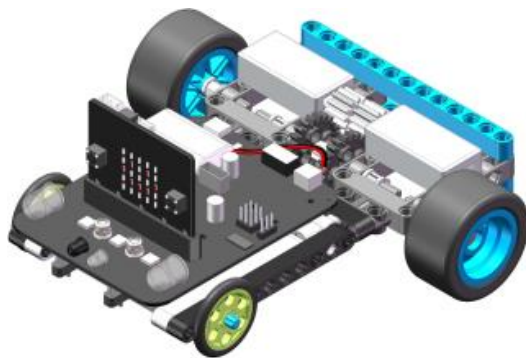
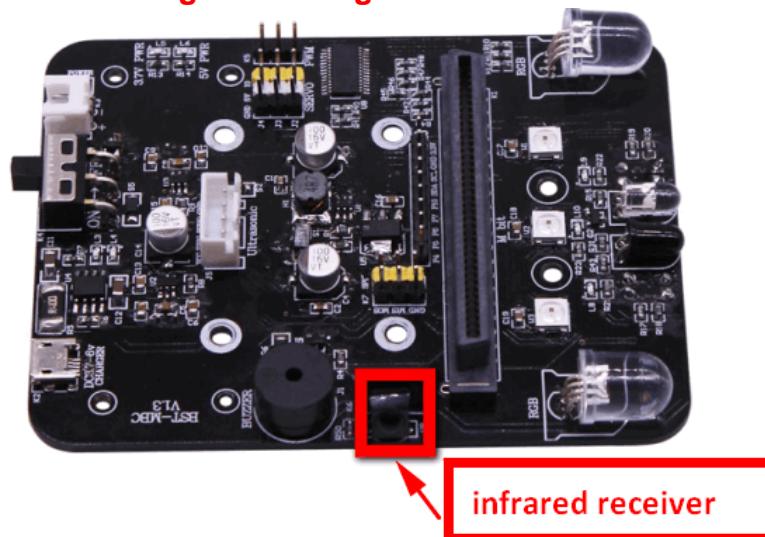


Lesson3 of Building:bit Following car---“Infrared remote control”

**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 Following car and press the button of the infrared remote controller. The Following car will have corresponding action. On the Infrared remote controller, small light button, add button, subtract 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, rotate left, rotate right buttons control the motion state of 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 Following car *1

micro:bit handle *1

USB data cable *1

2-1.Two programming methods:

Online programming:

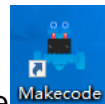
First,we need to connect the micro:bit to the computer by USB data cable, the computer will pop up a USB flash drive.Then, click on the URL in the USB flash drive: <http://microbit.org/> to enter the edit process interface, click to

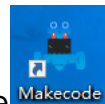
【Extensions】 , and copy the package URL:

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:

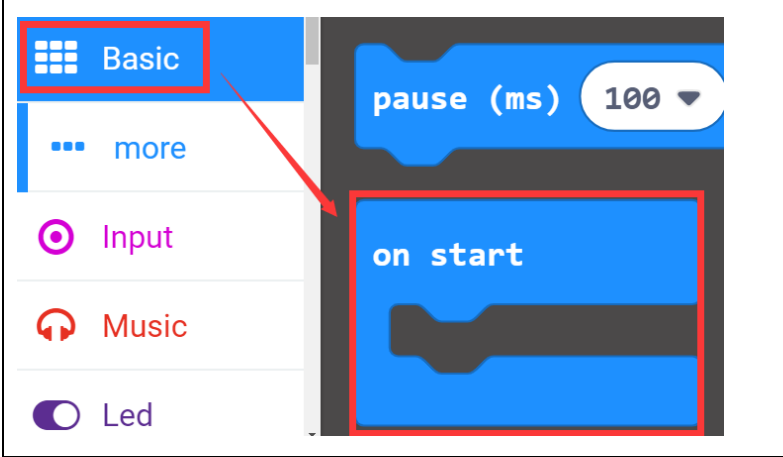
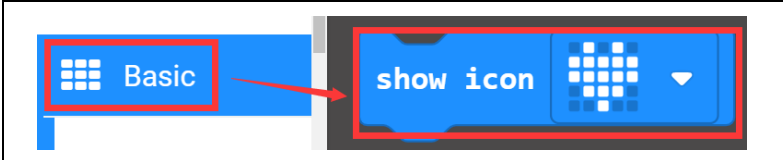


