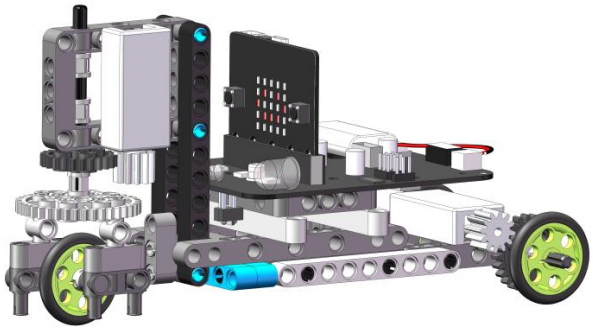
**Lesson1 of Building:bit Independent steer car---“Infrared remote control”**

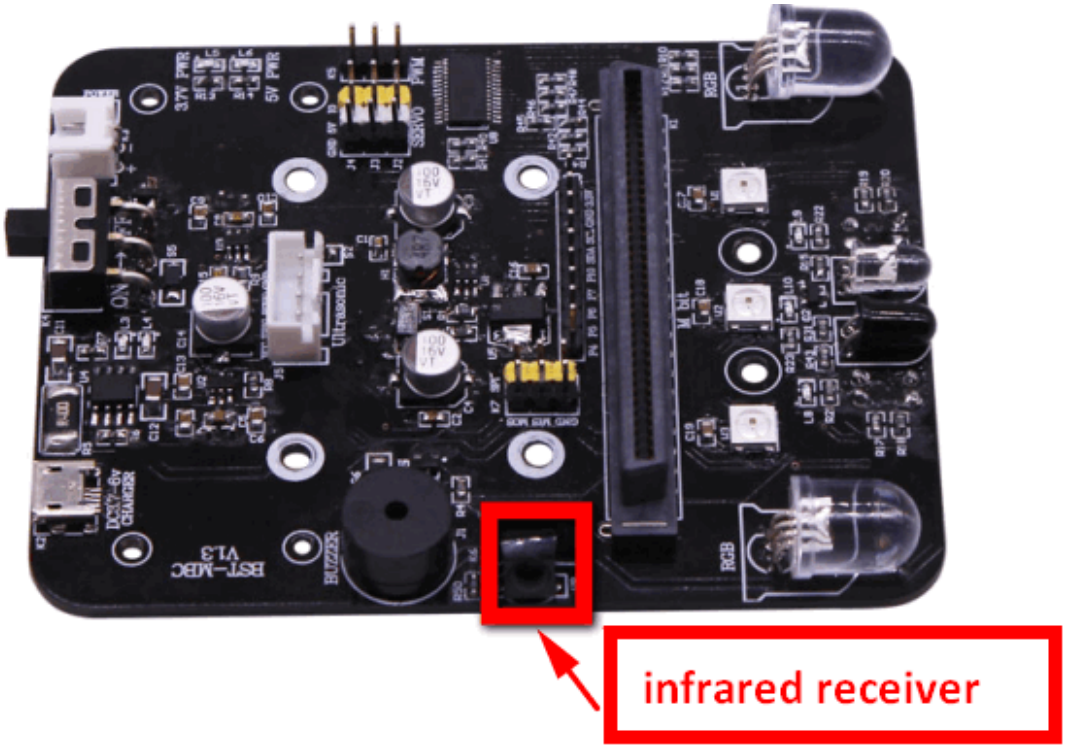


**Note：**

**1. When performing infrared remote control, the remote controller should face the infrared receiver on the expansion board.**

**2. There is a plastic piece on the bottom of the infrared remote controller that needs to be taken down for normal use.**

**3. The infrared light emitted by the infrared remote controller and the infrared receiver is invisible to the human eye. It can be seen under the camera without filtering infrared light.**




**1.Experimental phenomena**

After downloading the program, open the power switch of the Independent steer car and press the button of the infrared remote controller. The Independent steer car will have corresponding action. On the Infrared remote controller, small light button, add button, subtract button,rotate left button, rotate right button to control the color of the colorful lights, the red power button

to turn off light, 1~7 button represents the music do, re, mi, fa, sol, la, si. The front, back, left, right buttons control advance, back, turn left,turn right of the Independent steer car. 0 and 8, 9 are used to control the image of the dot matrix screen.

