 = : "02081FN"
  show icon
  MQTT Send Message "hello" to topic_0
  MQTT Send Message "hello" to topic_1
  set A to 0
  set B to 0

on radio received receivedString
  if receivedString == "Button_A" then
    change A by 1
    MQTT Send Message join "A:" A to topic_0
    OLED show line 0 text join "A:" A
  if receivedString == "Button_B" then
    change B by 1
    MQTT Send Message join "B:" B to topic_1
    OLED show line 1 text join "B:" B
```

 Latest news

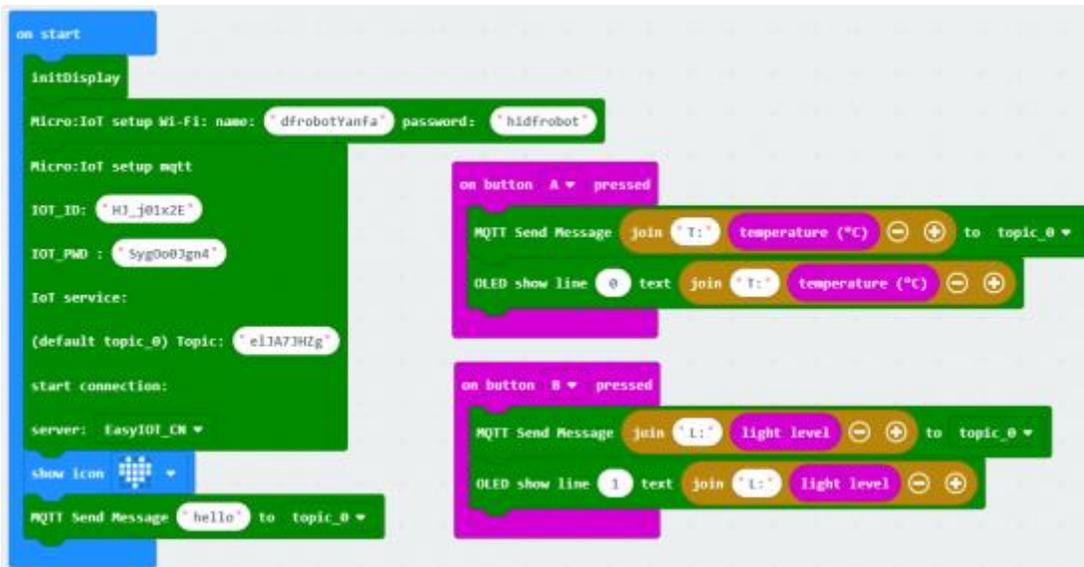
Time	Message
2019/9/4 13:24:16	B:2
2019/9/4 13:24:12	B:1

## Light and Temperature Monitor

Makecode programs: [light and temperature monitor](#)

Function description: use micro:bit to detect the light and temperature, and send the data to Easy IoT and OLED by pressing button A and B.

### Programs for reference



Programs effects:

Press down button A, send the current temperature value to the workshop; the first line of the OLED screen displays the temperature value. Press down button B, send the current light value to the workshop; the second line of OLED screen displays the light value.

T for temperature; L for Light



### Latest news

Time	Message
2019/9/16 17:29:57	L:255
2019/9/16 17:29:56	T:27
2019/9/16 17:29:30	hello

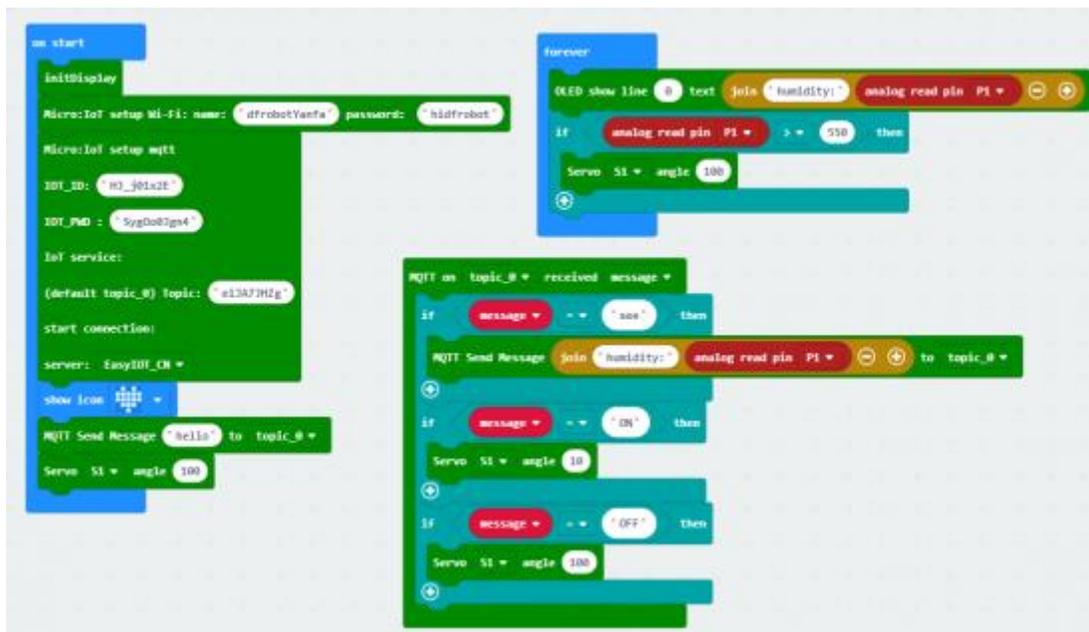
## Remote Watering System

makecode programs: [watering system](#)

Function description: connect a soil moisture sensor to the module's P1, connect a servo to pin S1. When the internet connected, send the command "see" to Easy IoT to check the soil humidity. We can decide if the plants need to be watered according to the humidity data. In order to avoid we forget to turn off the water system, we should first set that, when the humidity is over than 550, the water system will be turned off automatically.

Components: servo, soil moisture sensor(P1)

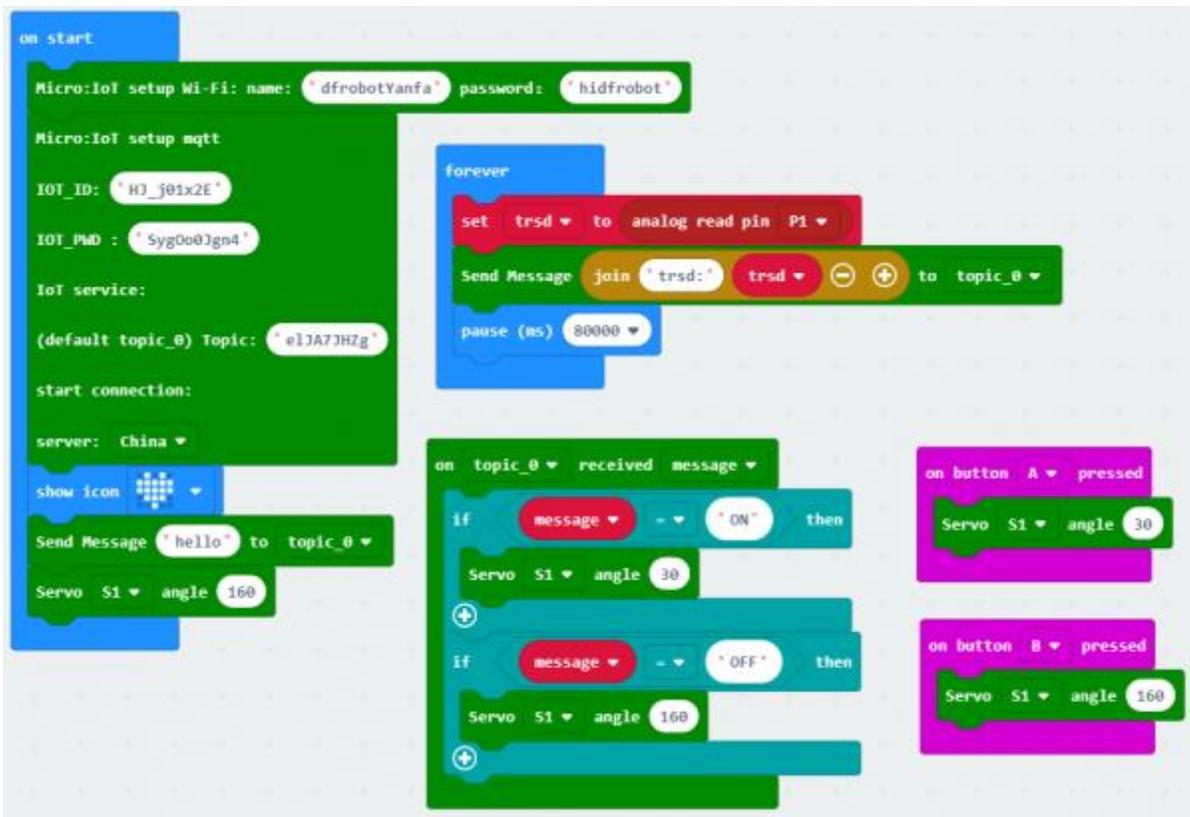
### Programs for reference



Program Effects:

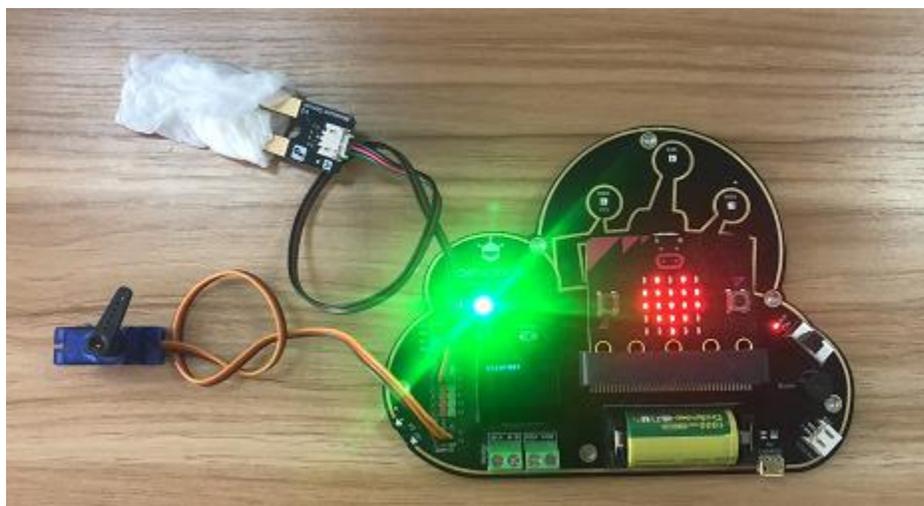
- Watering plants by remote control: when connected with WiFi, the servo rotates 100 degrees, and the watering system is closed. Send commands to micro:bit via Easy IoT, if the micro:IoT received the command "ON", the servo rotates 10 degrees to enable the watering system. When the micro:IoT received the command "OFF", the servo rotates to 100 degrees, or the humidity is larger than 550, the system will be disabled.

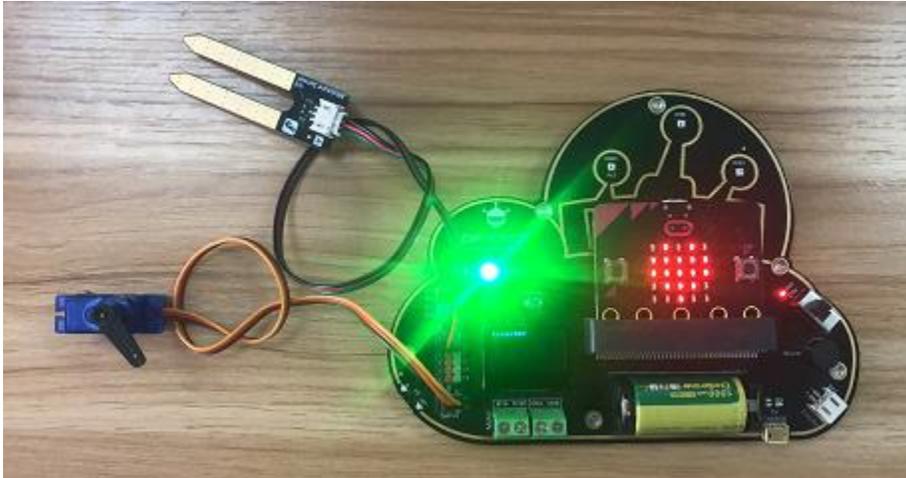
Note: when the servos turns to 100 degrees, the watering system is closed, servo to 10 degrees, the system opened.



### Latest news

Time	Message
2019/9/17 17:11:7	humidity:2
2019/9/17 17:11:6	see





## Automatic Clothes Hanger

Makecode Programs: [automatic clothes hanger](#)

Function Description: send humidity value to Easy IoT every 8 minutes, then user can determine if it is suitable to dry our clothes. When OBLOQ receives "OFF", the servo rotates to a certain degree to retract the clothes hanger; When OBLOQ receives "ON", the servo rotates to another degree to stretch the hanger.

Components: Servo(S1), soil moisture sensor(P1)

## Programs for Reference