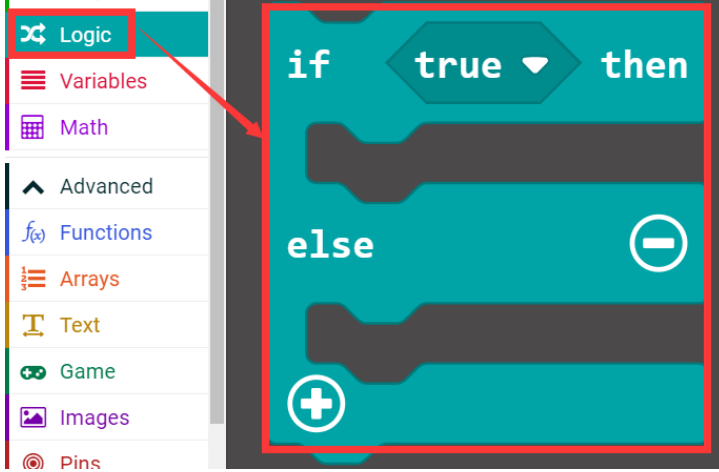

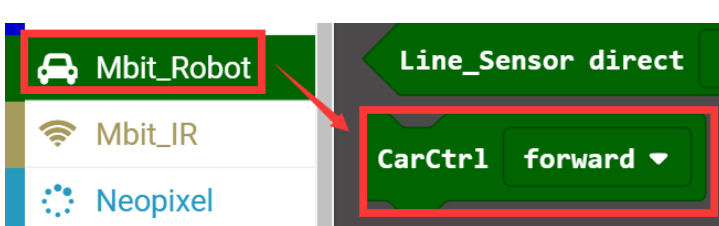
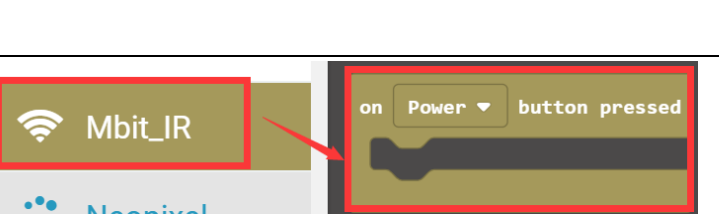
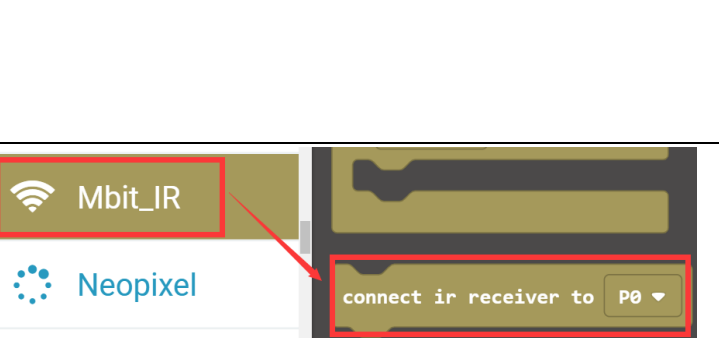
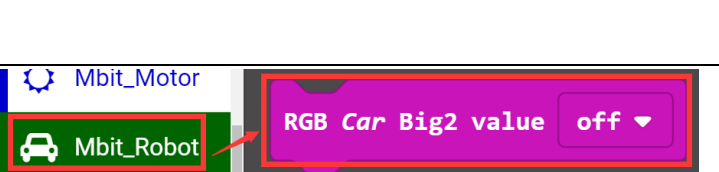
Open the offline programming software , click to 【Extension】 and copy the package URL: https://github.com/lzty634158/yahboom_mbit_en and https://github.com/lzty634158/YB_IR to the input field, and you can use the building blocks of the Yahboom software package.