## 2.Preparation before class

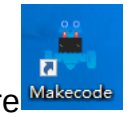
We needs to be ready:
Building Block Independent steer car *1
Infrared remote controller *1
USB data cable *1

### 2-1.Two programming methods:

Online programming:

First,we need to connect the micro:bit to the computer by USB data cable, the computer will pop up a USB flash drive.Then, click on the URL in the USB flash drive: http://microbit.org/ to enter the edit process interface, click to 【Extensions】, and copy the package URL: https://github.com/lzty634158/yahboom_mbit_en and https://github.com/lzty634158/YB_IR to the input field, and you can use the building blocks of the Yahboom software package.

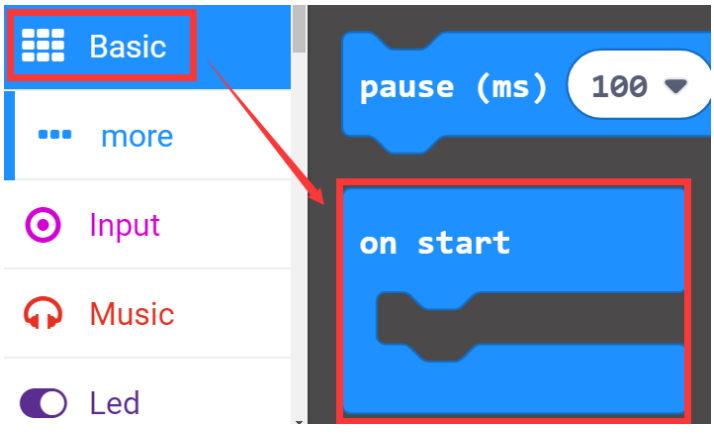
Offilne programming:

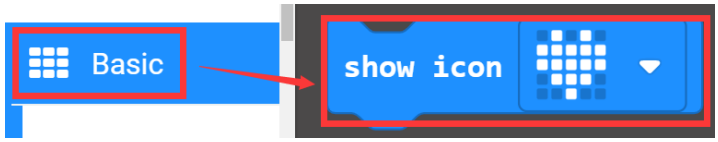
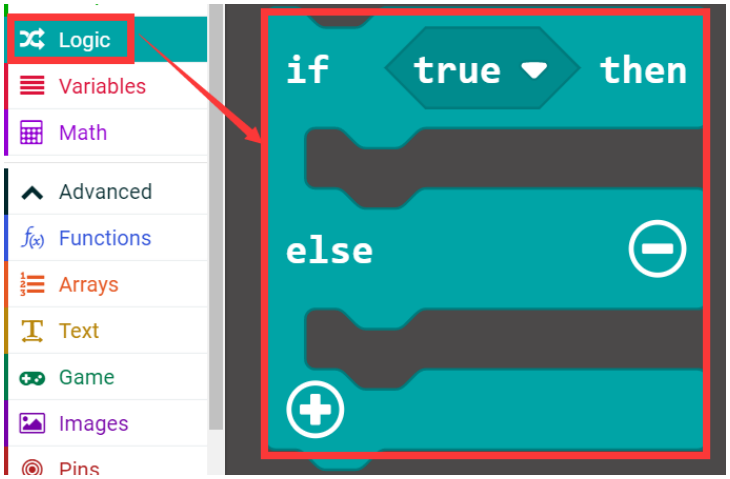
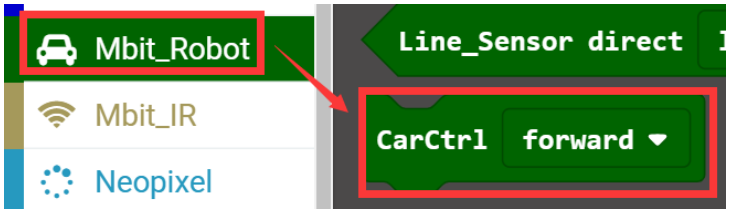
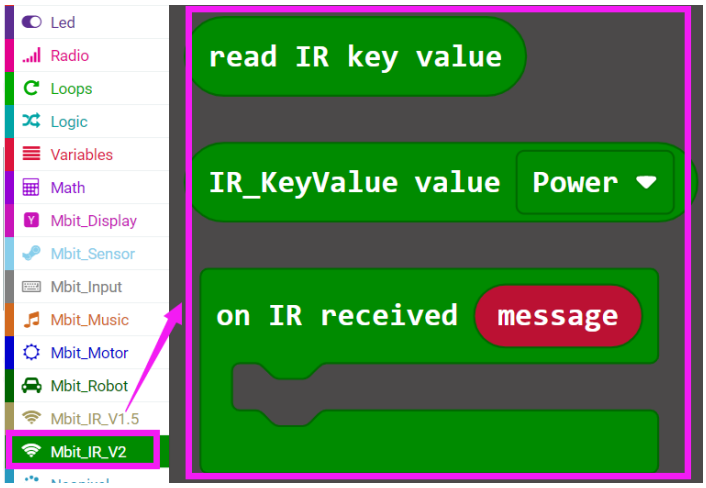
Open the offline programming software 

