WiFi camera control

> **The WiFi camera module in this case needs to be purchased separately**

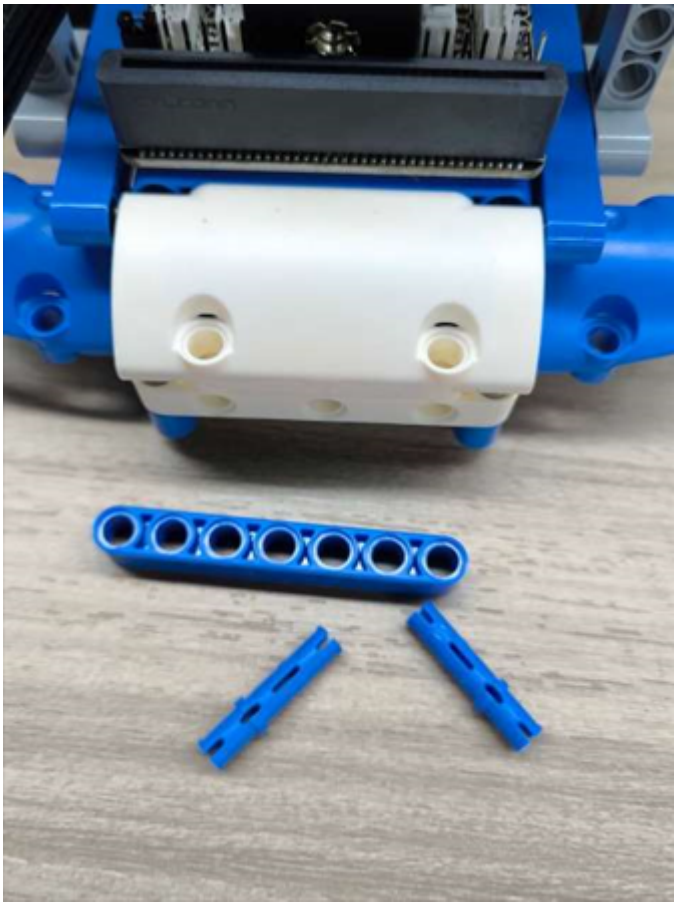
1. Learning objectives

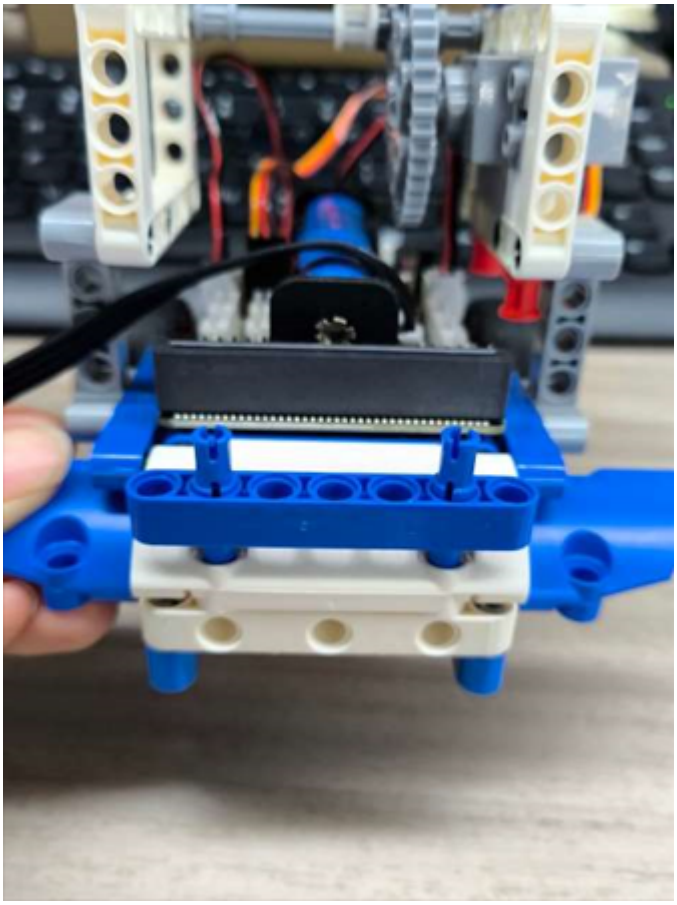
In this course, we mainly learn how to use MakeCode graphical programming to achieve WiFi remote control of Pretty car.