```
on start
  initDisplay
  Micro:IoT setup MI-FI: name: "dfrobotYanfa" password: "hidrobot"
  Micro:IoT setup mqtt
  IOT_ID: "H0_j01x2E"
  IOT_PWD: "Syg0o0Jgn4"
  IoT service:
  (default topic_0) Topic: "e1JA7JHZg"
  start connection:
  server: EasyIoT_CN
  show icon
  Servo S1 angle 160

forever
  set trsd to analog read pin P1
  OLED show line 0 text join "trsd:" trsd
  MQTT Send Message join "trsd:" trsd to topic_0
  pause (ms) 80000

MQTT on topic_0 received message
  if message == "ON" then
    Servo S1 angle 30
  if message == "OFF" then
    Servo S1 angle 160

on button A pressed
  Servo S1 angle 30

on button B pressed
  Servo S1 angle 160
```

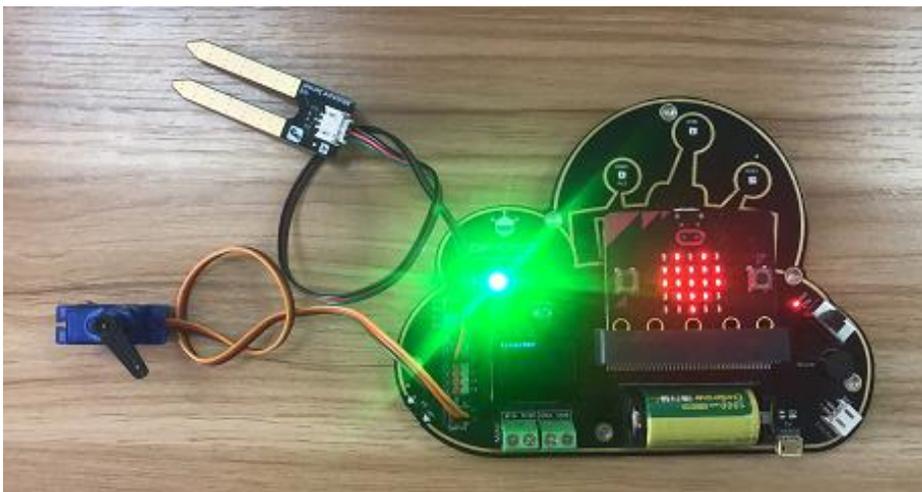
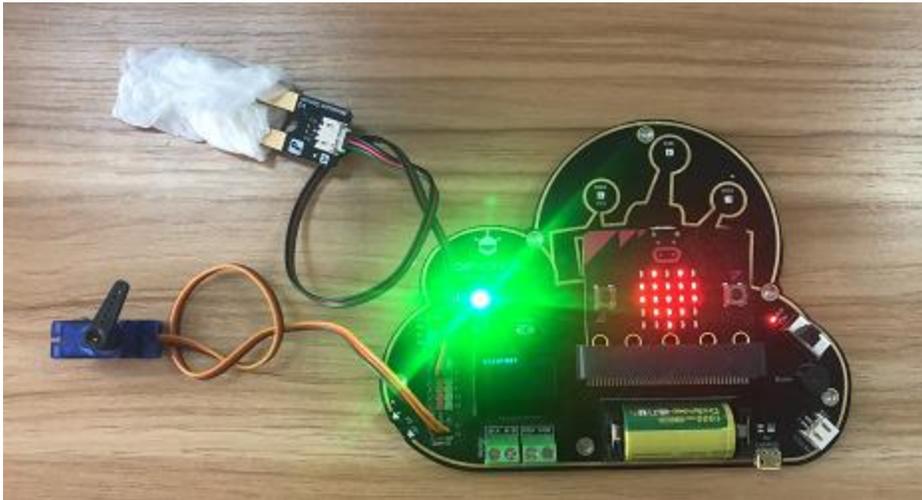
Program Effects:

When Easy IoT sends "ON", the servo rotates to 160 degrees to stretch the clothes hanger; when Easy IoT sends "OFF", the servo rotates to 30 degrees to retract the clothes hanger. Or we can control it manually by pressing the button A and B.



### Latest news

Time	Message
2019/9/17 15:48:16	OFF
2019/9/17 15:48:1	ON
2019/9/17 15:47:20	trsd:2



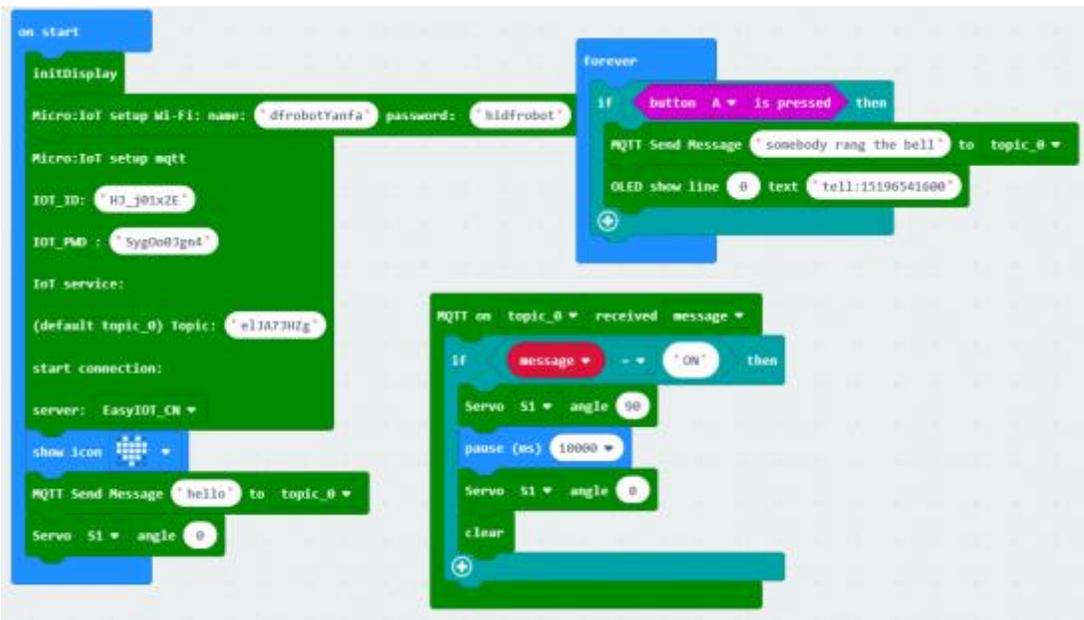
## Remote Entrance Guard System

Makecode programs: [remote entrance guard system](#)

Function description: imagine when you are on business, your friend suddenly calls you and says he is waiting at the door of your house, but you cannot go home immediately. However, you have this remote entrance guard system, so you can just use your phone to open the door and let he enter your house.

Component: servo (S1)

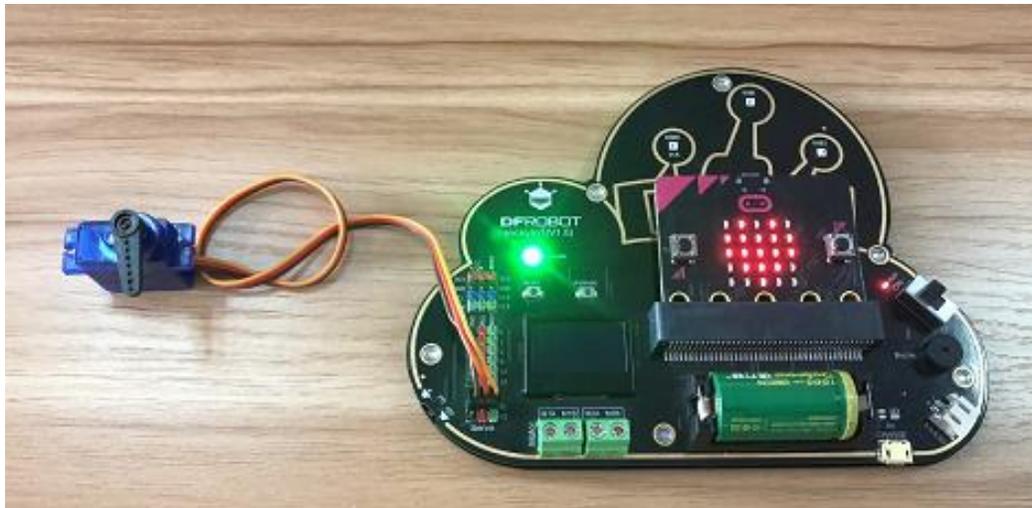
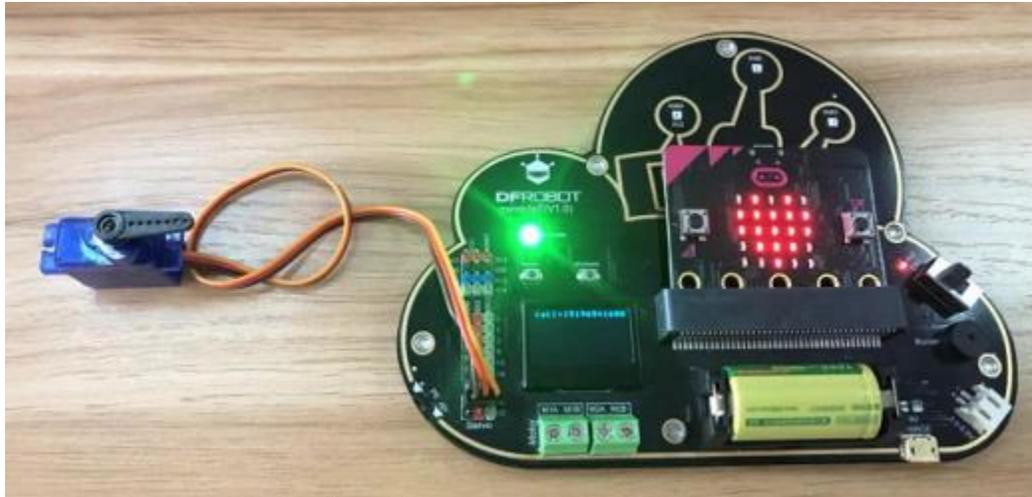
### Programs for Reference



Programs Effect:

Press down button A, the telephone number will be displayed. If the house master gets the phone call, he will send command "ON" via Easy IoT to the guard system, the servo rotates to 90 degrees to open the door. 1 minute later, the servo rotates to 10 degrees, the door will be closed.

Latest news	
Time	Message
2019/9/18 10:28:51	ON
2019/9/18 10:27:56	somebody rang the bell
2019/9/18 10:27:56	somebody rang the bell



*Remote controlled RGB LEDs*

Makecode Programs: remote controlled RGB LEDs

Function Description: you were gonna throw a home party, unfortunately, you have to work overtime, so you told your friends to start the party themselves. They want to turn on the festival lamps but cannot find the remote controller. This time you use you phone to turn the lamp on.

**Programs for reference**

```

on start
  turn off all leds
  Micro:IoT setup Wi-Fi: name: "dfrobotYanfa" password: "hidfrobot"
  Micro:IoT setup mqtt
  IDT_ID: "H3_j01x2E"
  IDT_Pwd: "9yg0e02gn4"
  IoT service:
  (default topic_0) Topic: "e11A7JH2g"
  start connection:
  server: EasyIoT_CN
  show icon
  MQTT Send Message "hello" to topic_0
  set RGB to 0

MQTT on topic_0 received message
  if message == "L" then
    call L
  if message == "B" then
    call music

music
  repeat 6 times
  do
    play tone Middle C for 1 beat
    show color
    play tone Middle C for 1 beat
    show color
    play tone Middle G for 1 beat
    show color
    play tone Middle A for 1 beat
    show color
    play tone Middle G for 1 beat
    show color
    play tone Middle G for 2 beat
    show color
  turn off all leds