, click to 【Extension】 and copy the package URL: https://github.com/lzty634158/yahboom_mbit_en and https://github.com/lzty634158/YB_IR to the input field, and you can use the building blocks of the Yahboom software package.

For detailed programming, please read the documentation before class 【1. Preparation before class】----【Introduction of programming method】. We use micro:bit official website for online programming in here.

## 3.Studying blocks

| Blocks | Instruction |
| --- | --- |
|  | Executed at boot time, the code is only executed once. |

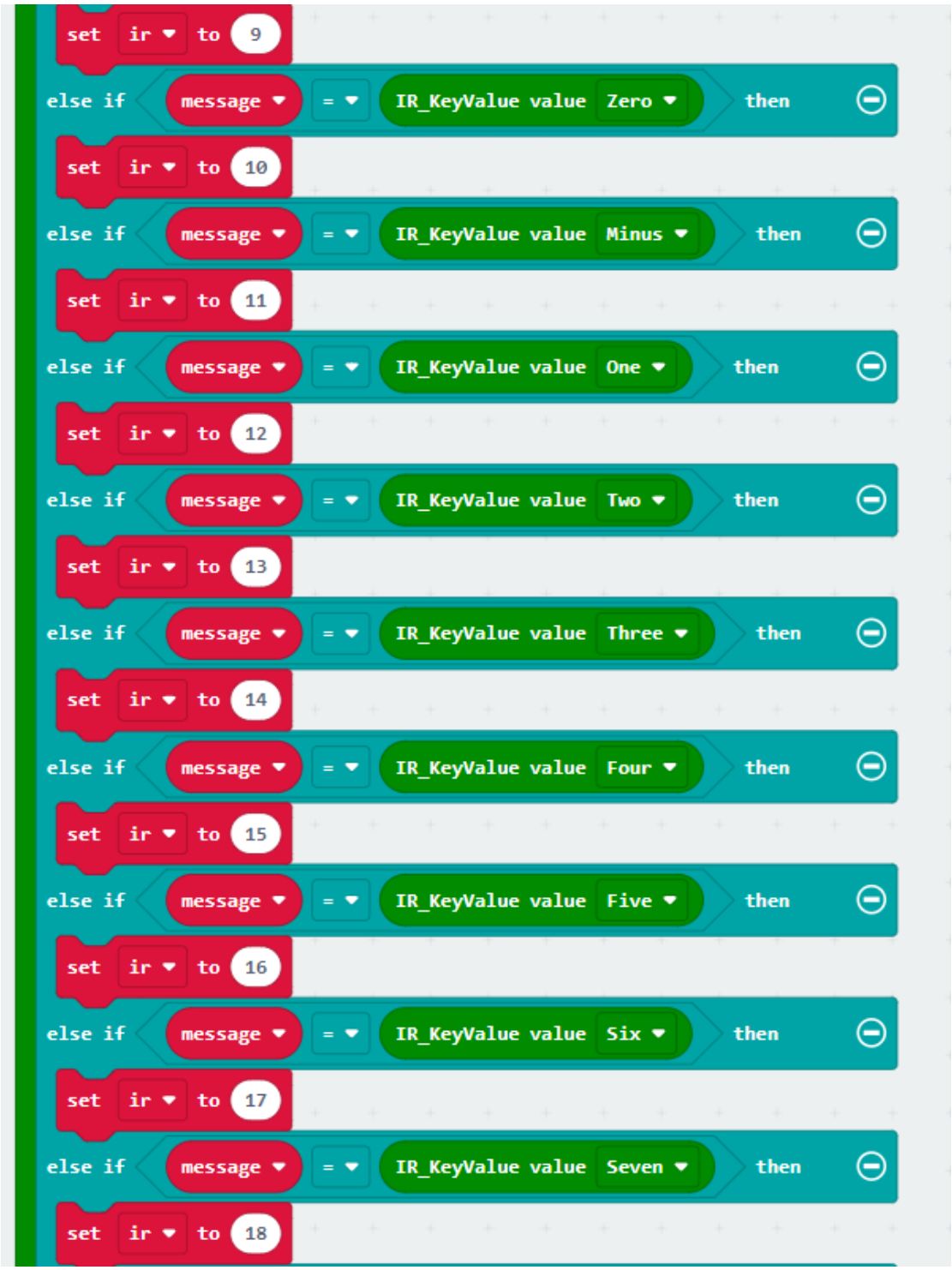
| | |
|---|---|
|  | Display image on the lattice of micro:bit. |
|  | If true then execute. If it is false, it will not be executed. |
|  | The program pauses for 100 milliseconds and the time can be modified by yourself. |
|  | The car control forward represents the motor rotates in the positive direction and the car control back represents and the motor rotates in the opposite direction. |
|  | When the power button on the remote controller is pressed, the code inside will be executed, and the button can be customized.<br>Set the infrared remote control receiving pin. In this experiment, the receiving pin is P8, so you must select P8, otherwise you will not receive the signal. |