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

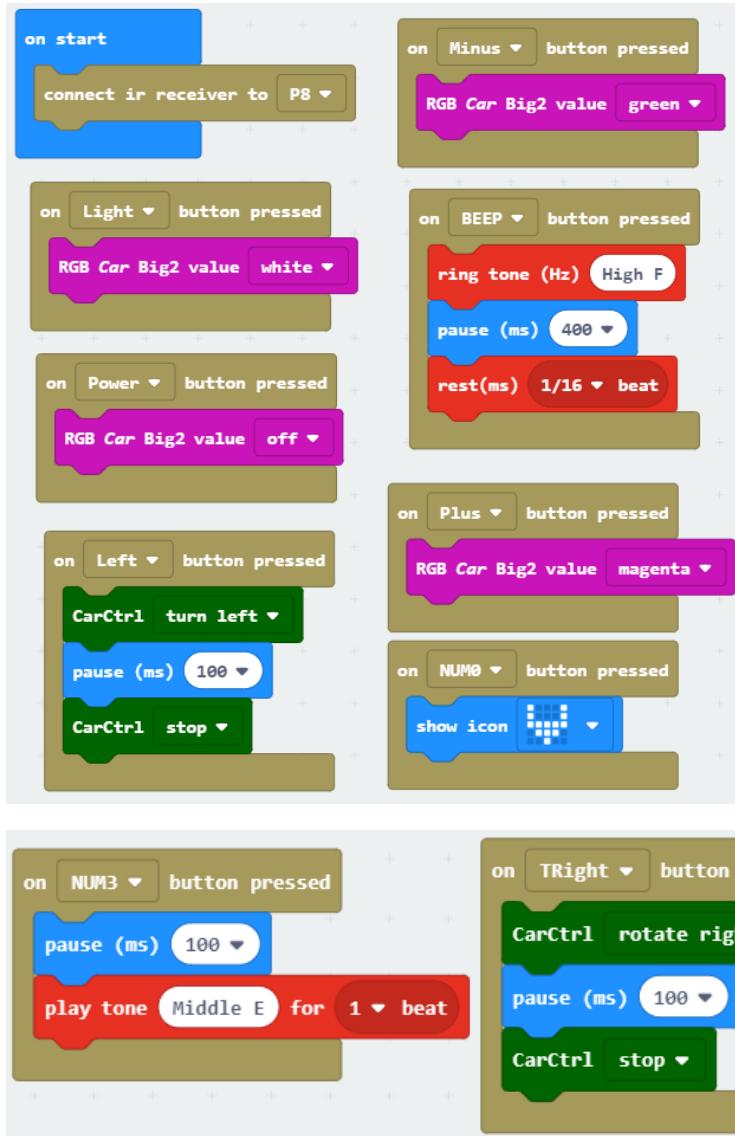
Blocks	Instruction
	<p>Executed at boot time, the code is only executed once.</p>
	<p>Display image on the lattice of micro:bit.</p>

	<p>If it is true then execute. If it is false, it will not be executed.</p>
	<p>The program pauses for 100 milliseconds and the time can be modified by yourself.</p>
	<p>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.</p>
	<p>When the power button on the remote controller is pressed, the code inside will be executed, and the button can be customized.</p>
	<p>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.</p>
	<p>Select the color of the lights.</p>