L
  repeat 255 times
  do
    if RGB < 4 then
      set led index RGB color red
      pause (ms) 1000
      set led index RGB color purple
      change RGB by 1
    if RGB == 4 then
      set RGB to 0
      turn off all leds
  
```

Programs Effect:

Send a command "L" via Easy IoT, to dispaly an effect of water lamp (alternatively emit red and purple). When the module receives the command "B", it will play the music "twinkle, twinkle, little start" and the RGB shines with the music.

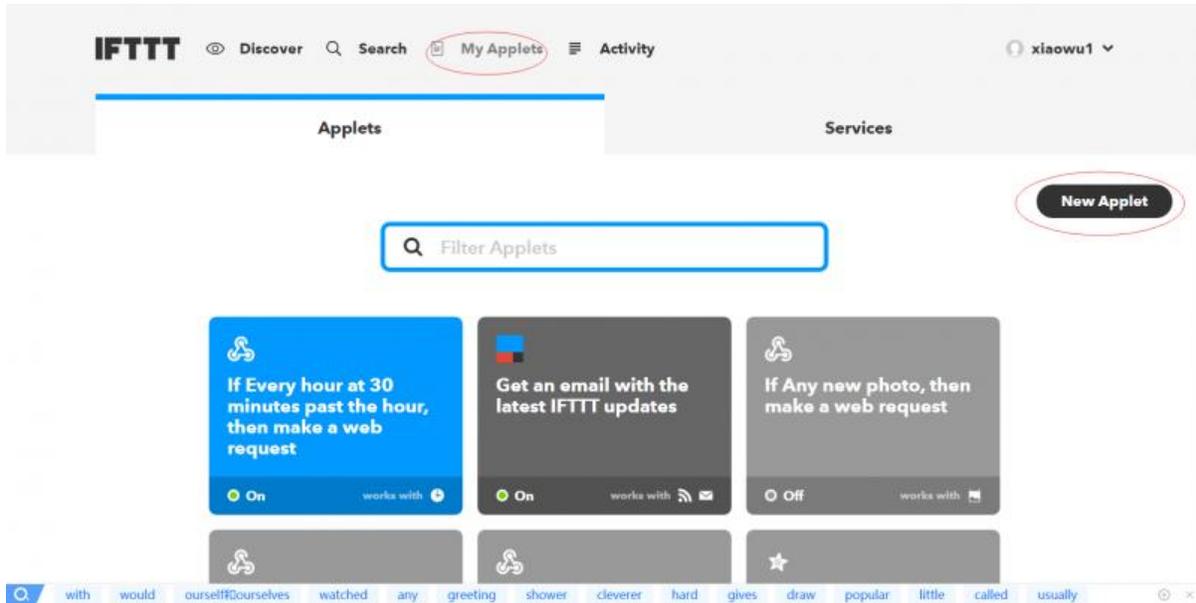
### Latest news

Time	Message
2019/9/18 14:55:25	L
2019/9/18 14:55:13	B
2019/9/18 14:55:11	hello

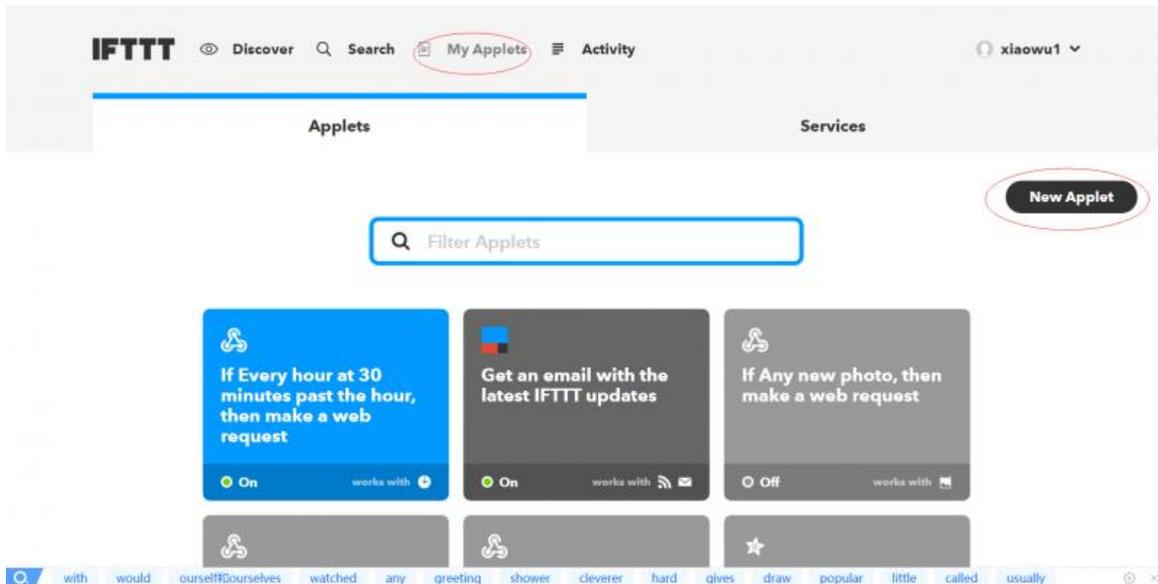


## Preparation

1. Enter IFTTT website, register an account. Website: <https://ifttt.com>
2. Sign in and enter the My Applets page, click New Applets.



3. Create an Applet, enter the Applet interface, click This and input webhooks.





## Choose a service

Step 1 of 6



4. Fill the Event Name in the next page.



## Complete trigger fields

Step 2 of 6

### Receive a web request

This trigger fires every time the Maker service receives a web request to notify it of an event. For information on triggering events, go to your Maker service settings and then the listed URL (web) or tap your username (mobile)

Event Name

The name of the event, like "button\_pressed" or "front\_door\_opened"

Create trigger

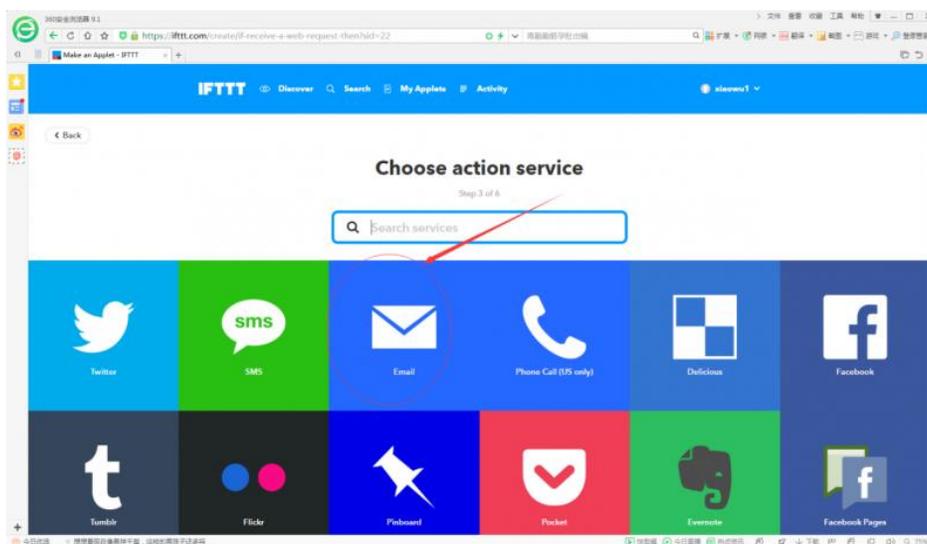
IFTTT Basic Tutorial

Send data to email

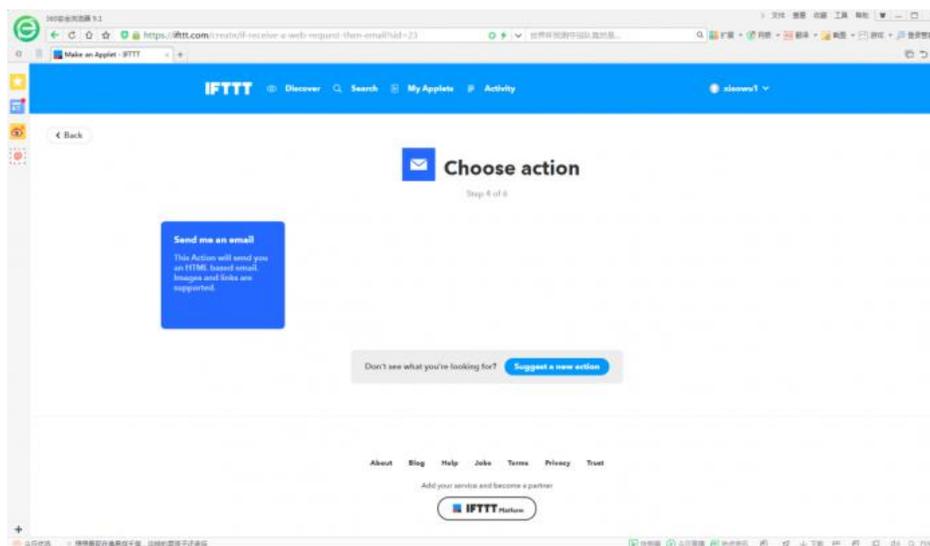
Function description: press the button to send a message to email.

- Configure IFTTT, set the event name as "aaa", then webpage will return back automatically. Click "That" and select "Email" in the next page.

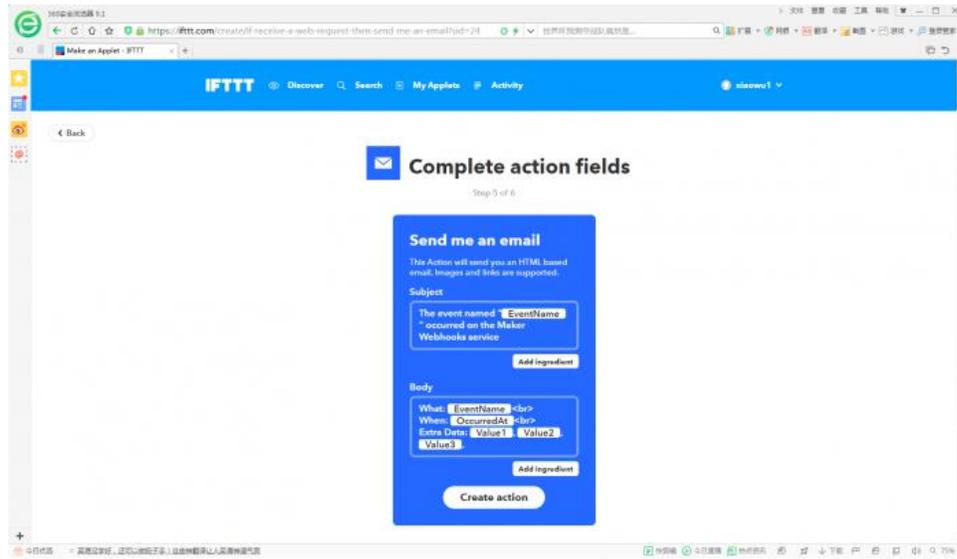
If  Then  That



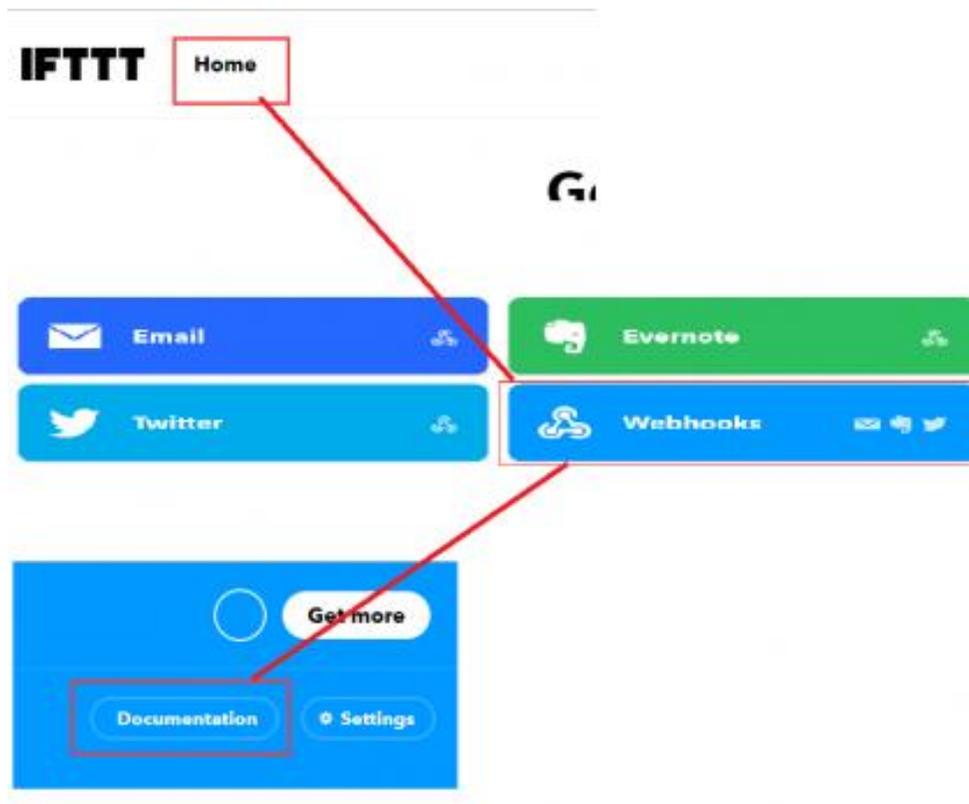
- Select "Send me an Email"



- Edit the content, here I chose the default one. Then click "creation action"



- To check the password, click HOME to enter "webhooks" and click "Documentation".



Documentation

Your key is: **dn6c2aLMKMXIB7E62gpO2D**

◀ Back to service

### To trigger an Event

Make a POST or GET web request to:

[https://maker.ifttt.com/trigger/\[event\]/with/key/dn6c2aLMKMXIB7E62gpO2D](https://maker.ifttt.com/trigger/[event]/with/key/dn6c2aLMKMXIB7E62gpO2D)

Makecode Programs: Send data to Email

