

MRT-Pthon 教学

设备要求

电脑一台，建议使用带有WiFi的笔记本或者台式机

MRT6一台

MRT-Python软件介绍

电脑系统要求

Windows 7 x64 /x86_32bit

Windows10 x64 /x86_32bit

软件安装

1 根据自己系统 选择不同版本的软件

MRT_Python64bitV03016.rar

MRT-Python32BitV0315.rar

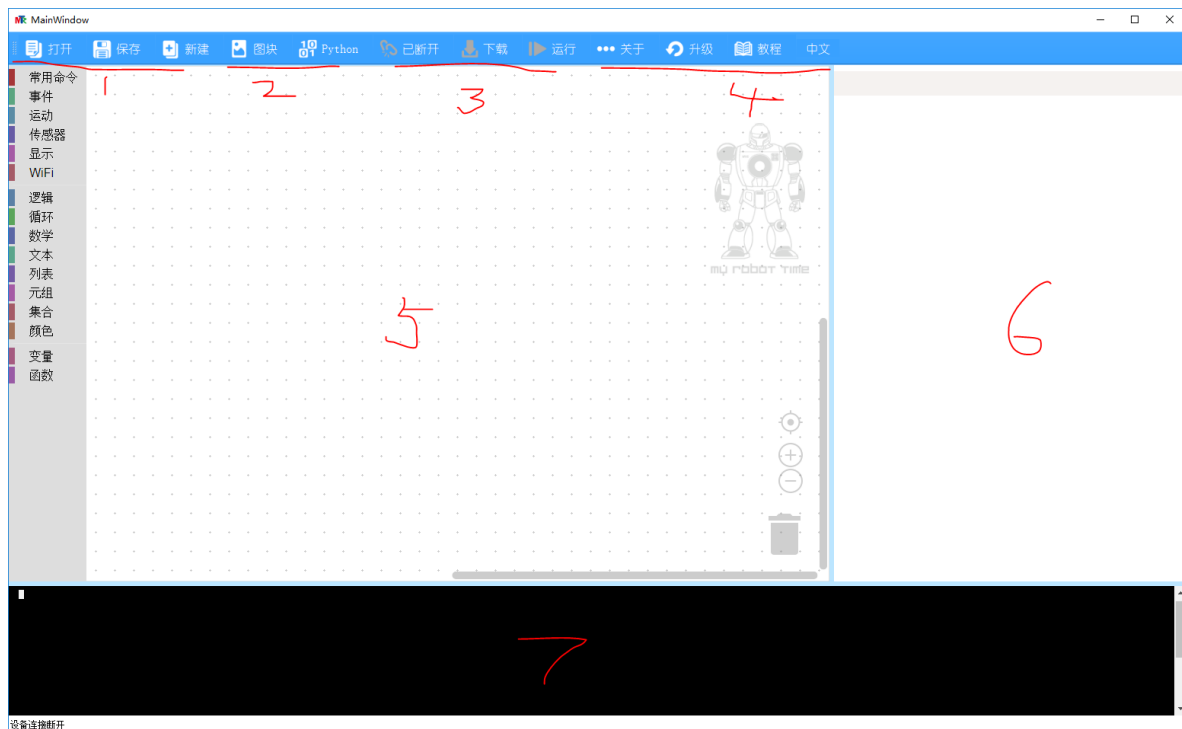
2 将软件解压到 电脑的英文路径中（中文路径暂不支持）如下图

1：英文路径（英文目录下即可，可以根据需要改变）

2：进入安装文件目录 双击或者点击右键打开 **MRT_Python.exe** 运行

3：选中 MRT_Python.exe 点击右键，创建**快捷键方式**，复制放在您所需要的位置，以方便您的使用

备注：如果运行不了 请确认路径是否正确，以及软件版本是否和系统要求一致



1: 文件操作区域 **打开**--打开文件内容 **保存**---保存文件 **新建**---新建文件

2: 编程模式切换区域 **图块**--Blockly 编程模式 **Python**--Python代码编程模式

3: 设备操作区域 **已断开**---设备连接或者断开 按键 **下载** --下载当前编辑的代码到设备 **运行** ----运行已下载的程序

说明: 1 下载和运行 必须连接设备成功后才可使用

2 下载的时候 必须先下载程序后运行, 否则 运行的代码将是上一次的代码

(因可能设备要求只下载不运行代码)