	Can play different tones.
---	---------------------------

4.Programming

Next, we start write infrared remote control of Following car, as shown below:



```

on start
  connect ir receiver to P8

on Light button pressed
  RGB Car Big2 value white

on Power button pressed
  RGB Car Big2 value off

on Left button pressed
  CarCtrl turn left
  pause (ms) 100
  CarCtrl stop

on Left button pressed
  CarCtrl turn left
  pause (ms) 100
  CarCtrl stop

on Minus button pressed
  RGB Car Big2 value green

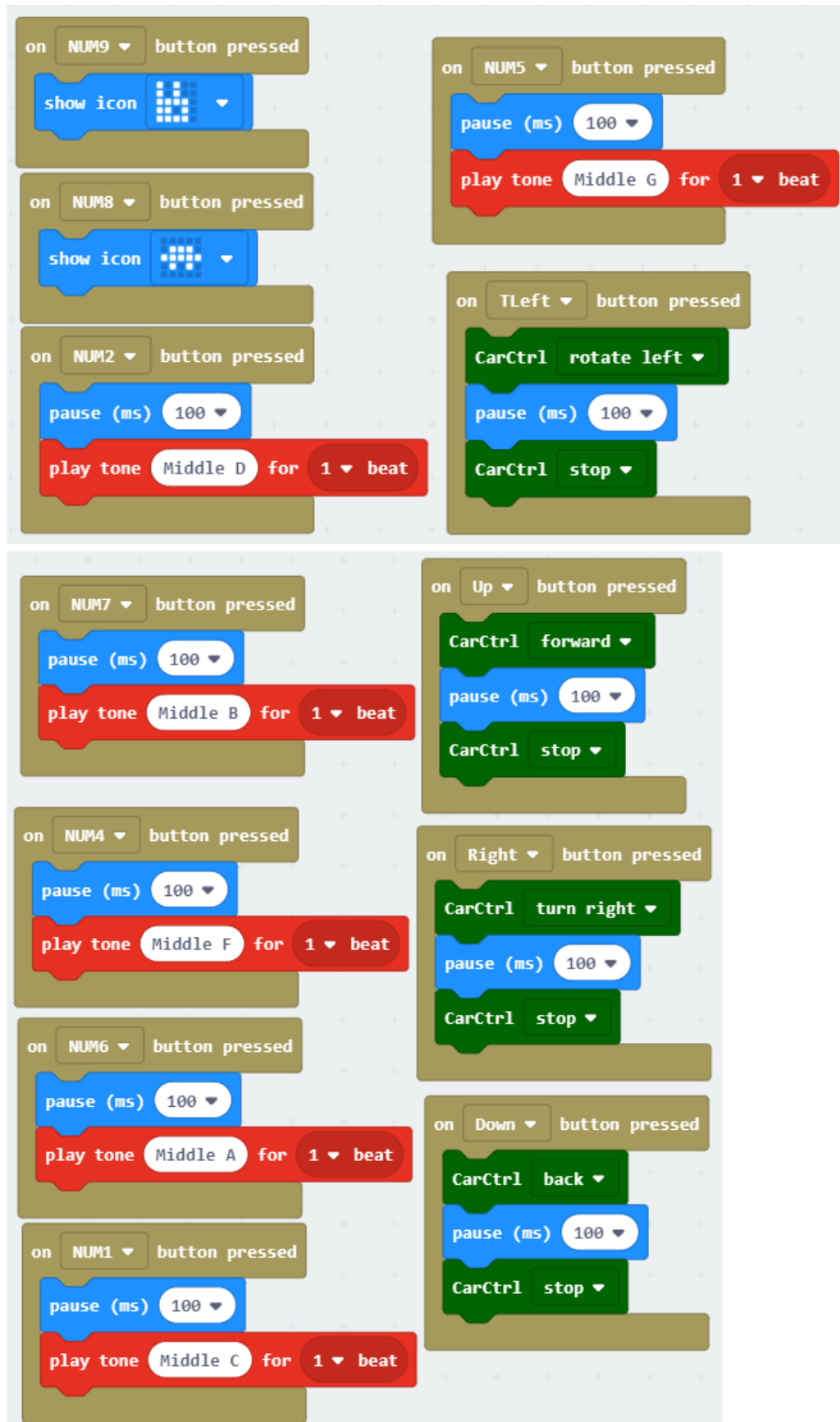
on BEEP button pressed
  ring tone (Hz) High F
  pause (ms) 400
  rest(ms) 1/16 beat

on Plus button pressed
  RGB Car Big2 value magenta

on NUM0 button pressed
  show icon

on NUM3 button pressed
  pause (ms) 100
  play tone Middle E for 1 beat

on TRight button pressed
  CarCtrl rotate right
  pause (ms) 100
  CarCtrl stop
  
```



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