```
on start
  initDisplay
  Micro:IoT setup Wi-Fi: name: 'dfrobotYanfa' password: 'hidfrobot'
  Webhooks config: event: 'aaa' key: 'dn6c2aLMKMXIB7E62gpO2D'

on button A pressed
  IFTTT(post)
  value1 'HI'
  value2 'DFRobot' to topic_0
  value3 ''
  timeout(ms) 10000
  OLED show line 0 text 'HI'
  OLED show line 1 text 'DFRobot'
```

- Result: press down button A, the message "HI DFROBOT" will be sent to the pre-set Email.

## The event named "aaa" occurred on the Maker Webhooks service ☆

**Webhooks via IFTTT** 16:52

---

What: aaa  
When: September 18, 2019 at 04:52PM  
Extra Data: HI, DFRobot, ,

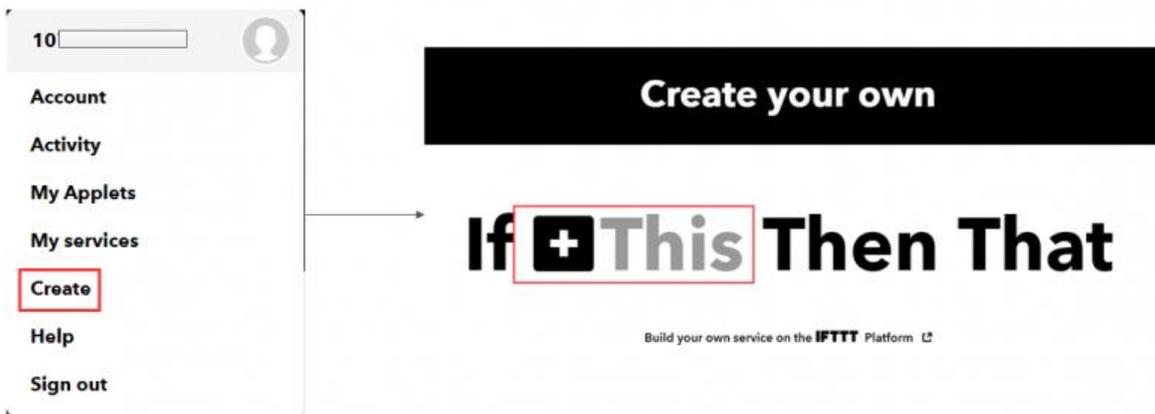
 **If Maker Event "aaa", then Send me an email at 1078883988@qq.com**

[Unsubscribe](#) from these notifications or [sign in](#) to manage your [Email service](#).

### Send note to Evernote

Function Description: send note to Evernote

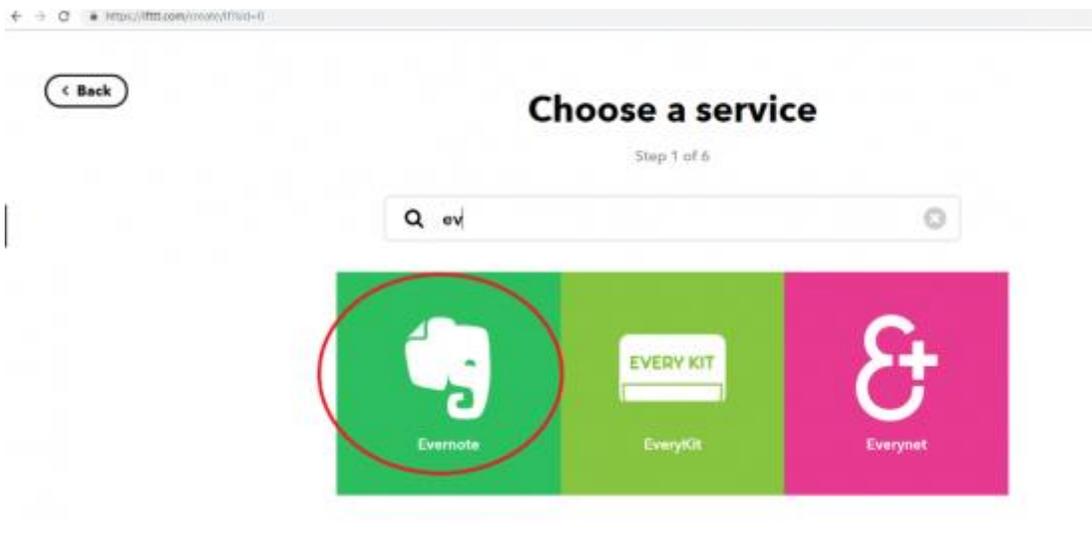
- Create a new This. Click Account->Create->This, and create webhooks and name the Event as "kkk".



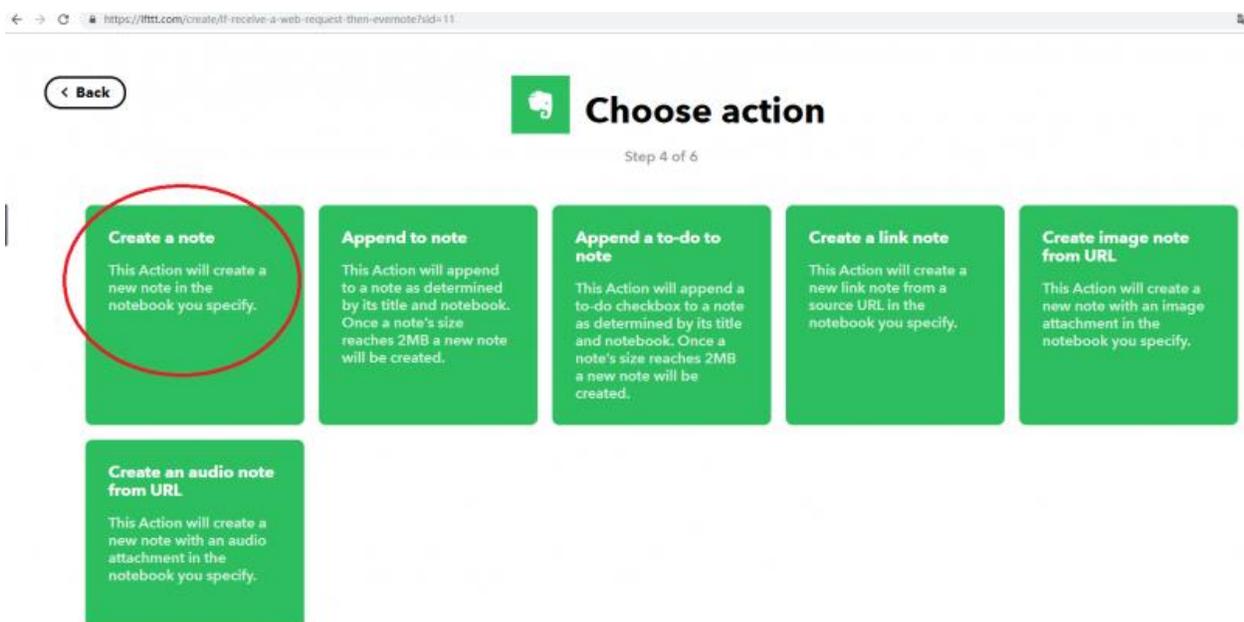
The image shows a screenshot of the IFTTT user interface. On the left, a vertical menu is visible with the following items: 'Account', 'Activity', 'My Applets', 'My services', 'Create' (highlighted with a red box), 'Help', and 'Sign out'. An arrow points from the 'Create' button to a large black button labeled 'Create your own'. Below this button is the IFTTT logo, which consists of the text 'If + This Then That' where the plus sign is inside a square. At the bottom of the logo area, it says 'Build your own service on the IFTTT Platform' with a small external link icon.

- The webpage will back to the last page, then click That and search Evernote.

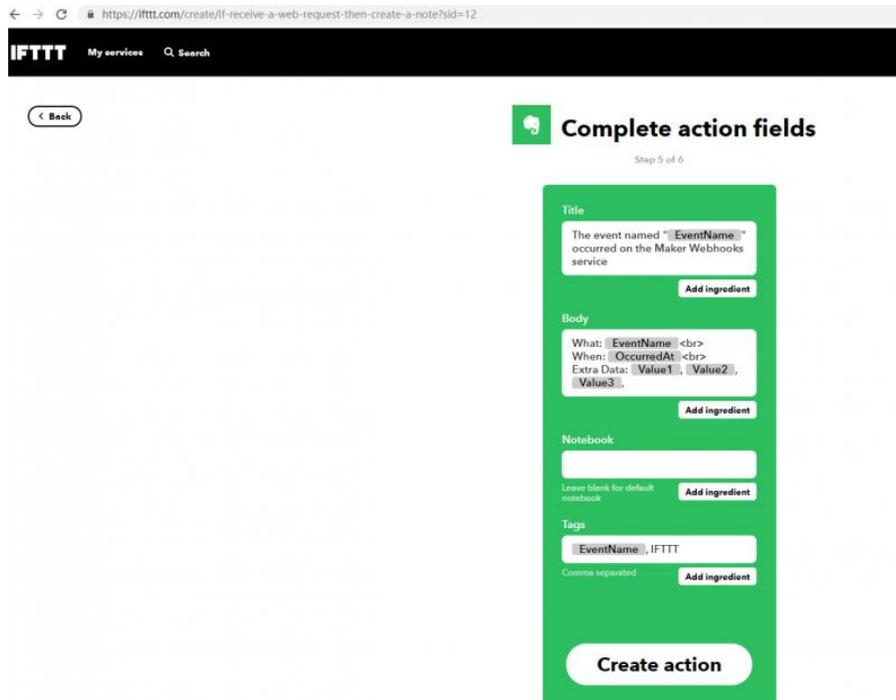
# If Then That



- Select what you like, I chose Create a note here.



- Select the default form



- Click "Creation action", the password is the same as the previous Email project.

Makecode Programs: [send date to Evernote](#)

### Programs for reference