4: 软件的一些信息区域 **关于**---关于软件的介绍 **升级**--升级软件的介绍 **教程**---软件的教程 **中文**---语言切换

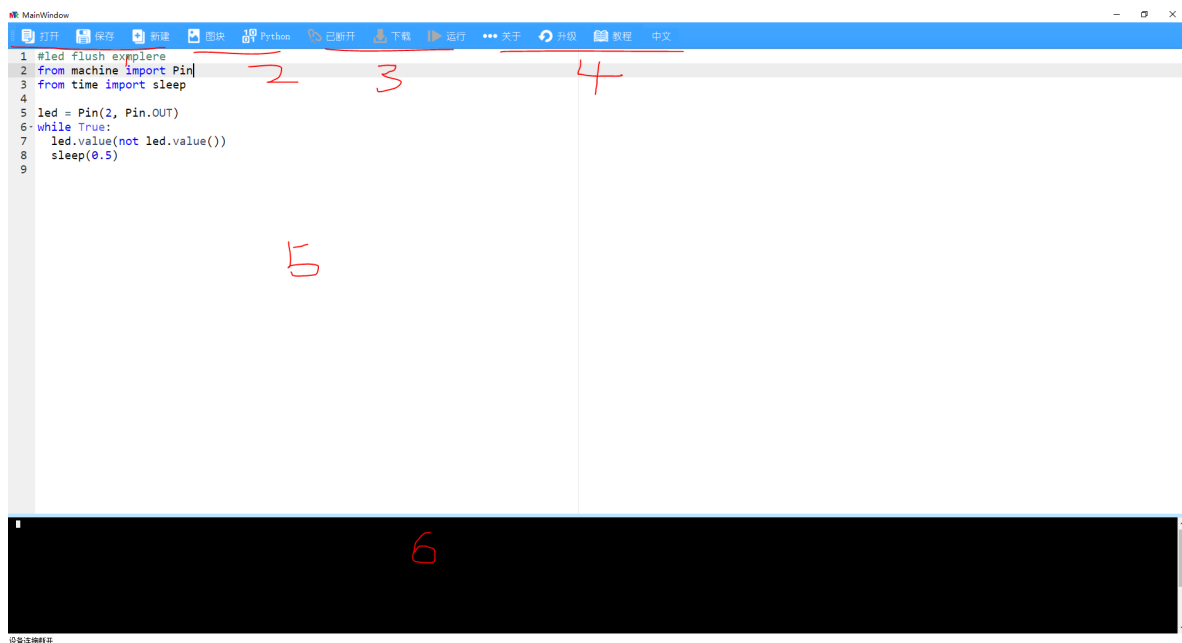
5: Blockly代码编程区域

6: Python 代码查看区域

7: MRT6设备 调试及输出窗口

8: 软件最底部为 设备状态和文件路径显示栏

Python代码编程模式



1: 文件操作区域 **打开**--打开文件内容 **保存**---保存文件 **新建**---新建文件

2: 编程模式切换区域 **图块**--Blockly 编程模式 **Python**--Python代码编程模式

3: 设备操作区域 **已断开**---设备连接或者断开 按键 **下载** --下载当前编辑的代码到设备 **运行** ----运行已下载的程序

说明: 1 下载和运行 必须连接设备成功后才可使用

2 下载的时候 必须先下载程序后运行, 否则 运行的代码将是上一次的代码

(因可能设备要求只下载不运行代码)

4: 软件的一些信息区域 **关于**---关于软件的介绍 **升级**--升级软件的介绍 **教程**---软件的教程 **中文**---语言切换

5: Python代码编程区域

6: MRT6设备 调试及输出窗口

7: 软件最底部为 设备状态和文件路径显示栏

MRT6设备介绍

开机启动

断开电源 或者插入Microusb 接口给MRT6设备供电



等待3秒左右进入开机界面



进入 是否设置WIFI界面 倒计时 10秒

可以选择 右按键 进入WIFI 连接设置界面 可以选择左按键跳过WIFI设置

10秒过后 如果设备已经连接网络则跳入 编程模式 否则一直提示 设置WIFI



设置WIFI

连接设备默认WIFI

1:将WiFi热点或者路由器设置成:

SSID:MRT-python

密码:12345678

2: 如果设备正常启动，将会自动链接到此热点

3: 电脑和设备务必处于同一个局域网,连接到同一个路由器

连接定义WIFI

按右键 选择确认 进入设置WIFI 界面



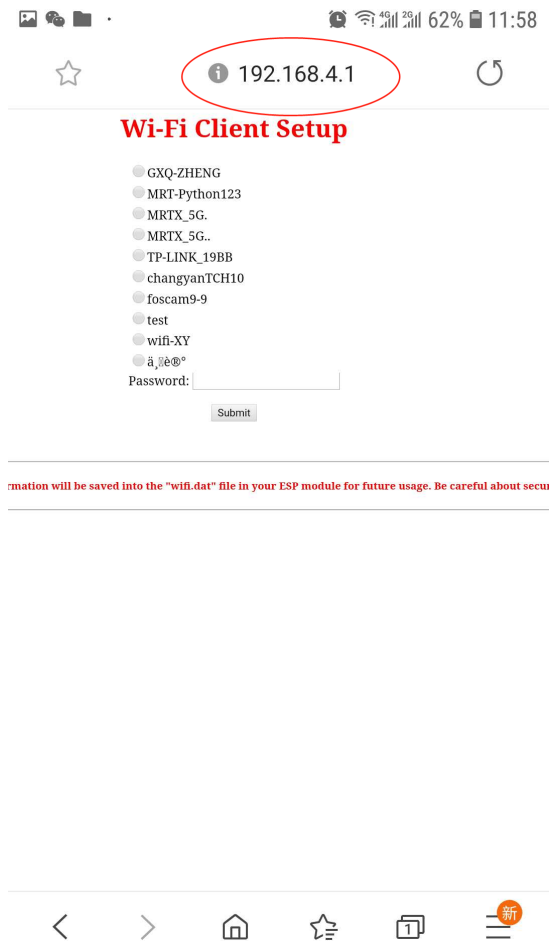
利用手机或者笔记本，WIFI 连接设备热点 MRT_XXXXX 密码：12345678