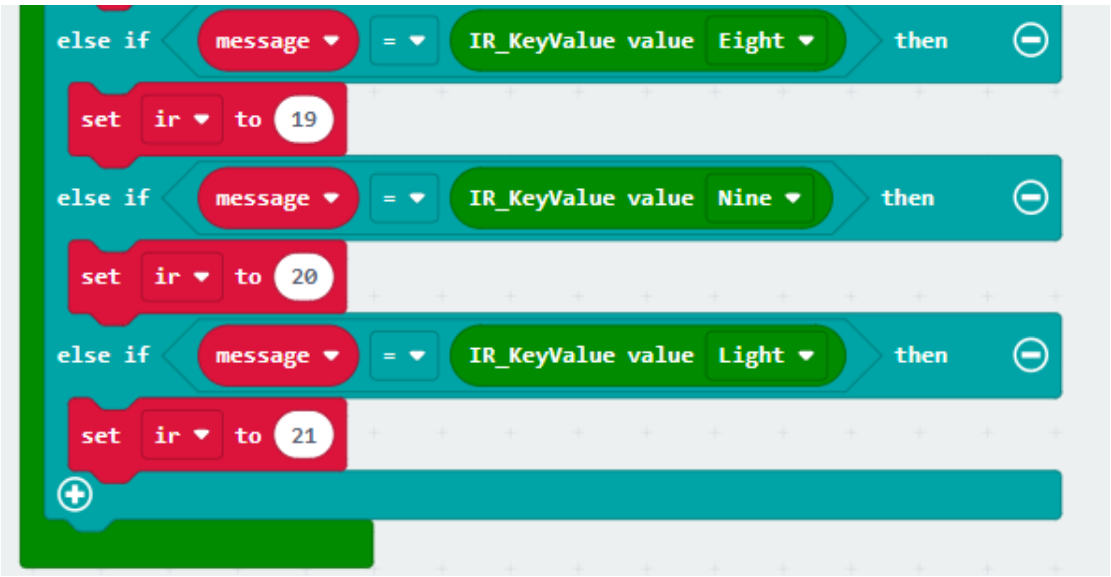
| | |
|---|---|
|  | Select the color of the lights. |
|  | Can play different tones. |