```
on start
  initDisplay
  Micro:IoT setup Wi-Fi: name: "dfrobotYanfa" password: "hidfrobot"
  Webhooks config: event: "kkk" key: "dN6c2aLHKMXI87E62gp02D"

on button A pressed
  IFTTT(post)
    value1 "HI"
  MQTT Send Message value2 "DFRobot" to topic_0
    value3 ""
    timeout(ms) 10000
  OLED show line 0 text "HI"
  OLED show line 1 text "DFRobot"
```

- Result: press down button A, send "HI DFROBOT" to the Evernote.

## The event named "kkk" occurred on the Maker Webhooks service

What: kkk

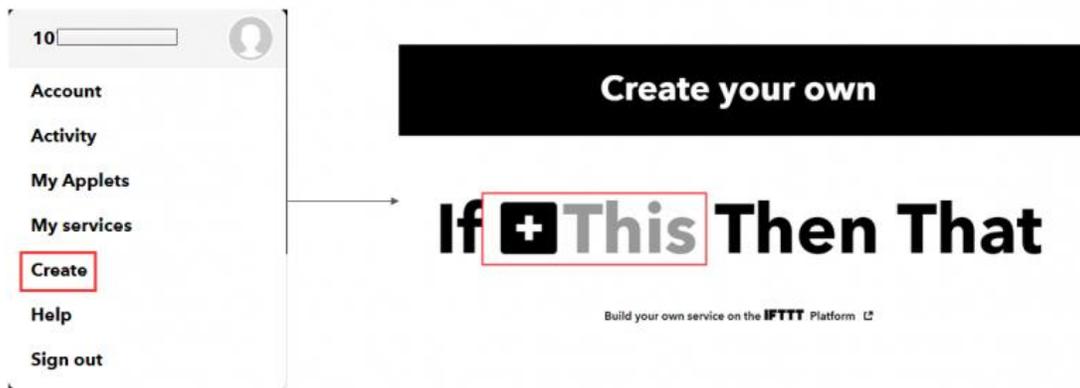
When: September 3, 2019 at 04:05PM

Extra Data: HI, DFRobot, }

### Send Message to Twitter

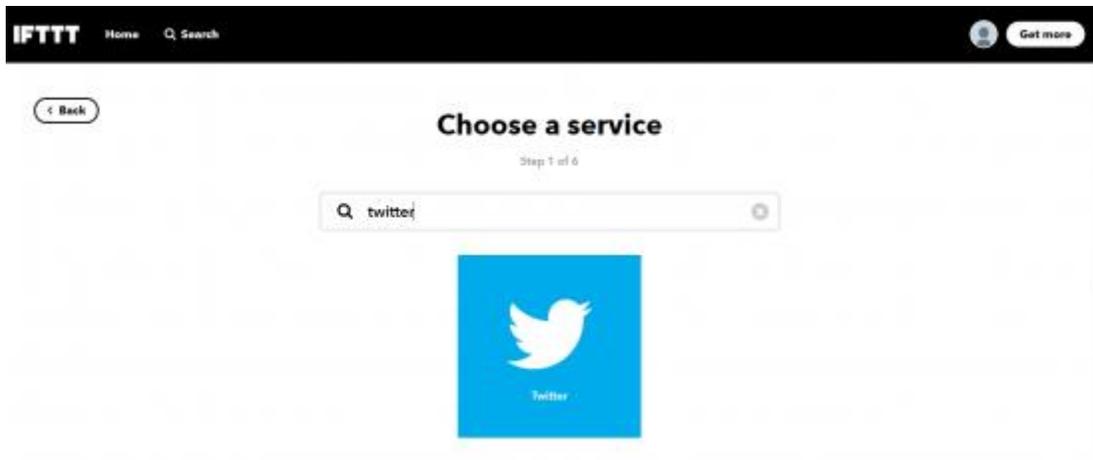
Function description: when you get up in the morning, press down button A and B, send the temperature and light intensity to Twitter.

- Create a new This. Click Account->Create->This, and create webhooks and name the Event as "Twitter".

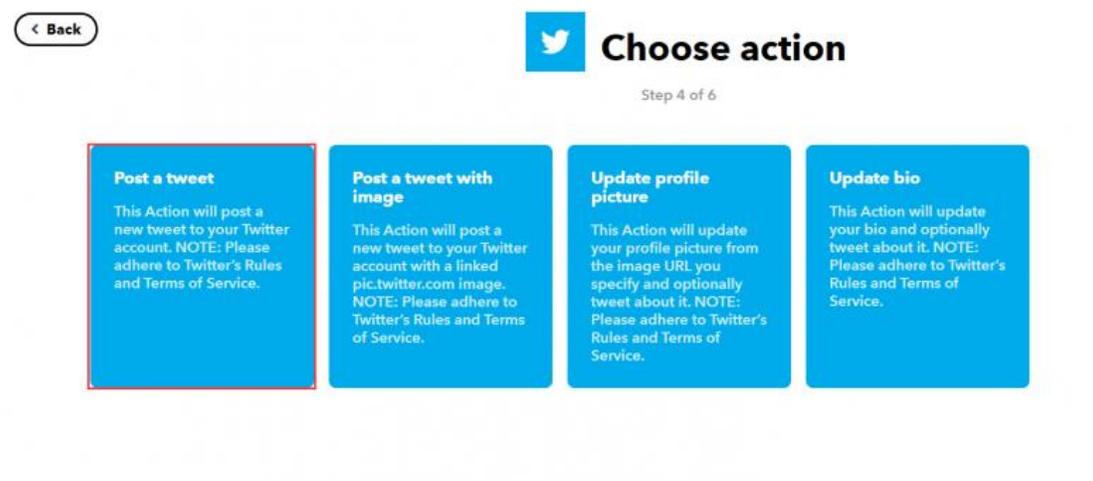


- The webpage will back to the last page, then click That and search Twitter.

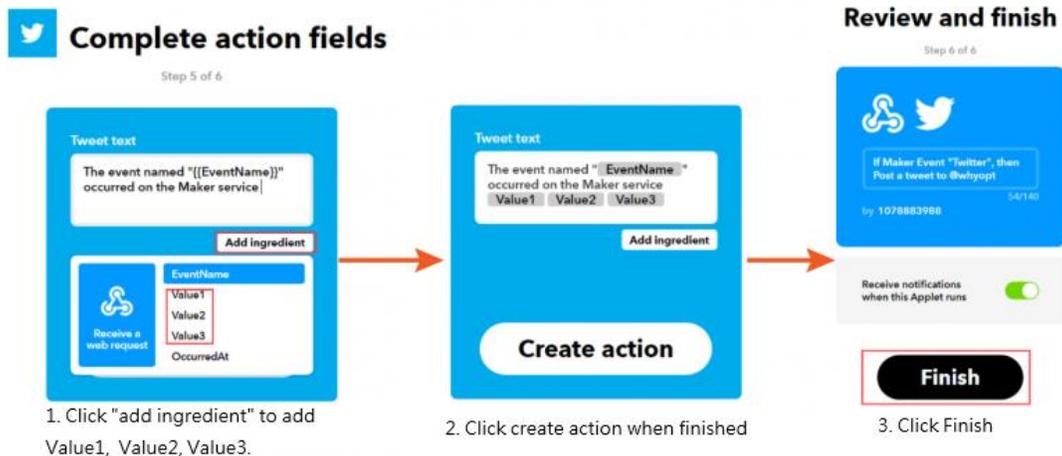




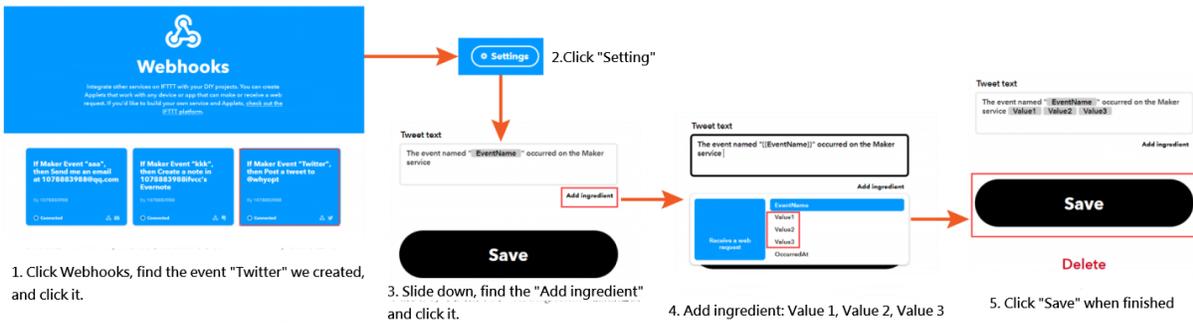
- Select what you like, I chose "post a tweet" here.



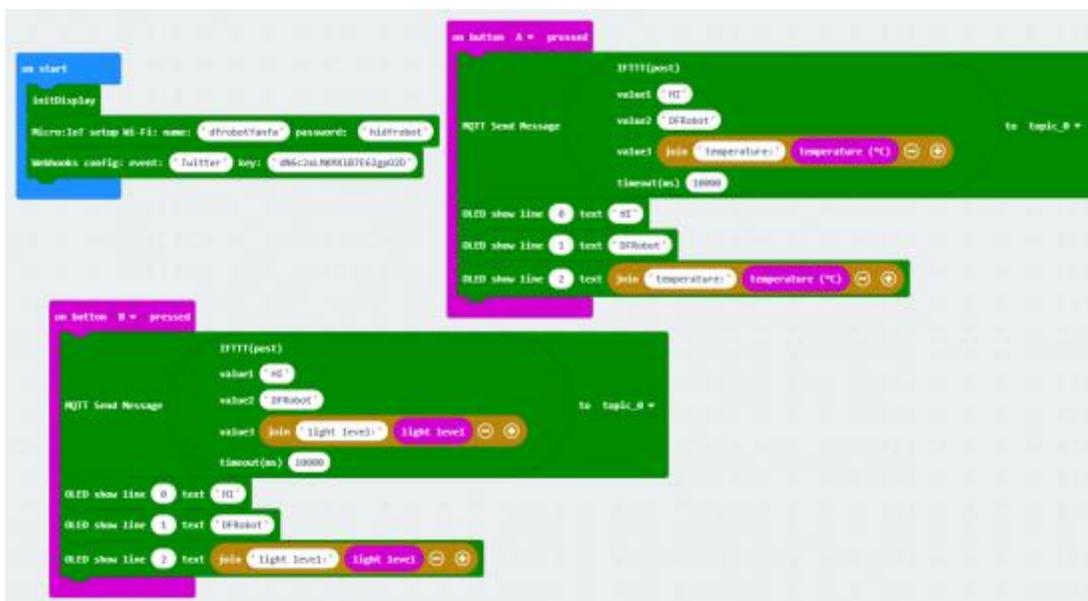
- Complete the following steps after that.



- If accidentally selected the default setting, add the Event as the following steps.



### Makecode Programs: Send message to Twitter



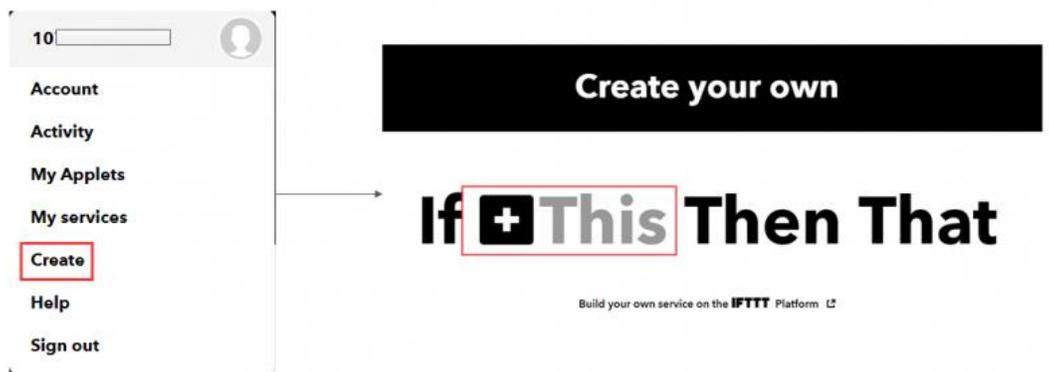
- Result: press down button A, send the message to Twitter: Hi DFROBOT and the current temperature; Press down button B, send: HI DFROBOT, and the current light intensity.



## Send Message to iPhone

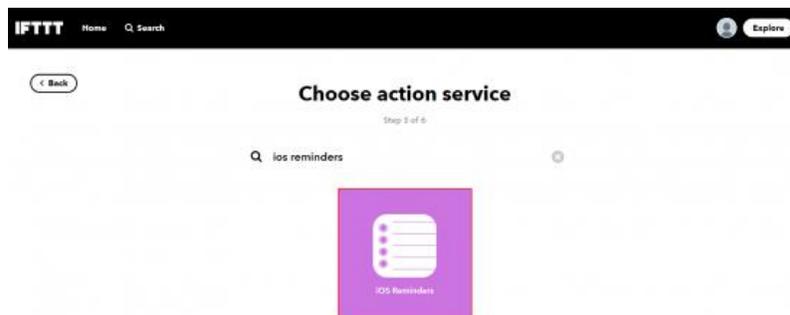
Function description: the climate is changeable during the seasonal alteration. Use the micro:bit to detect temperature. If temperature is more than 30 degree, send a message to iphone to notice the user to pay attention to heatstroke. If temperature is less than 20 degree, send a message to iphone to notice the user to protect themselves from the cold.

- Create a new This. Click Account->Create->This, and create webhooks and name the Event as "Reminder"

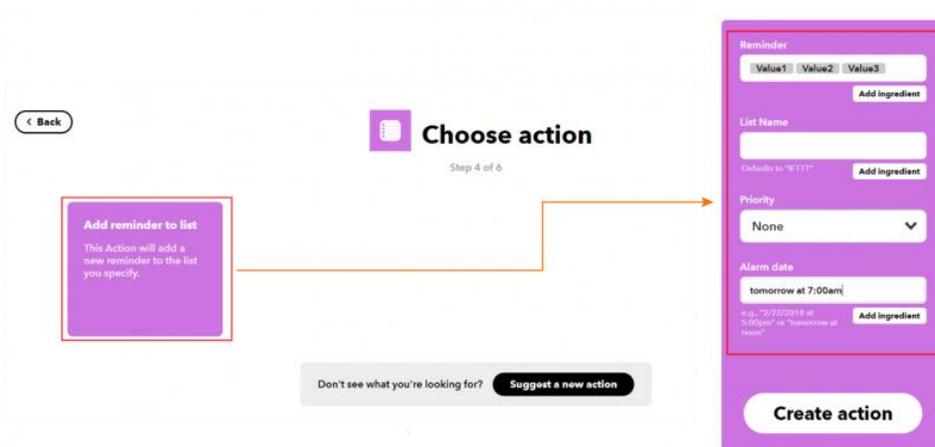