连接设备AP热点



打开浏览器

输入: 192.168.4.1



进入设置界面

选中你所需要的WIFI热点，输入密码 点击提交

此热点必须是和你电脑在同一局域网的WIFI热点

如果密码正确则 提示成功设置



然后重启设备

连接设备

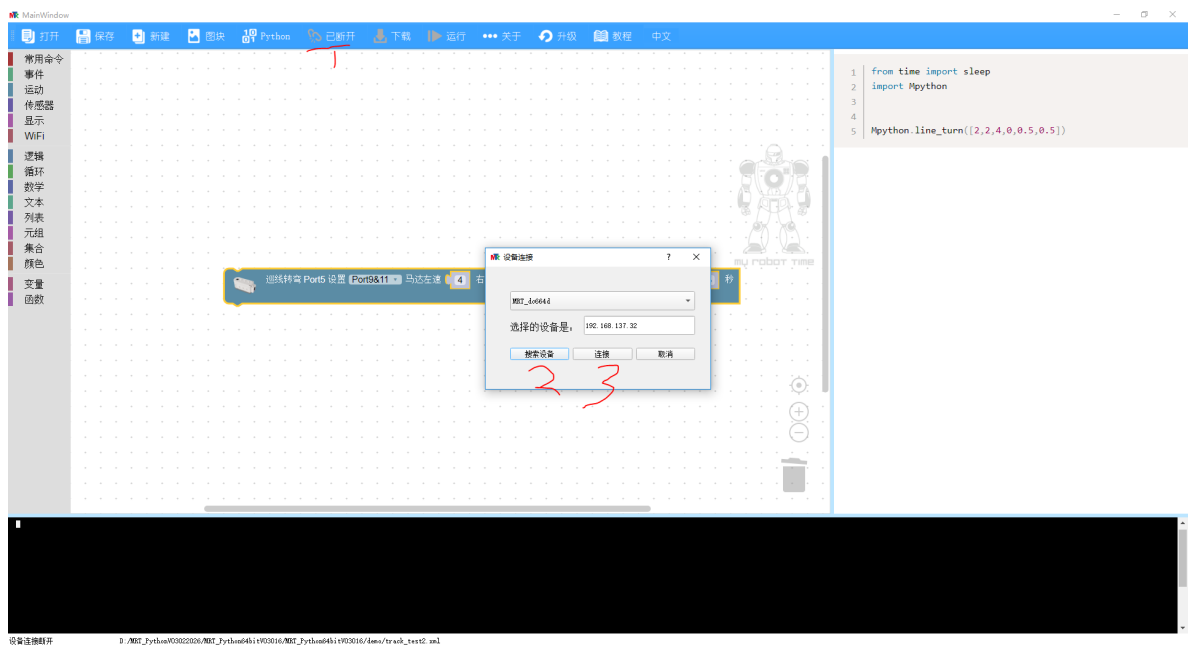
重启设备，按左键跳过WiFi设置提示界面，如果设备正确连接wifi 则 显示 **设备待连接状态**