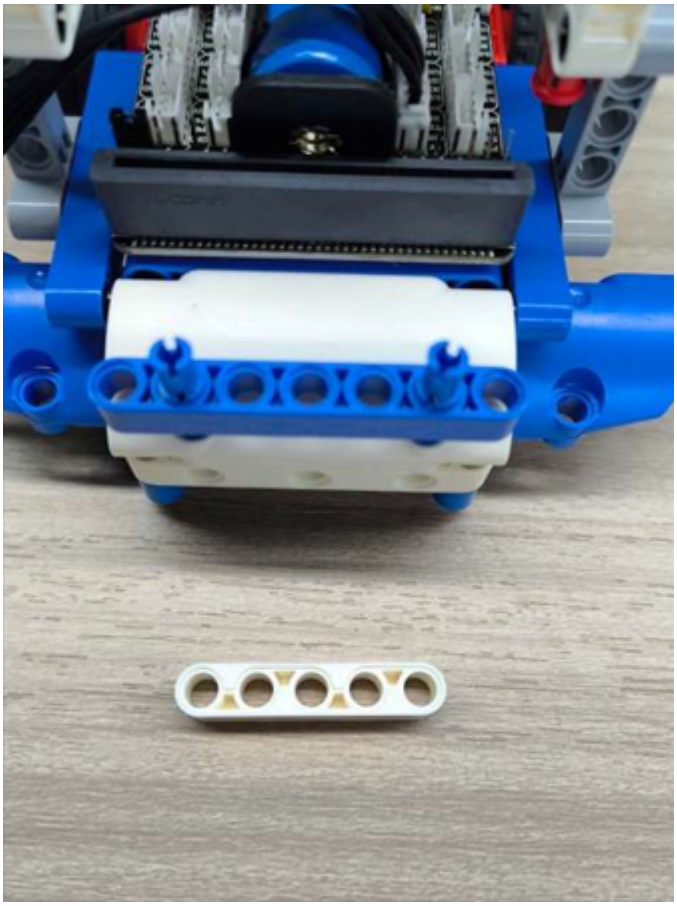
2. Building blocks

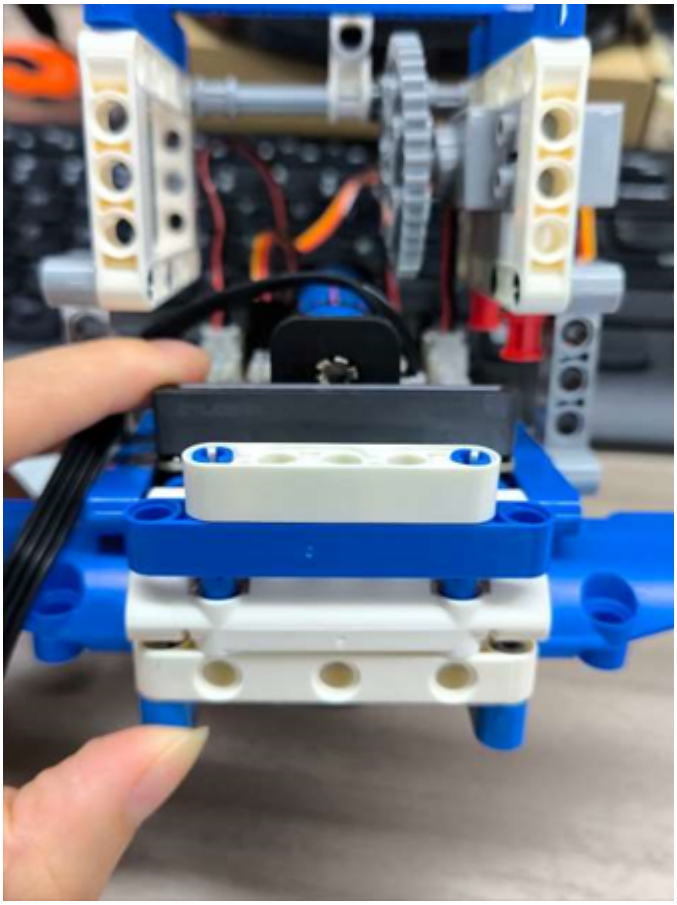
For detailed steps of building blocks, please refer to the installation drawings of **[Assembly Course]--[Pretty car]** in the materials or the building block installation album.

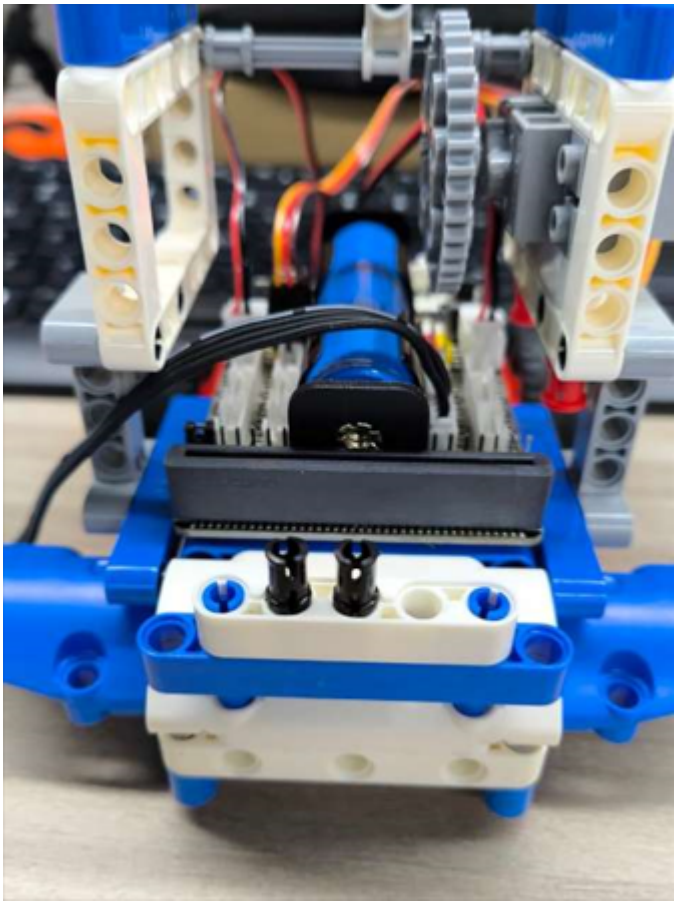
Then follow the steps below to install the WiFi camera module:











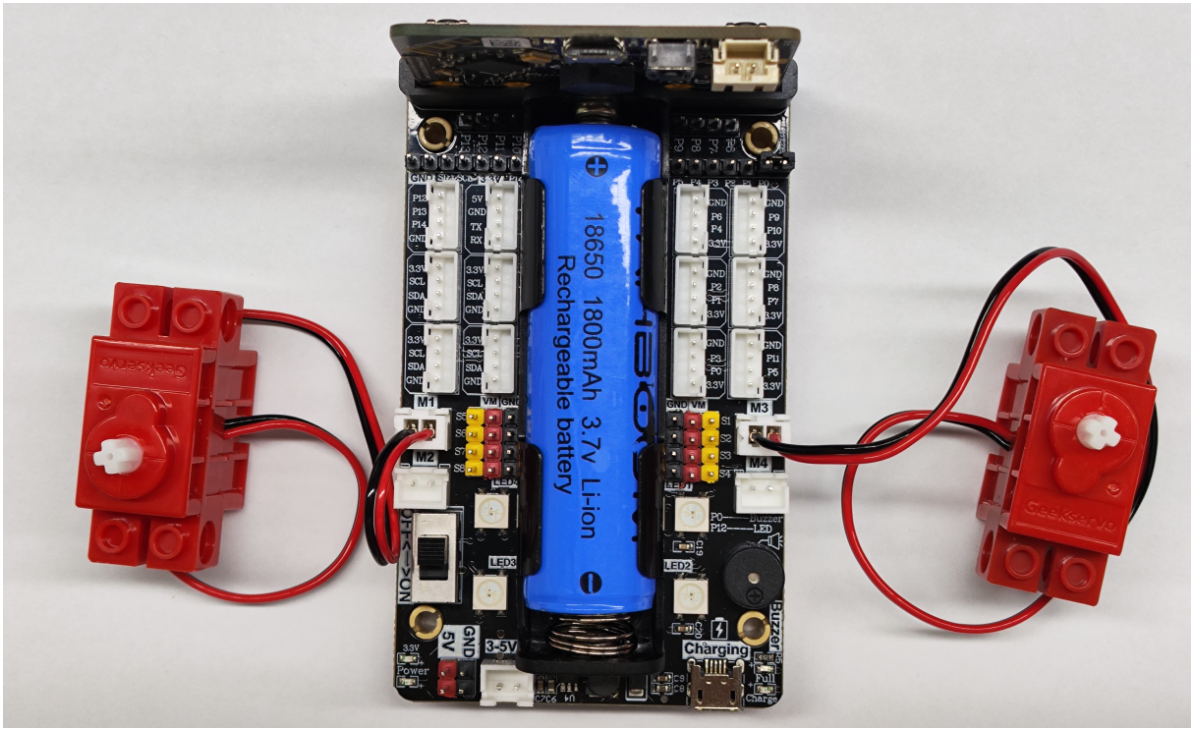
3. Wiring

Motor wiring

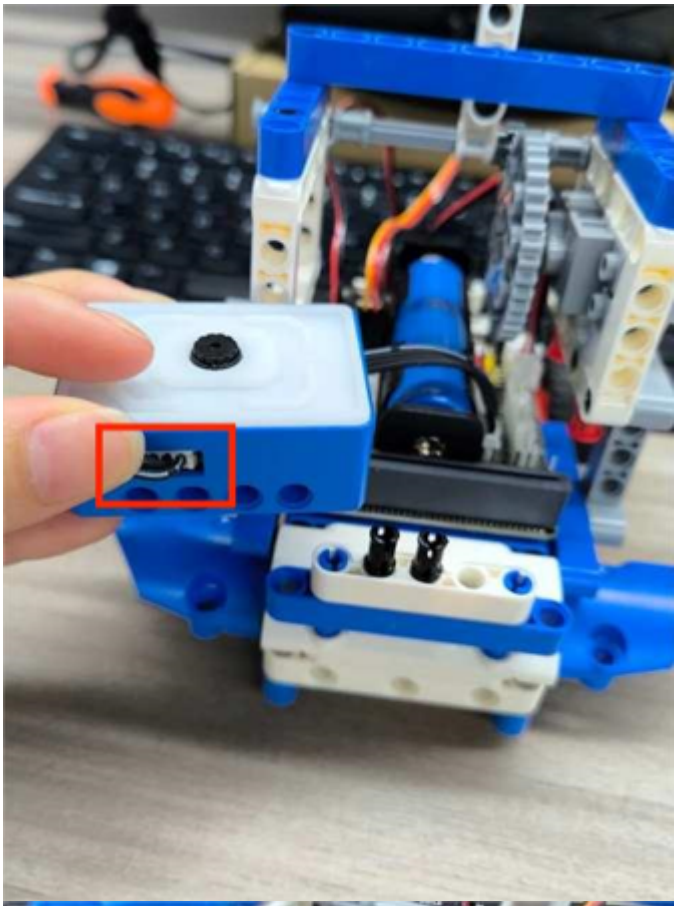
The motor wiring on the left side of the car is inserted into the M1 interface of the Super:bit expansion board, and the black line is close to the battery side;

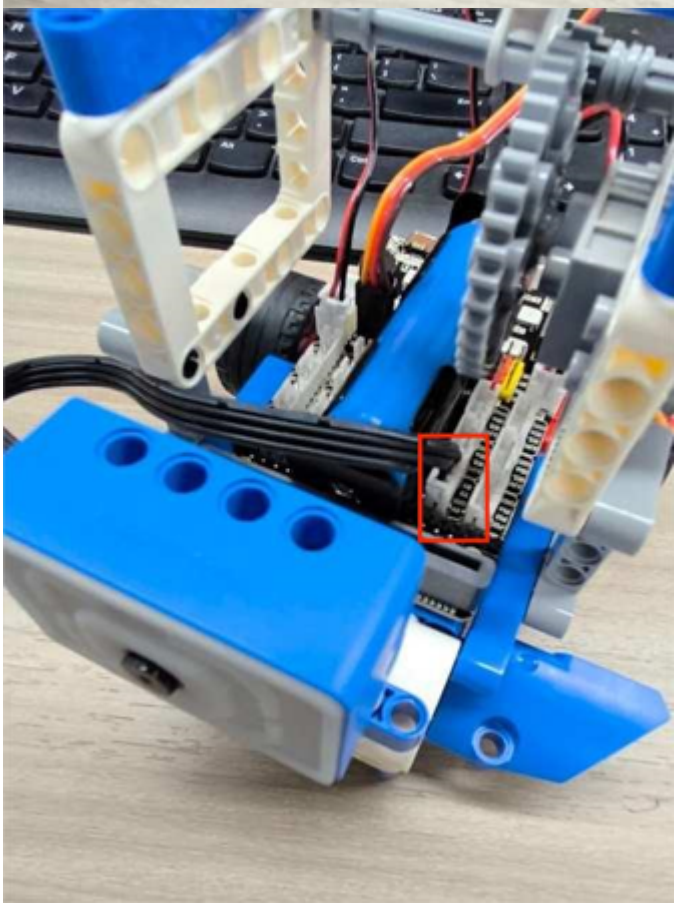
The motor wiring on the right side of the car is inserted into the M3 interface of the Super:bit expansion board, and the black line is close to the battery side;