- The webpage will back to the last page, then click That and search IOS Reminders

If  Then  That



- Select what you like, I chose "add reminder to list" here.



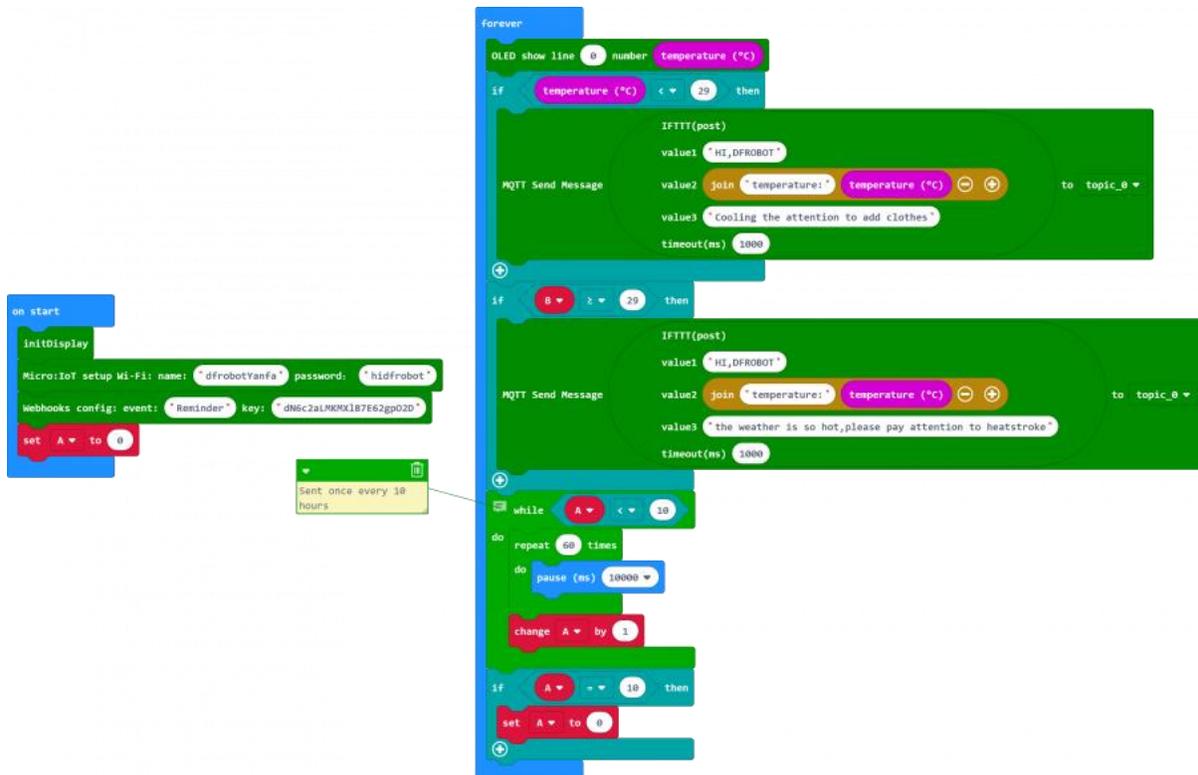
1. Click "add reminder to list"

2. Add ingredient under "Reminder". List Name and Priority keep the default status. Set Alarm date to "tomorrow at 7:00 am."

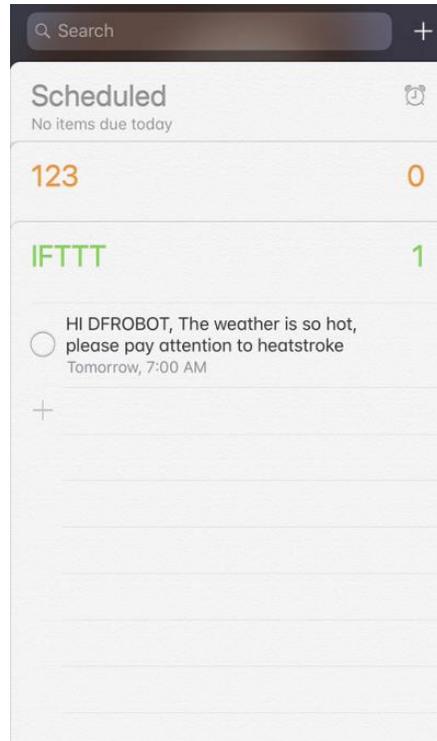
Note: the alarm date can be set according to your actual use.

Makecode: [Send a message to iPhone](#)

## Programs for reference



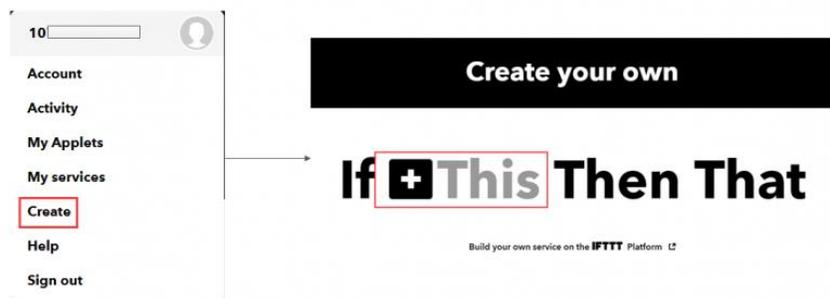
- Result: Send a message to iPhone every 10 hours. When Temperature > 30, send: the weather is so hot, please pay attention to heatstroke. When temperature < 20, send: Cold outside, please put more clothes on.



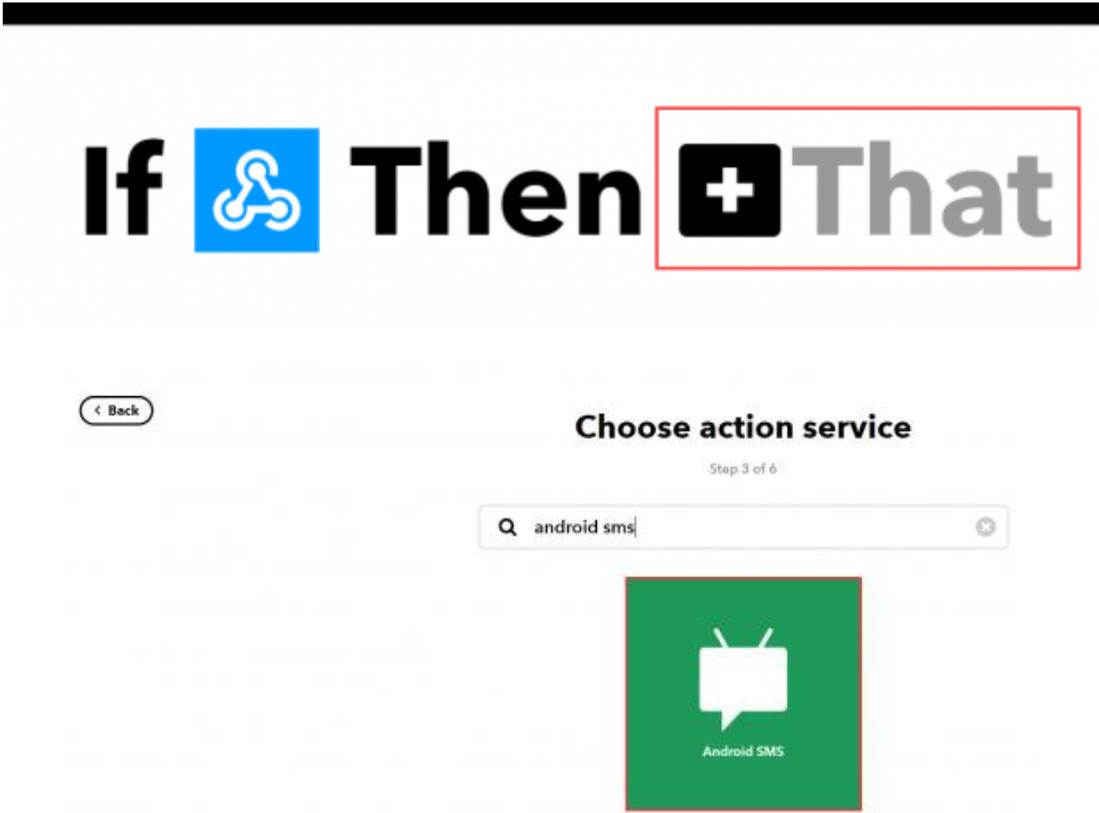
*Use an Android phone to Send Message*

Function description : We often forget to water out plants, sometime, you have to be on business for a long time, but you want someone to take care of your plants. We can use the IoT expansion board to make an application to solve this problem. When the soil moisture is less than 350, the system automatically sends a message to your friend to water the plants. If it is over 520, send a message that the moisture is suitable.

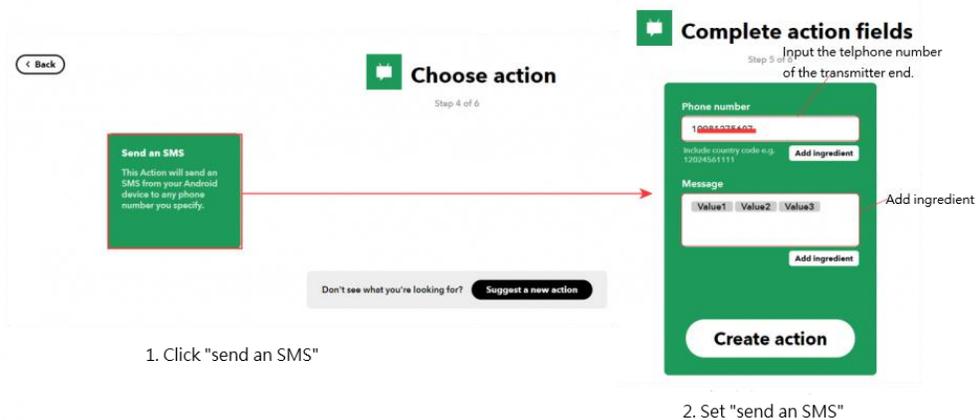
- Create a new This. Click Account->Create->This, and create webhooks and name the Event as "message"



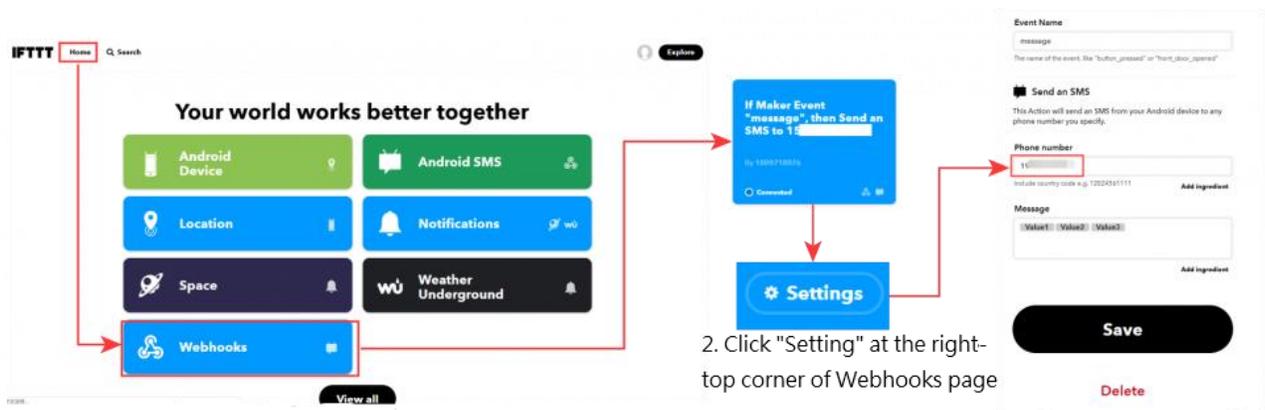
- The webpage will back to the last page, then click That and search "Android sms"



- Select what you like, I chose "send an sms" here.



- Set the phone number to receive the message -Enter->Home->webhooks->add the phone number to receive phone number

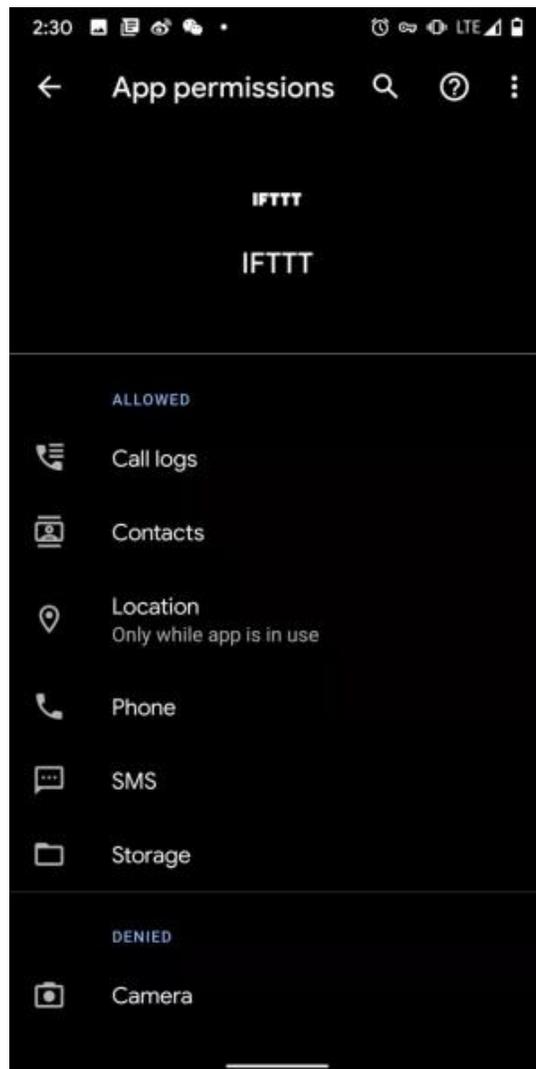


1. Click "home" and click Webhooks

2. Click "Setting" at the right-top corner of Webhooks page

3. Slide down, input the telephone number to receive messages.

Note: the type of the phone to send messages must be Android, and there should be a IFTTT app installed on the phone. IFTTT needs to have the access permission to Message.

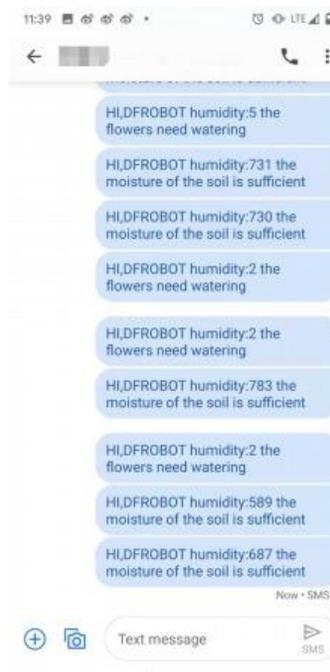


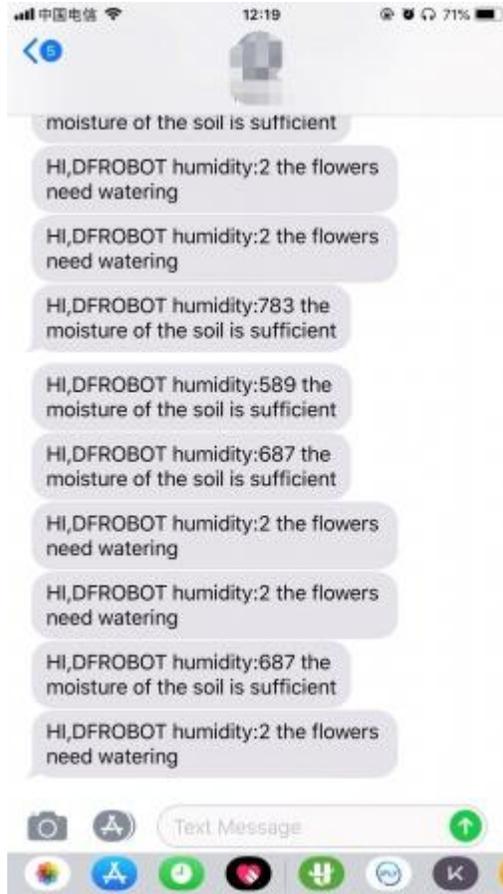
## Makecode Programs: [Send messages to Android phone](#)