如果不显示 或者跳回WiFi设置界面，则表示 设备没连接wifi

打开软件

打开电脑端MRT-PYTHON软件，点击**连接按钮**，并**搜索设备**



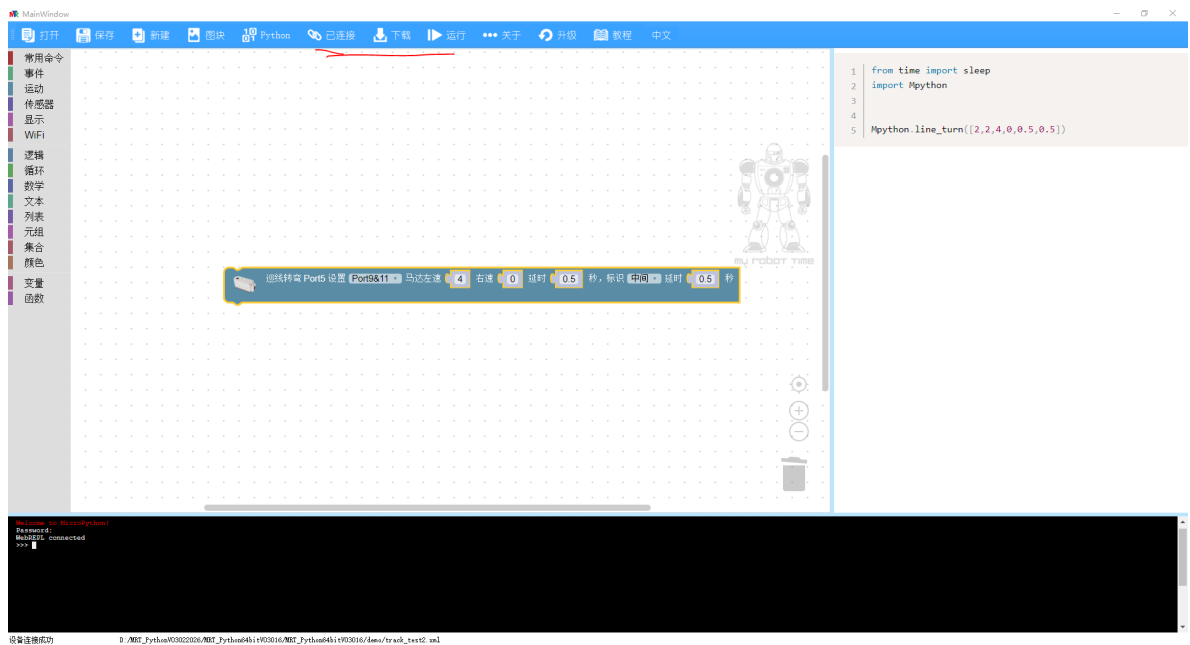
可以看到设备名称和设备IP地址，然后点击**连接**，即可连接到设备，进入设备界面编程模式

提示：1：如果搜索失败，可以重复几次，或者直接在IP地址栏 输入设备IP地址 来连接设备

2：还是提示失败，则确认一下，电脑PC和设备是否处在同一局域网网络内，可以利用电脑Ping 命令来测试一下电脑和设备是否相通

成功连接后

软件显示连接，状态栏显示设备连接，代码调试窗口有信息输出，设备按键区域 可用



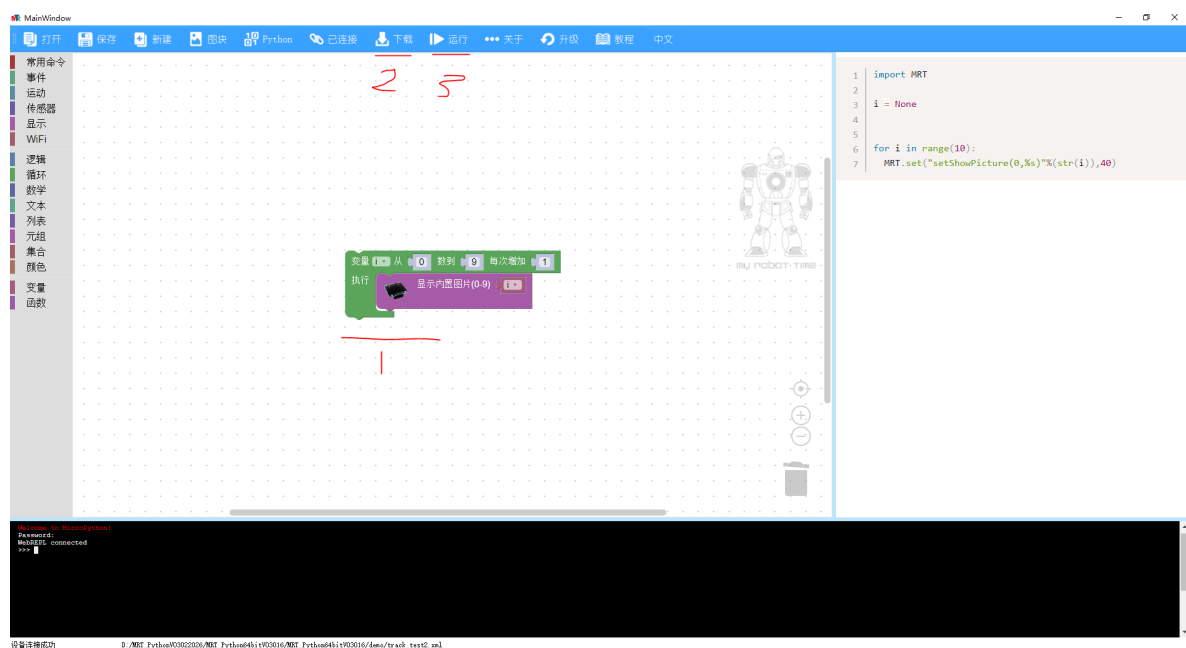
设备显示 Connect OK,即可进入对设备编程模式



提示：设备结束后 请断开连接，重新搜索设备 请重启设备 进入被待连接状态

BLOCKLY编程教学

简单编程 入门



1：编辑代码（或者打开 选择 demo 程序）

2：下载代码

3: 点击 运行 代码 即可在设备上看到运行效果

4: 随时可以在点击 **断开运行** 代码

提示: 务必先下载代码 后运行代码

OLED 编程

本产品包含一个高清OLED显示界面，可以显示内置图片、文字、数字等多种的应用。

显示内置图片例程

打开文件夹：例程\OLED测试例程\OLED.xml

例程功能：OLED界面来回切换设备内部保存的9张图片

Blockly 编程代码:



生成的Python代码是:

```
import MRT
from time import sleep

i = None

while True:
    for i in range(1, 10):
        MRT.set("setShowPicture(0,%s)"%(str(i)),40)
        sleep(1)
```

点击 软件 下载->运行

设备效果如下:



OLED显示字符例程

OLED可以利用行显示，坐标的方式控制字符的显示位置

打开文件夹：例程\OLED测试例程\OLED_文字.xml

例程功能：在第1行显示字符"MRT!",在坐标（0，50）处,来回显示1到10的数字。

BLOCKLY程序：



生成的Python代码是：

```
import MRT
from time import sleep

i = None

while True:
    for i in range(1, 11):
        MRT.set("setClear()", 64)
        MRT.set('setDisplay_row(%s,%s)'%(str(1), '\"'+ str("MRT!") + '\"'), 68)
        MRT.set('setDisplay_str(%s,%s,%s)'%( str(0), str(50), '\"'+ str(i) +
        '\"'), 28)
        sleep(1)
```

点击 软件 下载->运行

设备效果如下：

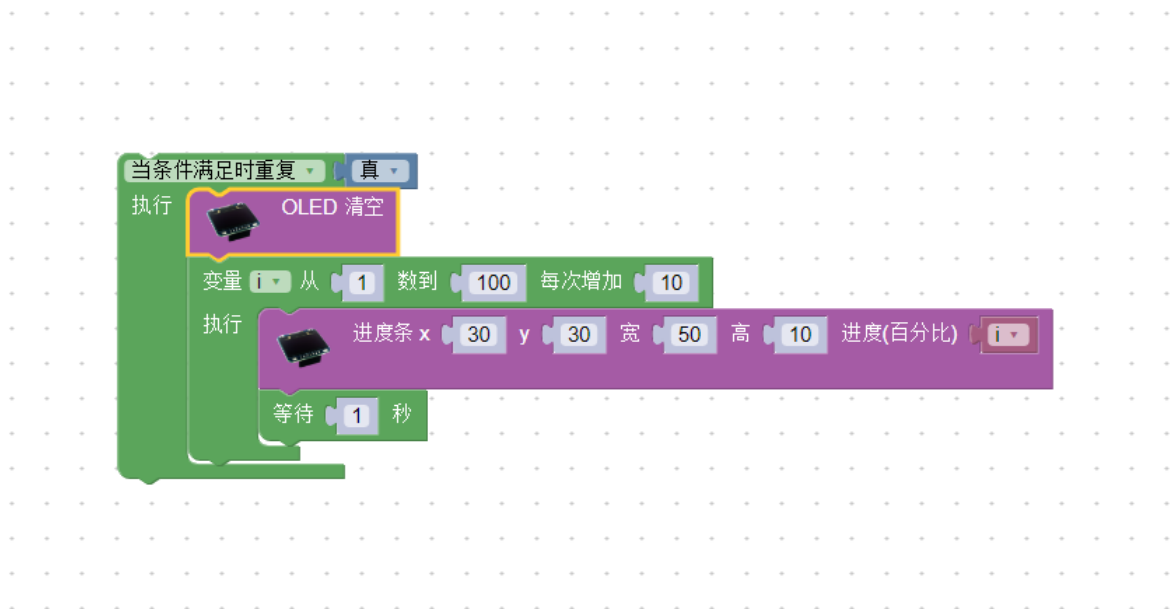


OLED显示进度条

打开文件夹：例程\OLED测试例程\OLED_进度条.xml

例程功能：进度条0-100%显示进度条，进度条的长度宽度也可以按需要设置

Blockly 编程：



Python 代码：

```

import MRT
from time import sleep

i = None

while True:
    MRT.set("setClear()",64)
    for i in range(1, 101, 10):
        MRT.set('setDrawbar(%s,%s,%s,%s,%s)'%
            (str(30),str(30),str(50),str(10),str(i)),70)
        sleep(1)