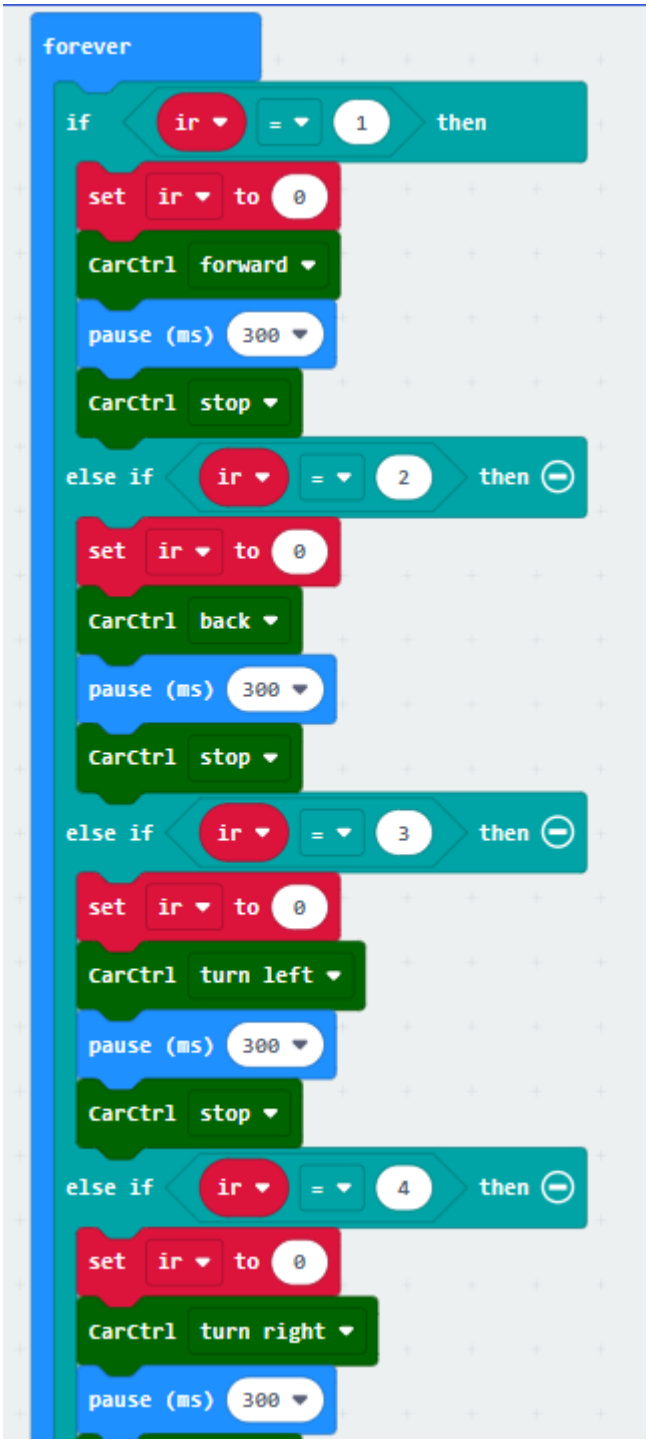
**4.Programming**

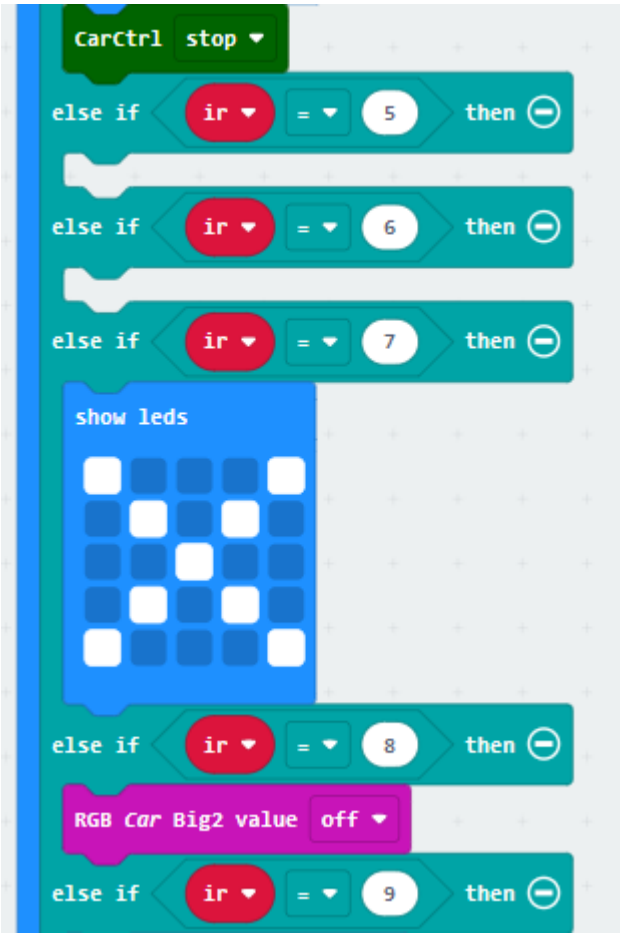
Next, we started to write the program for the infrared remote control of the building block Independent steer car, as shown below:

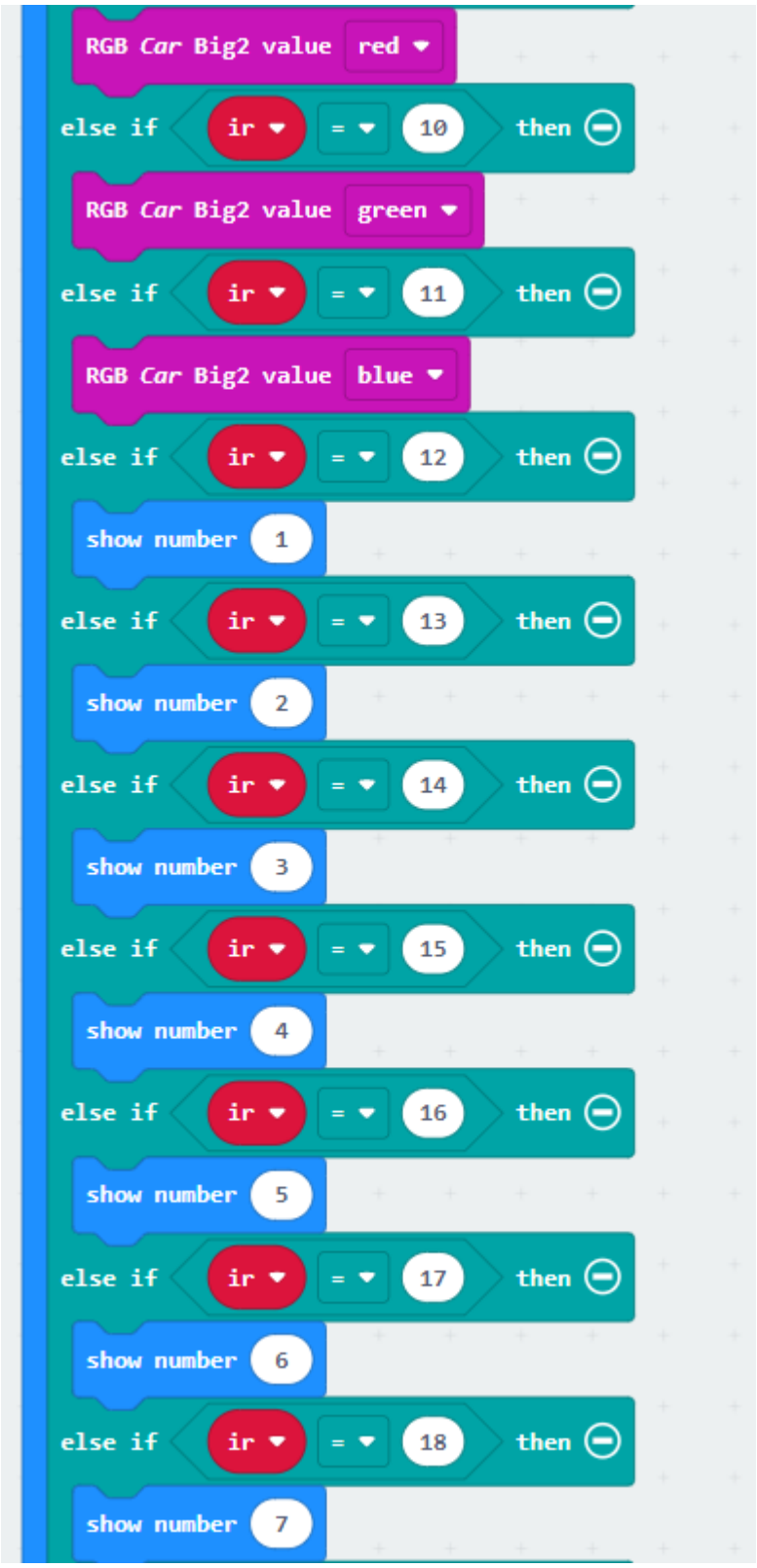

on IR received message
set ir to message
if message = IR_KeyValue value Up then
set ir to 1
else if message = IR_KeyValue value Down then
set ir to 2
else if message = IR_KeyValue value Left then
set ir to 3
else if message = IR_KeyValue value Right then
set ir to 4
else if message = IR_KeyValue value SpinLeft then
set ir to 5
else if message = IR_KeyValue value SpinRight then
set ir to 6
else if message = IR_KeyValue value Beep then
set ir to 7
else if message = IR_KeyValue value Power then
set ir to 8
else if message = IR_KeyValue value Add then

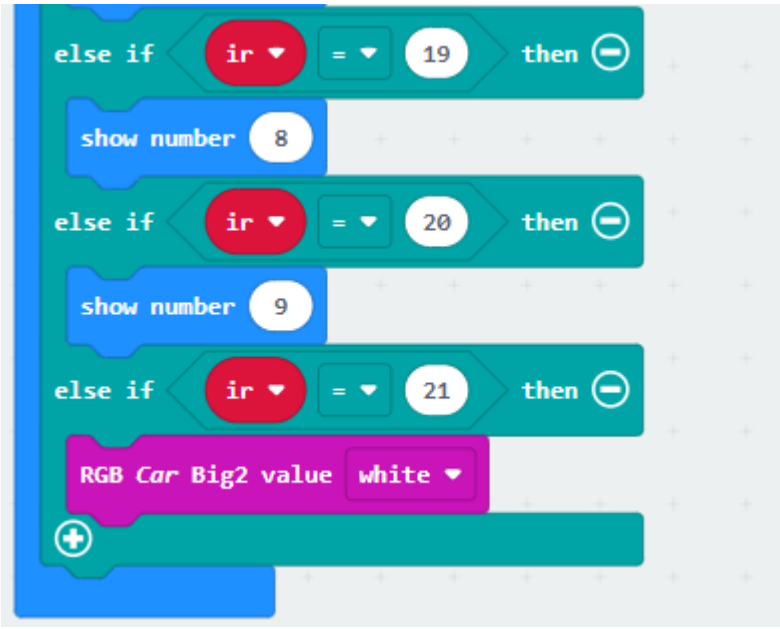

set ir to 9
else if message = IR_KeyValue value Zero then
set ir to 10
else if message = IR_KeyValue value Minus then
set ir to 11
else if message = IR_KeyValue value One then
set ir to 12
else if message = IR_KeyValue value Two then
set ir to 13
else if message = IR_KeyValue value Three then
set ir to 14
else if message = IR_KeyValue value Four then
set ir to 15
else if message = IR_KeyValue value Five then
set ir to 16
else if message = IR_KeyValue value Six then
set ir to 17
else if message = IR_KeyValue value Seven then