```
on start
  initDisplay
  Micro:IoT setup MI-F1: name: dfrobotvafn password: hidfrobot
  webhooks config: event: message key: ch704eINTEy0vn1aQ1PKxp
  set A to 0
  set B to 0

forever
  set humidity to analog read pin P1
  OLED show line 0 text join "humidity:" humidity
  if humidity < 350 and A == 0 then
    IFTTT(post)
    value1 HI,DFROBOT
    value2 join "humidity:" humidity
    value3 "the flowers need watering"
    timeout(ms) 1000
    set A to 1
    set B to 0
  if humidity > 520 and B == 0 then
    IFTTT(post)
    value1 HI,DFROBOT
    value2 join "humidity:" humidity
    value3 "the moisture of the soil is sufficient"
    timeout(ms) 1000
    set A to 0
    set B to 1
```

- Result: when moisture < 350, send: the flowers need watering; When moisture > 250, send: the moisture of the soil is sufficient.





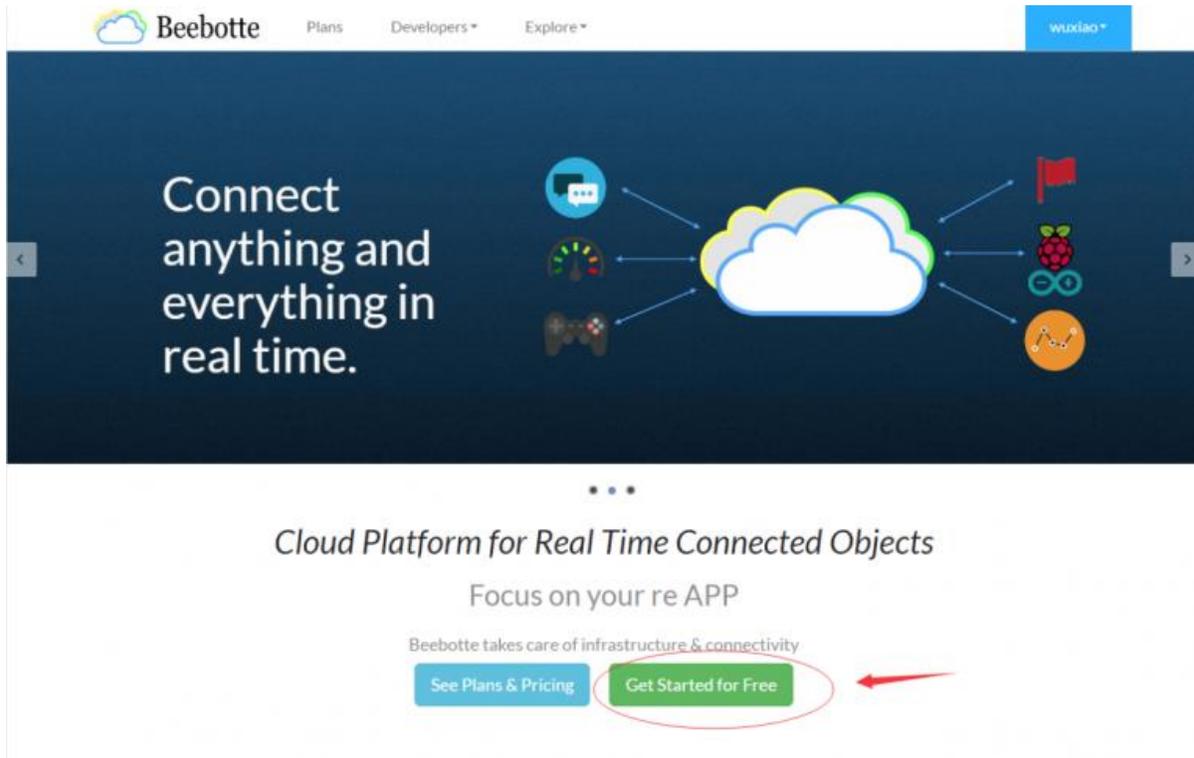
### *Beebotte IoT Platform*

- Beebotte is a Cloud Platform for Real Time Connected Objects
  - Connecting anything and everything in real-time using a rich API supporting REST, WebSockets and MQTT
  - Designed to empower Internet of Things and real-time communicating applications
  - Beebotte brings you a platform as a service connecting thousands of objects and delivering millions of messages
  - One platform suited for diverse applications like instant messaging, dashboards, online gaming and score boards, domotics, Internet of Things and reporting
  - Seamless scalability to meet your growing demands
- How to use Beebotte

### **Send data configuration from MQTT to Beebotte**

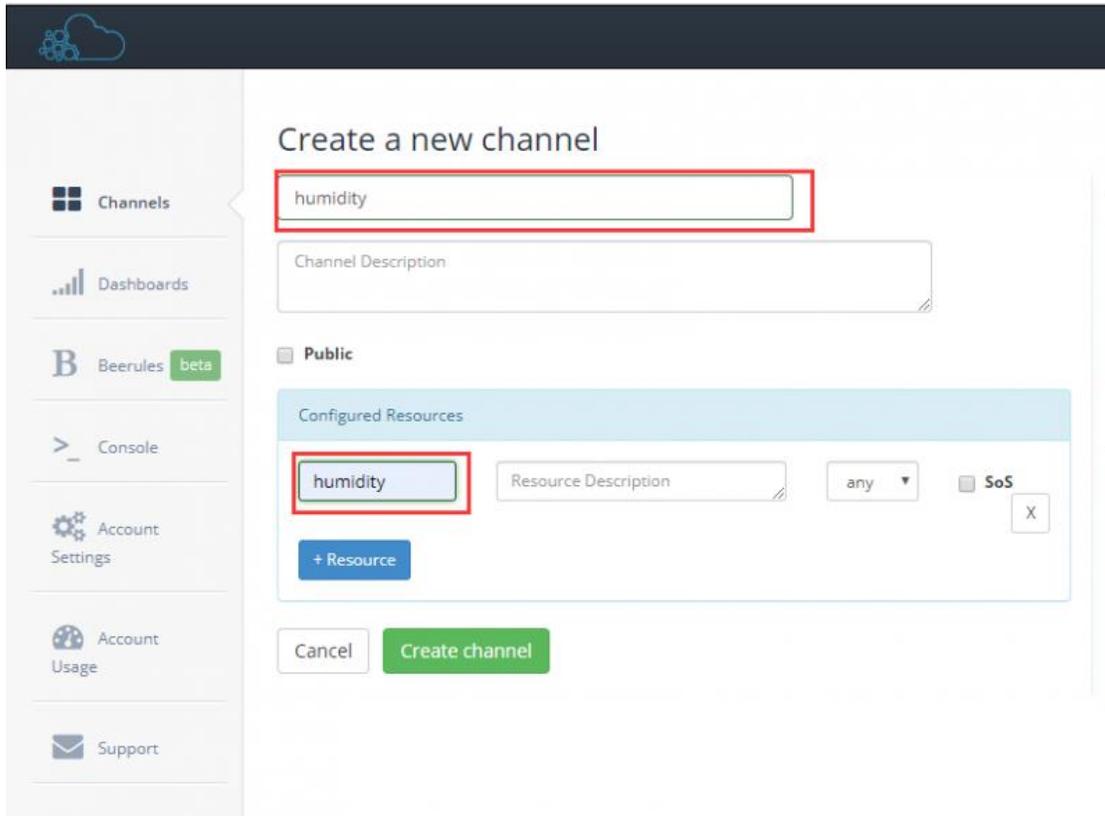
- Website: <https://beebotte.com>

- Enter the Website, register an account. Then Back to Homepage and click Get Started For Free

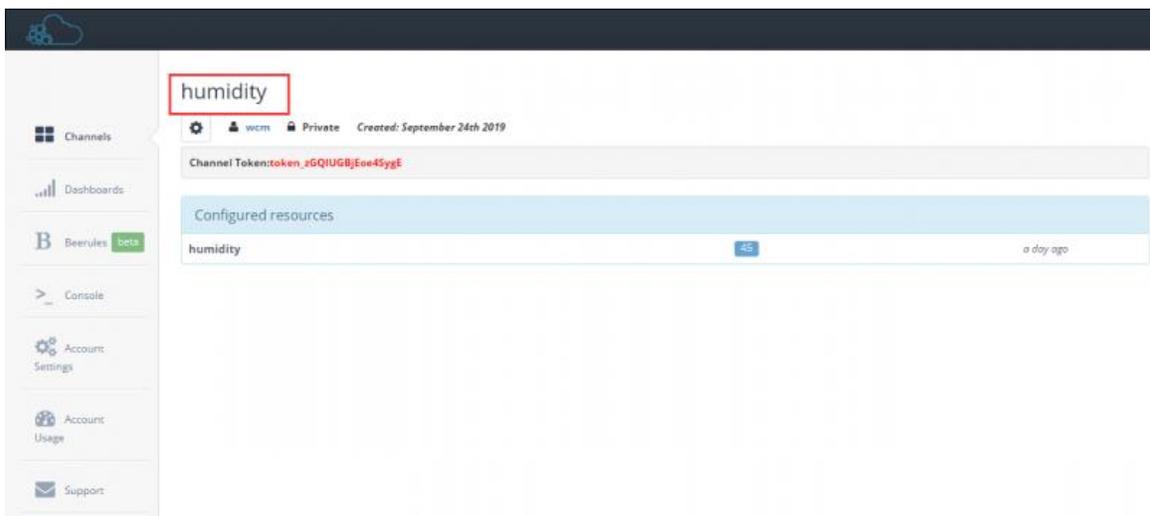


- Enter Create New Interface to create your Channel Name, Channel Description, Resource name, Resource Description. You can create more than one Resource.

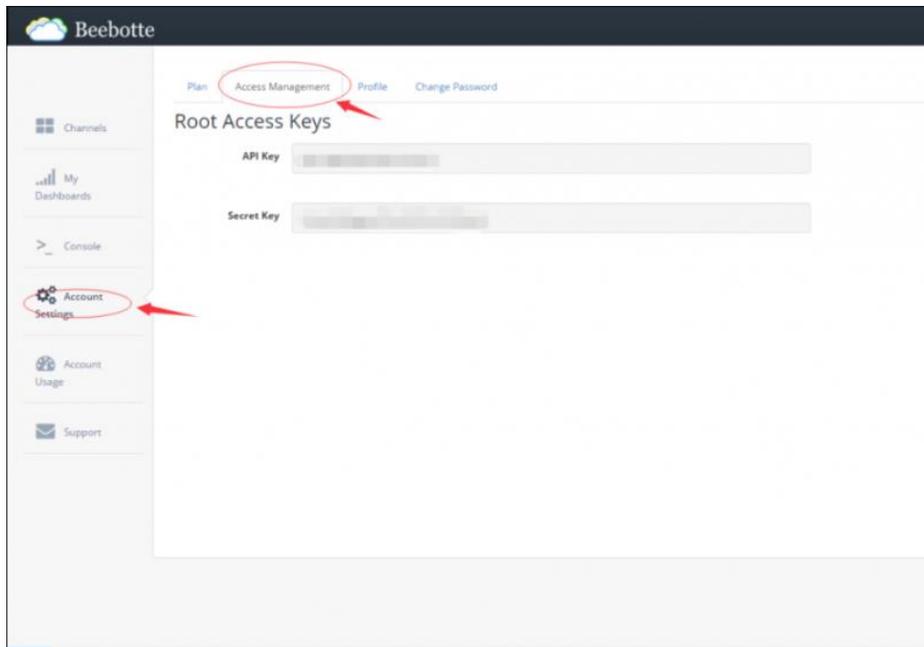
- For example, create a humidity channel to receive data