```

点击 软件下载->运行

设备效果如下：



例程\OLED测试例程 文件下，还有更多OLED编程例程，以供学习参考。

传感器编程

编程软件提供了19种各式传感器，可以充分利用传感器，搭建出非常多样的需求。

颜色传感器

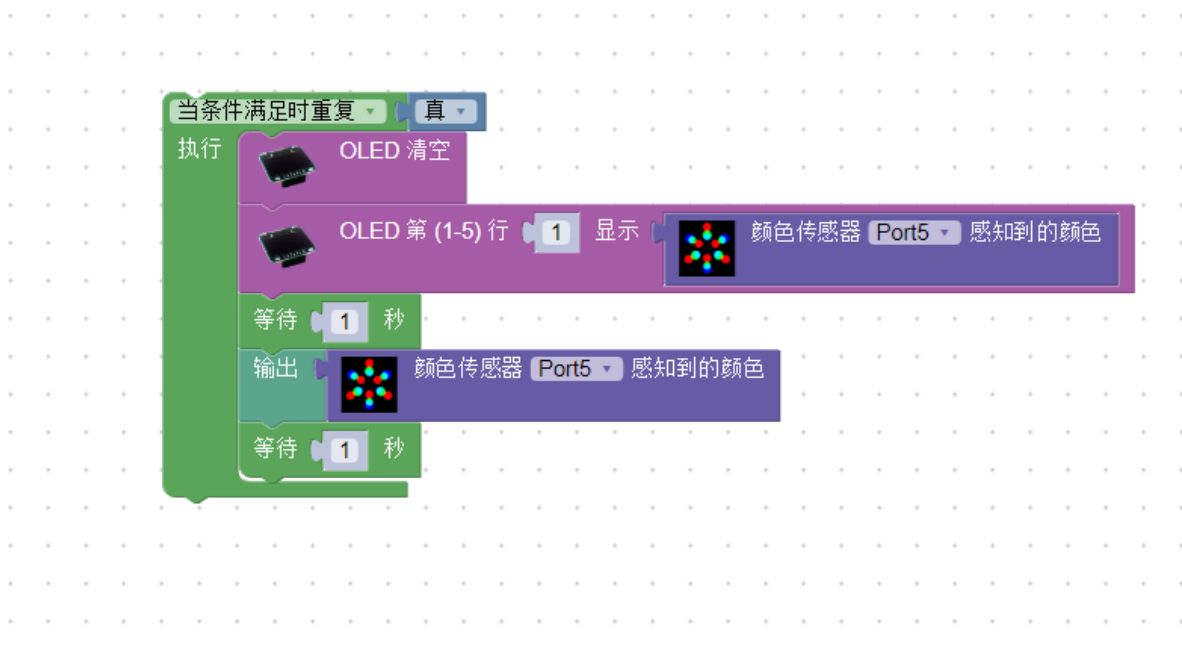
颜色传感器可以识别RGB中的颜色，并返回颜色数字表述。

返回值	颜色
3	blue/蓝色
2	Red/红色
1	Green/绿色
0	其他颜色

打开文件夹：例程\传感器测试例程\color.xml

例程功能：不断读取颜色传感器值，并在OLED上显示返回值，在调试窗口打印出传感器返回值。

Blockly 编程：



Python 例程：

```
import MRT
from time import sleep

while True:
    MRT.set("setClear()",64)
    MRT.set('setDisplay_row(%s,%s)'%(str(1) ,'\''+
str(MRT.value("getColorRecognition(1,4)\n",22 ,4)) + '\'',68)
    sleep(1)
    print(MRT.value("getColorRecognition(1,4)\n",22 ,4))
    sleep(1)
```

设备连接：

将颜色传感器连接到主设备的5端口，点击软件 下载->运行，将传感器对准颜色卡。



RGB/LED控制

RGB/LED 灯可以控制点亮，并显示不同的RGB颜色。RGB颜色可以通过值输入或者通过输入16进制颜色值控制。

打开文件夹：例程\传感器测试例程\RGB.xml

例程功能：不断变化颜色，并在OLED 显示颜色的16进制RGB数值

Blockly 编程：


```
MRT.value("setRGBValue(2,1,%s,%s,%s)"%(int('#33cc00'[1:3],  
16),int('#33cc00'[3:5], 16),int('#33cc00'[5:7], 16)),24 ,1)  
MRT.set('setDisplay_row(%s,%s)'%(str(1) , '\"'+ str('#33cc00') + '\"'),68)  
sleep(1)  
MRT.value("setRGBValue(2,1,255,255,255)\n",24 ,1)  
sleep(1)
```

设备连接

将RGB灯连接到主板1口，点击连接下载->运行



环境温湿度传感器

环境温湿度传感器可以采集传感器所处环境的 温度和湿度。

打开文件夹：例程\OLED测试例程\humiture.xml

例程功能：在OLED 上显示传感器返回的温度和湿度

Blockly编程:



Python代码：

```
import MRT
from time import sleep

while True:
    MRT.set("setClear()",64)
    MRT.set('setDisplay_row(%s,%s)'%(str(1) ,'\''+
str(MRT.value("getAirTemperature(1,4)\n",4 ,4)) + '\'',68)
    MRT.set('setDisplay_row(%s,%s)'%(str(2) ,'\''+
str(MRT.value("getAirHumidity(1,4)\n",4 ,4)) + '\'',68)
    sleep(1)
```