As shown below:



Wiring between camera and car





4. Programming

Method 1 Online programming:

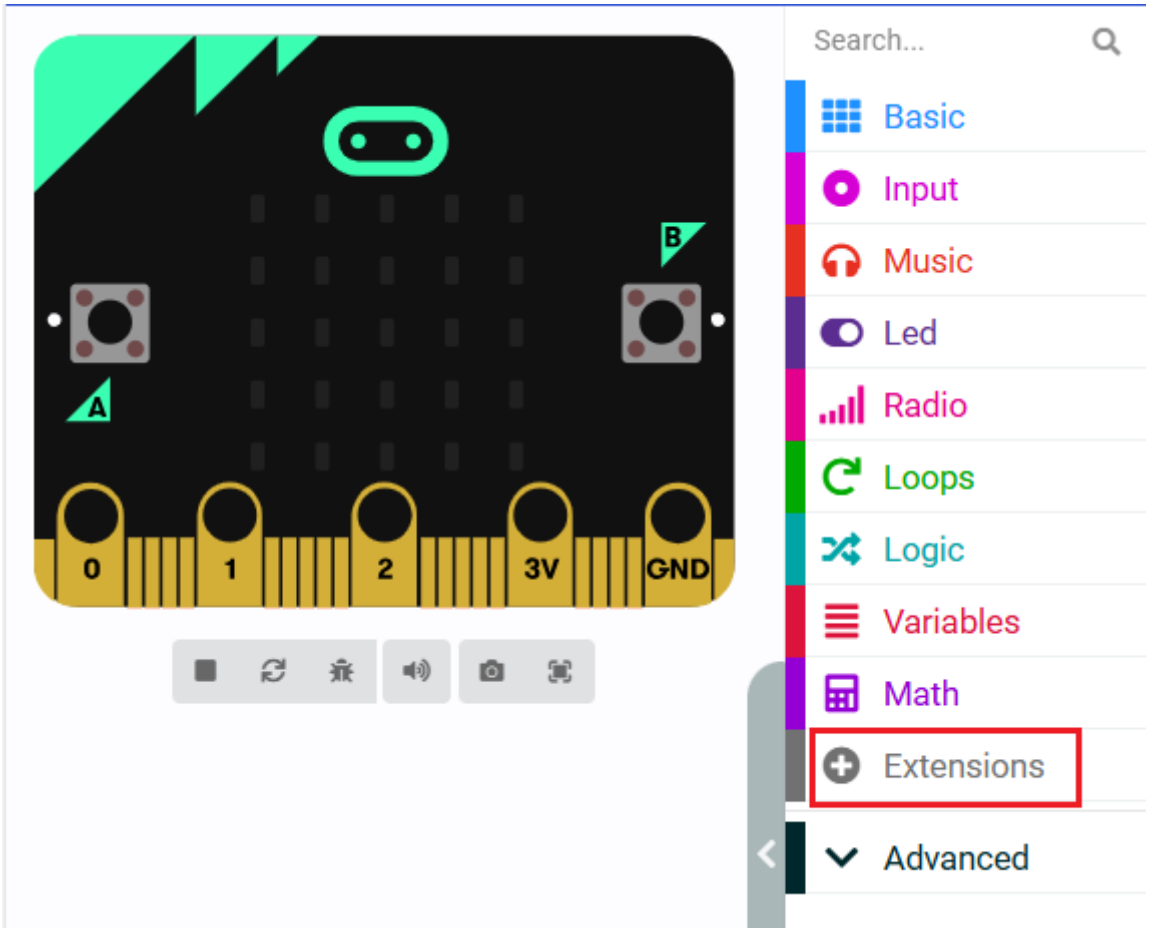
First, connect micro:bit to the computer via a USB data cable. A U disk will pop up on the computer. Click the URL in the U disk: https://makecode.microbit.org/ to enter the programming interface. Then, add the Yabo smart software package to start programming.

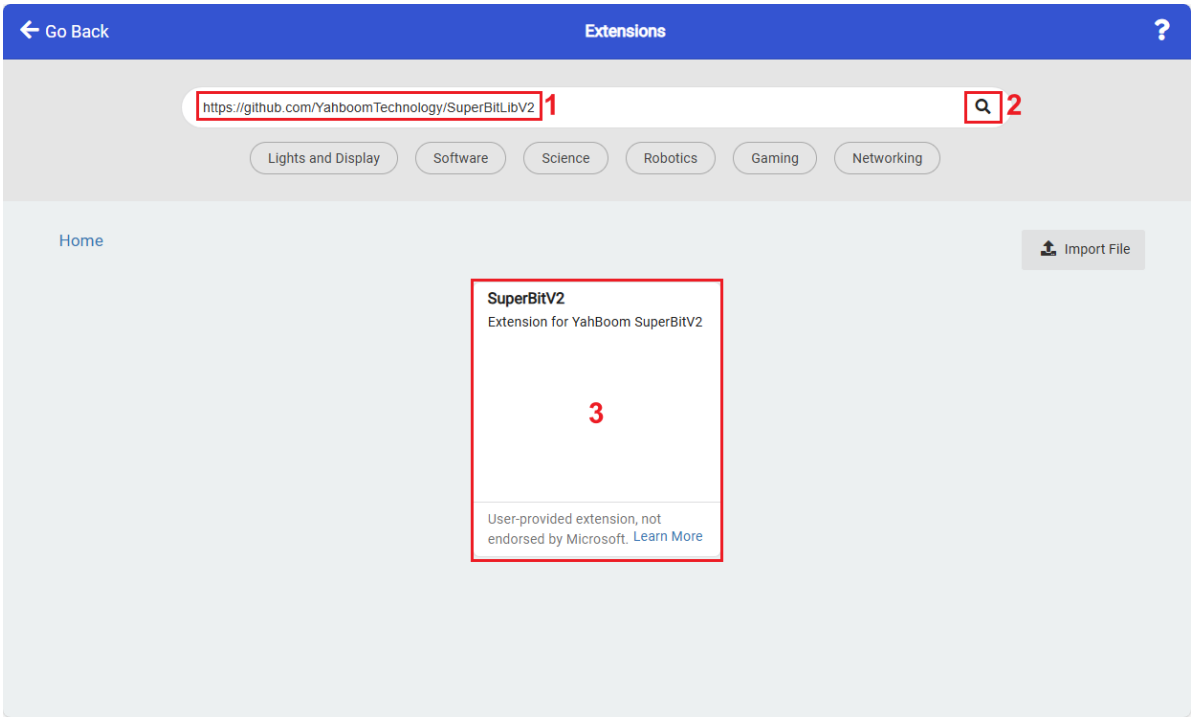
Method 2 Offline programming:

Open the offline programming software MakeCode and enter the programming interface. Click [New] and add the Yabo smart software package to start programming.

superbit kit expansion package: https://github.com/YahboomTechnology/SuperBitLibV2
WiFi camera expansion package: https://github.com/yahboomtechnology/ESP32-wifi-Microbit

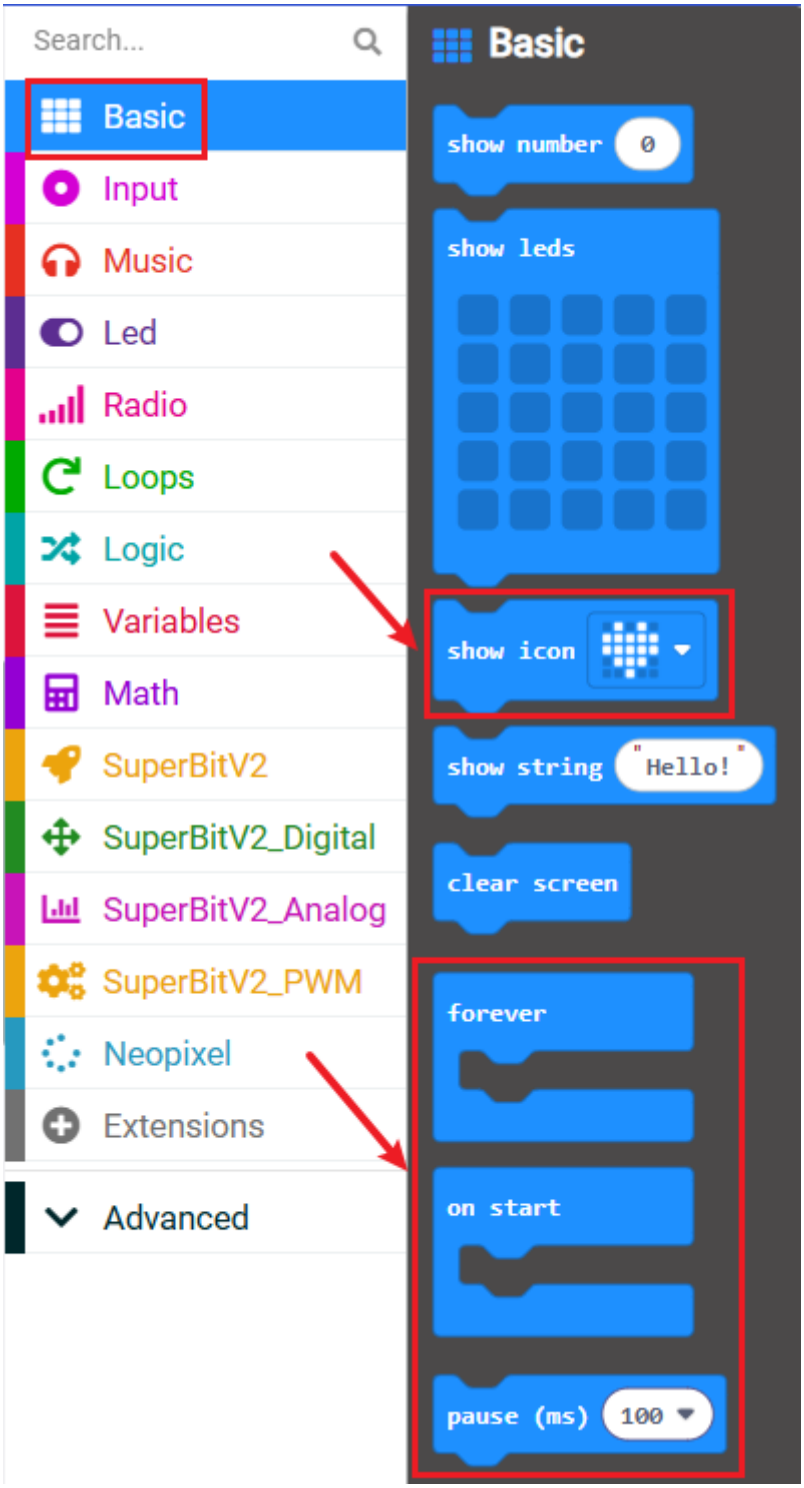
4.1 Add expansion package

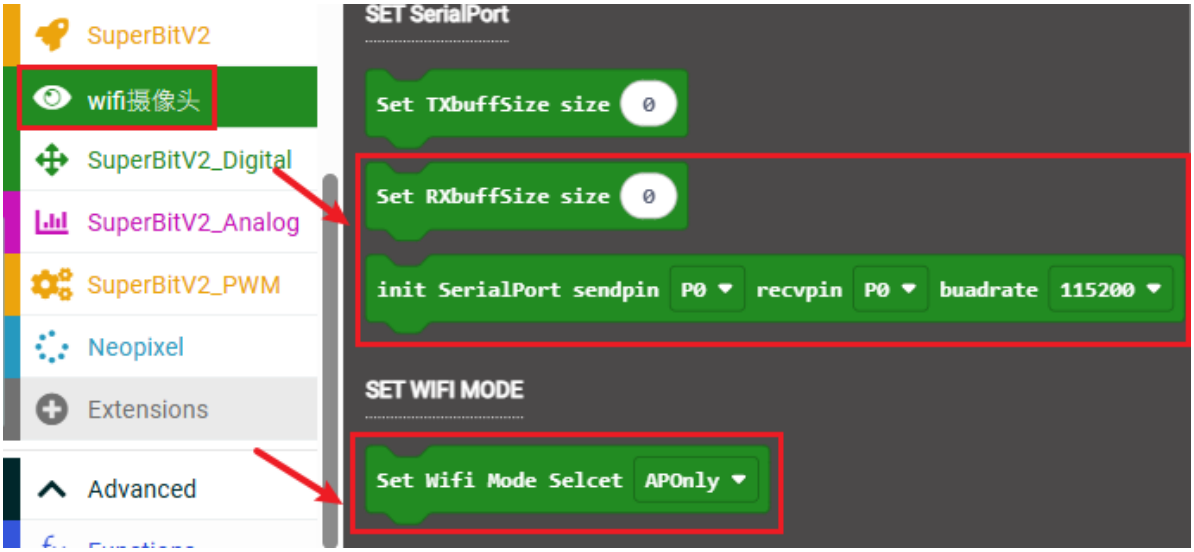


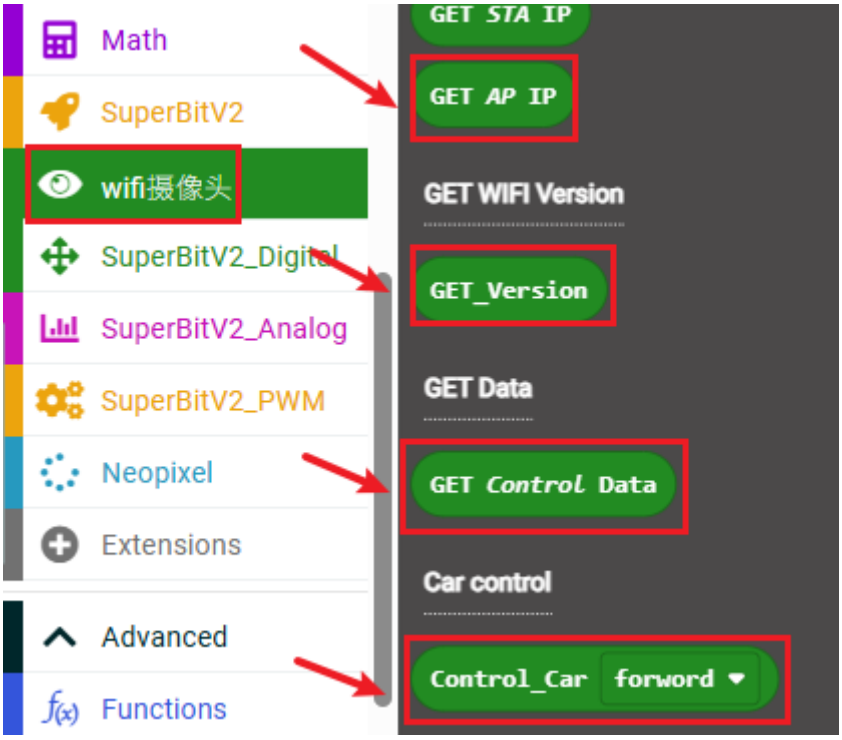


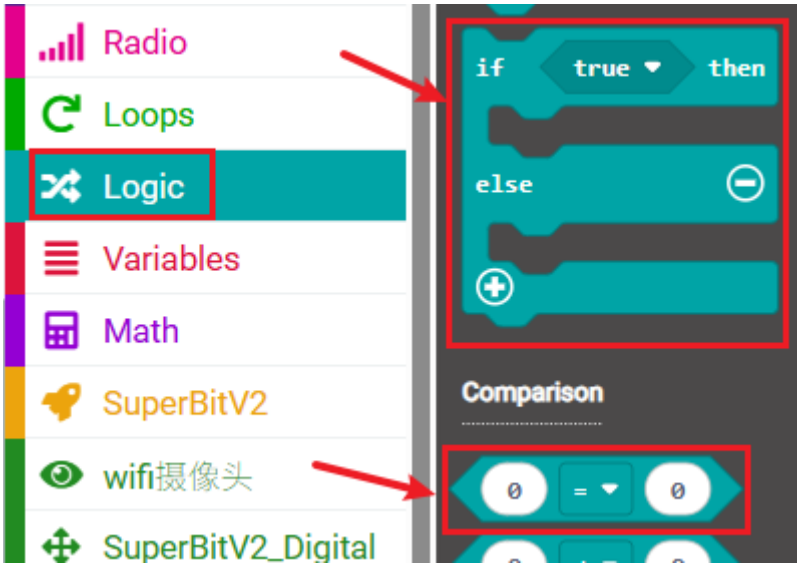
4.2 Blocks used

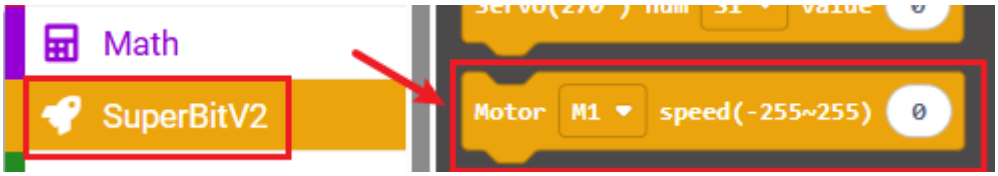
The location of the blocks required for this programming is shown in the figure below.