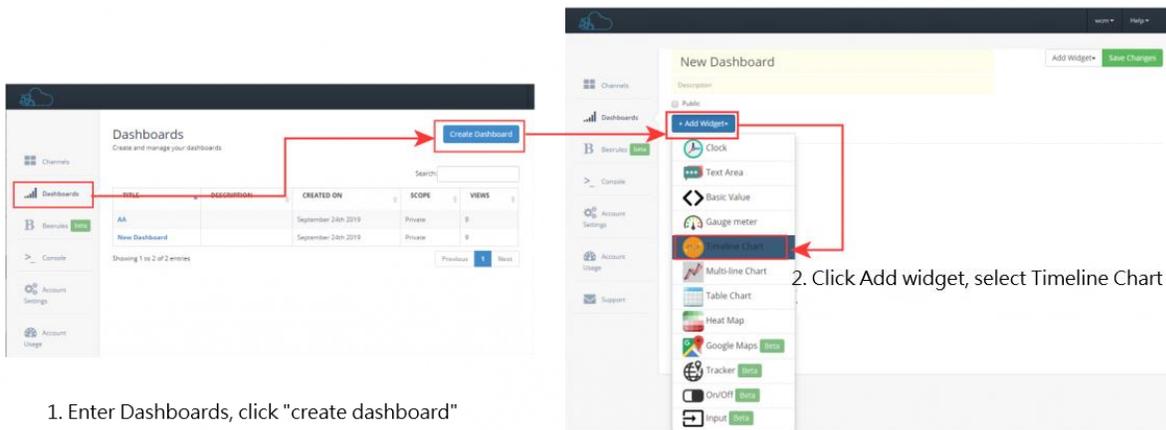
- The following interface will appear when you completed.



- Click Account Settings->Access Management to check the API Key and Secret Key

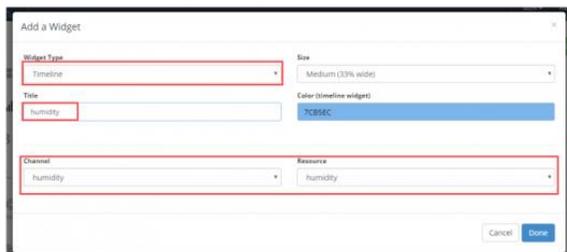


- Create Dashboards to check the history of the update value. -Create time table



1. Enter Dashboards, click "create dashboard"

2. Click Add widget, select Timeline Chart



3. Select the related options according to content of chart.  
Note: choose the correct channel and resource. (The two should be the same.)



4. Change the title to "humidity" and click "save changes".

## Record the data of the Soil Moisture

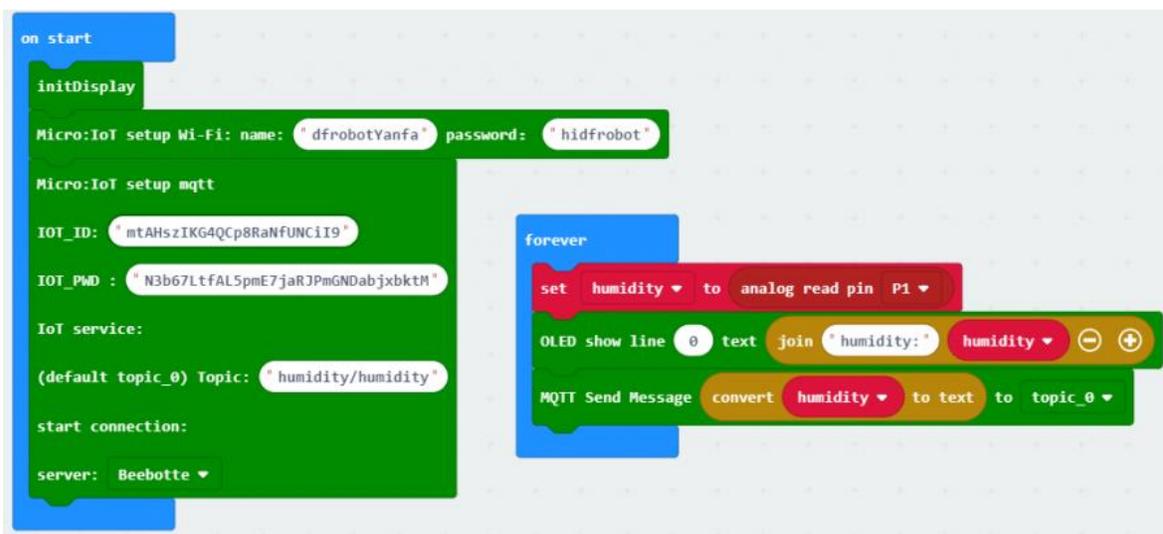
Function description: Create a channel-humidity and time table. When connected with Internet, the moisture will be updated to Beebotte platform, and we will check the change of the moisture through the curve graph in Dashboards.

Makecode Programs: [Record the data of the Soil Moisture](#)

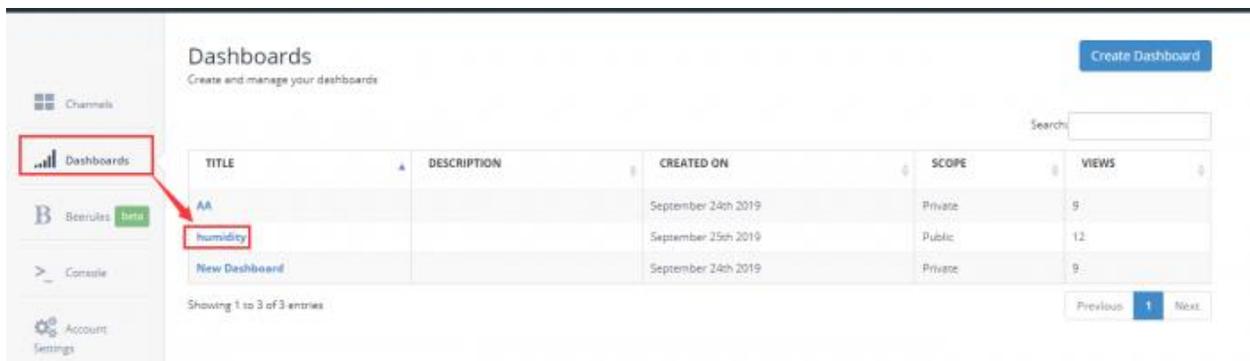
## Programs for reference

Note: IoT\_id should be API Key; IoT\_pwd should be Secret Key; TOPIC is the name of Channel/Resource, there is a / between them.

Chose Beebotte as the server in the Program.



- Record the data of the Soil Moisture: enter Dashboards, click humidity check the data.



## Basic Tutorials

### *Drive a motor*

makecode programs: [drive a motor](#)

#### **Programs for reference:**

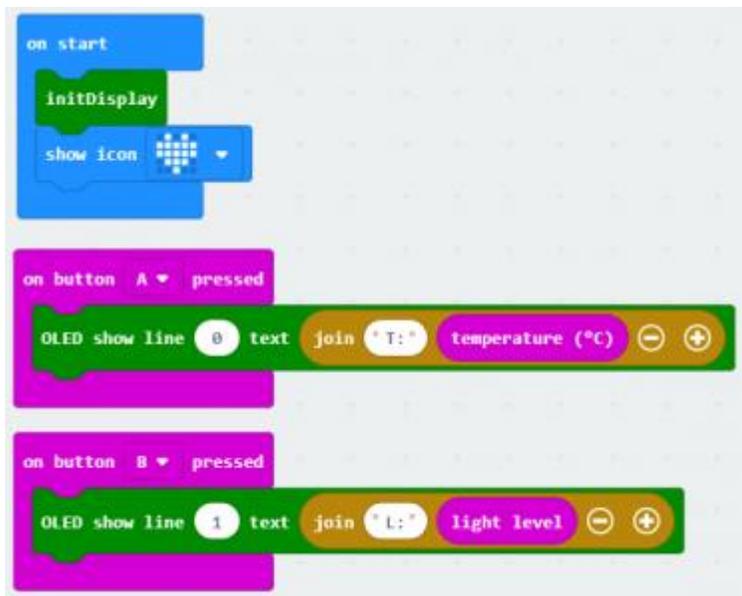


Result: the motor rotates forward at speed of 255 for 1 second, and then rotate backward for 1 second.

### *Light and Temperature Monitor*

makecode programs: [light and temperature monitor](#)

#### **Programs for reference:**



Function description: use micro:bit to detect the light and temperature value, display the values on the OLED.

Results: OLED displays the current temperature and light in real-time.

FAQ

---

For any questions, advice or cool ideas to share, please visit the [DFRobot Forum](#)

More Documents

---