设备连接：

将温湿度传感器连接到主板5端口,点击软件 下载->运行 。

设备运行效果：



更多传感器例程参考

在文件：例程\传感器测试例程\下包含了所有成功的使用例程。

注意事项：务必正确连接主板，传感器连接主板端口需要与程序所设置的端口一值

电机编程

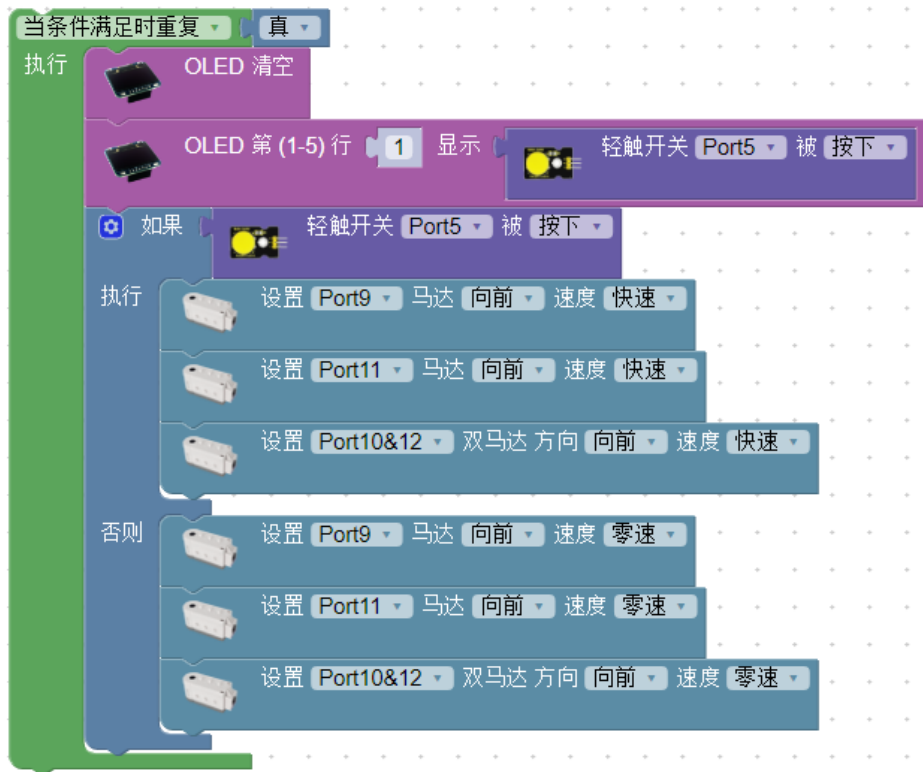
电机运行例程

本设备可以同时控制4个电机，每个电机都可以单独控制，也可以一组一组控制，端口9和11是一组，端口10和12是是一组。

打开：例程\电机测试例程\Button_Driver.xml

例程功能：当按下按键时4个电机同时转动，松开按键时4个电机停止，OLED显示按键是否按下，按下 True 松开False

Blockly编程：



Python 编程：

```

import MRT

while True:
    MRT.set("setClear()",64)
    MRT.set('setDisplay_row(%s,%s)'%(str(1) ,'\'+
str(MRT.value("getTouch(1,4)\n",2 ,4) == 1) + '\'),68)
    if MRT.value("getTouch(1,4)\n",2 ,4) == 1:
        MRT.set("setDcMotor(0,1,1,3)\n",18)
        MRT.set("setDcMotor(0,2,1,3)\n",18)
        MRT.set("setDoubleDcMotor(0,2,1,3)\n",36)
    else:
        MRT.set("setDcMotor(0,1,1,0)\n",18)
        MRT.set("setDcMotor(0,2,1,0)\n",18)
        MRT.set("setDoubleDcMotor(0,2,1,0)\n",36)
  