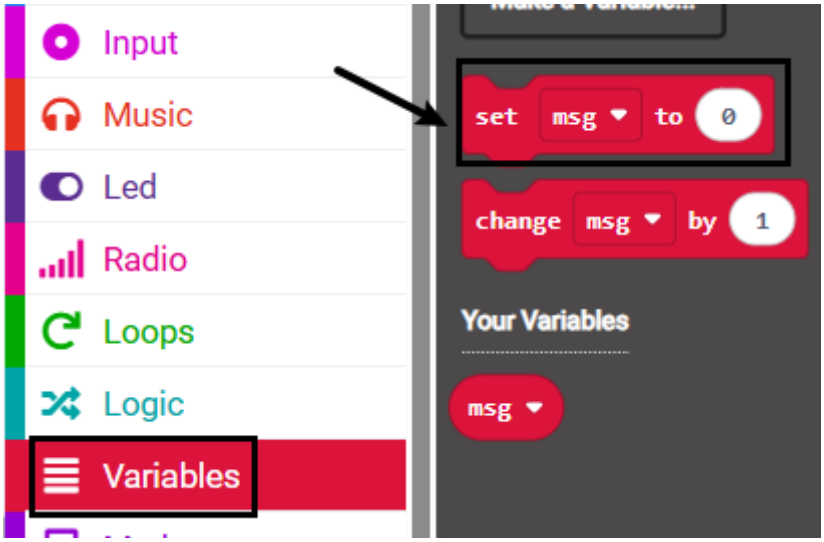

Search...
Basic
Input
Music
Led
Radio
Loops
Logic
Variables
Math
SuperBitV2
SuperBitV2_Digital
SuperBitV2_Analog
SuperBitV2_PWM
Neopixel
Extensions
Advanced
Basic
show number 0
show leds
show icon
show string "Hello!"
clear screen
forever
on start
pause (ms) 100


SuperBitV2
wifi摄像头
SuperBitV2_Digital
SuperBitV2_Analog
SuperBitV2_PWM
Neopixel
Extensions
Advanced
SET SerialPort
Set TXbuffSize size 0
Set RXbuffSize size 0
init SerialPort sendpin P0 recvpin P0 buadrate 115200
SET WIFI MODE
Set Wifi Mode Selcet APOnly


Math
SuperBitV2
wifi摄像头
SuperBitV2_Digital
SuperBitV2_Analog
SuperBitV2_PWM
Neopixel
Extensions
Advanced
Functions
GET STA IP
GET AP IP
GET WIFI Version
GET_Version
GET Data
GET Control Data
Car control
Control_Car forword

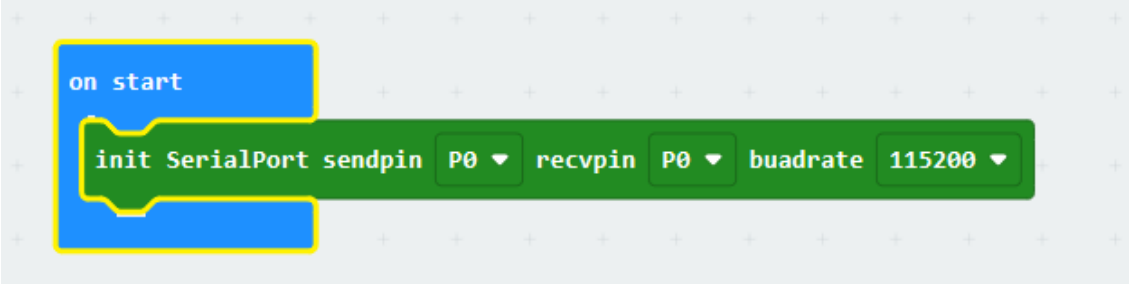

Radio
Loops
Logic
Variables
Math
SuperBitV2
wifi摄像头
SuperBitV2_Digital
if true then
else
Comparison
0 = 0


Math
SuperBitV2
Motor M1 speed(-255~255) 0

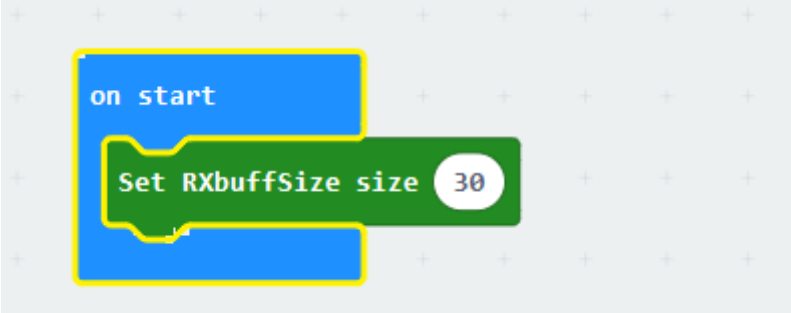

Input
Music
Led
Radio
Loops
Logic
Variables
set msg to 0
change msg by 1
Your Variables
msg

- ## 4.3 Introduction to main building blocks

- **Serial port initialization building block** This is used to define the pins for serial communication, and communicate with the WiFi camera. The baud rate is 115200 by default and cannot be changed

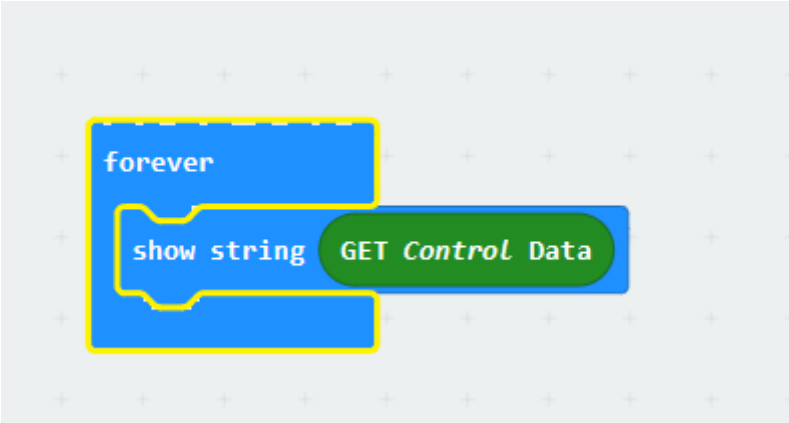


- **Set the size of the serial port receiving buffer** This building block is used to define the size of a packet of data that can be accepted for transparent transmission, for example