set ir to 18


else if message = IR_KeyValue value Eight then
set ir to 19
else if message = IR_KeyValue value Nine then
set ir to 20
else if message = IR_KeyValue value Light then
set ir to 21


forever
if ir = 1 then
set ir to 0
CarCtrl forward
pause (ms) 300
CarCtrl stop
else if ir = 2 then
set ir to 0
CarCtrl back
pause (ms) 300
CarCtrl stop
else if ir = 3 then
set ir to 0
CarCtrl turn left
pause (ms) 300
CarCtrl stop
else if ir = 4 then
set ir to 0
CarCtrl turn right
pause (ms) 300


CarCtrl stop
else if ir = 5 then
else if ir = 6 then
else if ir = 7 then
show leds
else if ir = 8 then
RGB Car Big2 value off
else if ir = 9 then


RGB Car Big2 value red
else if ir = 10 then
RGB Car Big2 value green
else if ir = 11 then
RGB Car Big2 value blue
else if ir = 12 then
show number 1
else if ir = 13 then
show number 2
else if ir = 14 then
show number 3
else if ir = 15 then
show number 4
else if ir = 16 then
show number 5
else if ir = 17 then
show number 6
else if ir = 18 then
show number 7




The above is the program for this Independent steer car. After writing, we need to download it to the micro:bit board.