```

连接设备：

按键	port5
电机1	port9
电机2	port10
电机3	port11
电机4	port12

点击软件下载->运行



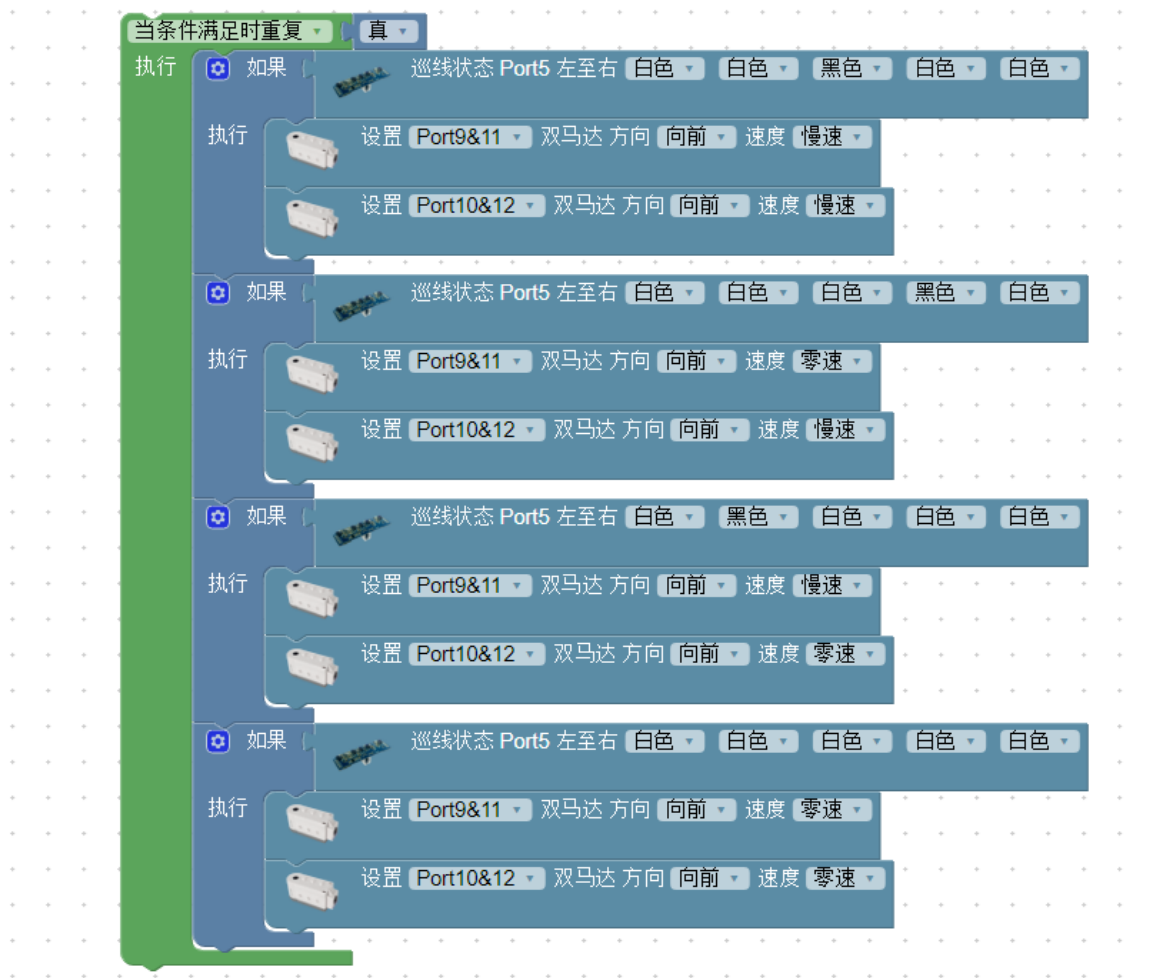
巡线例程

巡线功能需要借助巡线传感器实现，巡线传感器有5对红外对管，每个红外对管能够识别白色和黑色。

打开：例程\电机测试例程\巡线测试.xml

例程功能：根据巡线模块的不同感应模式，调整不同电机的转动

Blockly编程：



Python代码：

```
import Mpython
import MRT
```

```

while True:
    if Mpython.line_track([1,1,0,1,1]):
        MRT.set("setDoubleDcMotor(0,1,1,1)\n",36)
        MRT.set("setDoubleDcMotor(0,2,1,1)\n",36)
    if Mpython.line_track([1,1,1,0,1]):
        MRT.set("setDoubleDcMotor(0,1,1,0)\n",36)
        MRT.set("setDoubleDcMotor(0,2,1,1)\n",36)
    if Mpython.line_track([1,0,1,1,1]):
        MRT.set("setDoubleDcMotor(0,1,1,1)\n",36)
        MRT.set("setDoubleDcMotor(0,2,1,0)\n",36)
    if Mpython.line_track([1,1,1,1,1]):
        MRT.set("setDoubleDcMotor(0,1,1,0)\n",36)
        MRT.set("setDoubleDcMotor(0,2,1,0)\n",36)

```

连接设备：

巡线模块	port5
电机1	port9
电机2	port10
电机3	port11
电机4	port12

点击软件下载->运行



模拟黑线 电机会有不同反应，可以搭建巡线机器人模型来更好的应用功能。

WiFi 设置编程

打开：例程\WiFi测试例程\WiFi例程.xml

例程功能：设置WiFi例程,并在OLED上显示WiFi连接信息,调试窗口输出读取的WiFi配置信息

Blockly 编程例程：



Python代码:

```
import MRT
import network

wlan = network.WLAN(network.STA_IF)
wlan.active(True)

MRT.set("setClear()",64)
MRT.set('setDisplay_row(%s,%s)'%(str(1) ,'\n'+ str(wlan.isconnected()) +
'\n'),68)
MRT.set('setDisplay_row(%s,%s)'%(str(2) ,'\n'+ str(wlan.ifconfig()[0]) +
'\n'),68)
print(wlan.ifconfig())
```

设备显示:

点击软件 下载->运行



OLED 界面显示：True 已经连接WIFI，并显示本机设备IP地址