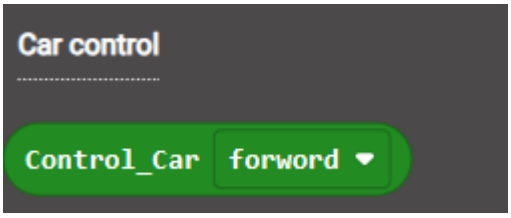


This defines that the maximum size of a packet is 30 characters. Exceeding this will result in incomplete data reception. **This value cannot be less than 25, otherwise the IP information will also be incomplete**

- **Block for obtaining transparent data** This block is mainly used to obtain information sent by the host computer to microbit, and transmits information through the wifi camera as the intermediate information



- **Block for car control** This block is mainly used to receive instructions sent by the app and transmit them to the car



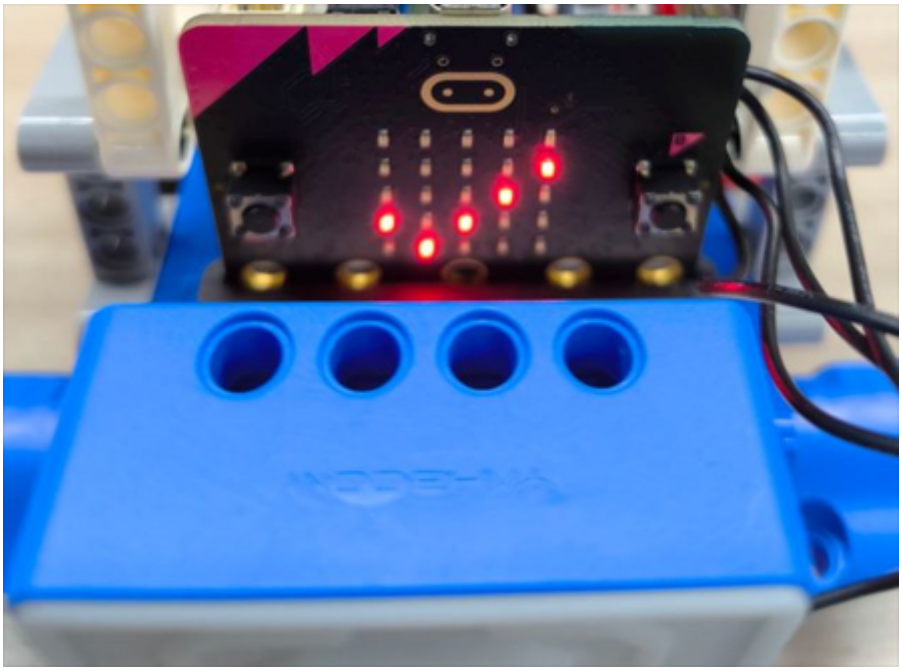
- You can know the function of other blocks by looking at their names. For how to use them, you can refer to the source code provided in this tutorial. This tutorial will not elaborate on them

4.4 Combined blocks

You can also directly open the **microbit-wifi_superkit_AP.hex** file provided in this experiment and drag it into the browser that opens the URL, and the program diagram of this project source code will be automatically opened

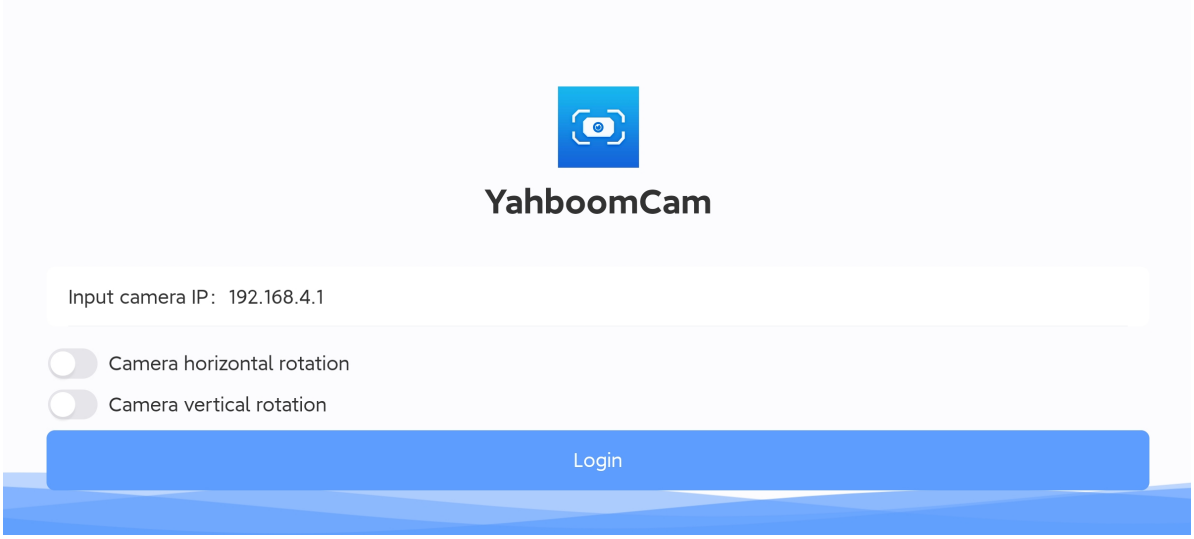
5. Experimental phenomenon

1. After powering on, wait until the microbit screen display no longer displays anything before you can control the app connection. At the beginning, the display is a heart icon. After successful startup, it will display its firmware version number. After waiting for a check icon to be displayed, it will start to display the default IP address of the hotspot AP mode "ap_ip:192.168.4.1". (This tutorial recommends using a simple self-heating AP mode for connection. To connect to WIFI using STA mode, see the following extended content for STA mode connection)



2. Download the APP: For Android, please use the browser to scan the following QR code to download and install YahboomCam. For Apple, please use the camera to scan the QR code to enter the App Store to download and install, or search for "YahboomCam" in the App Store to find "YahboomCam".

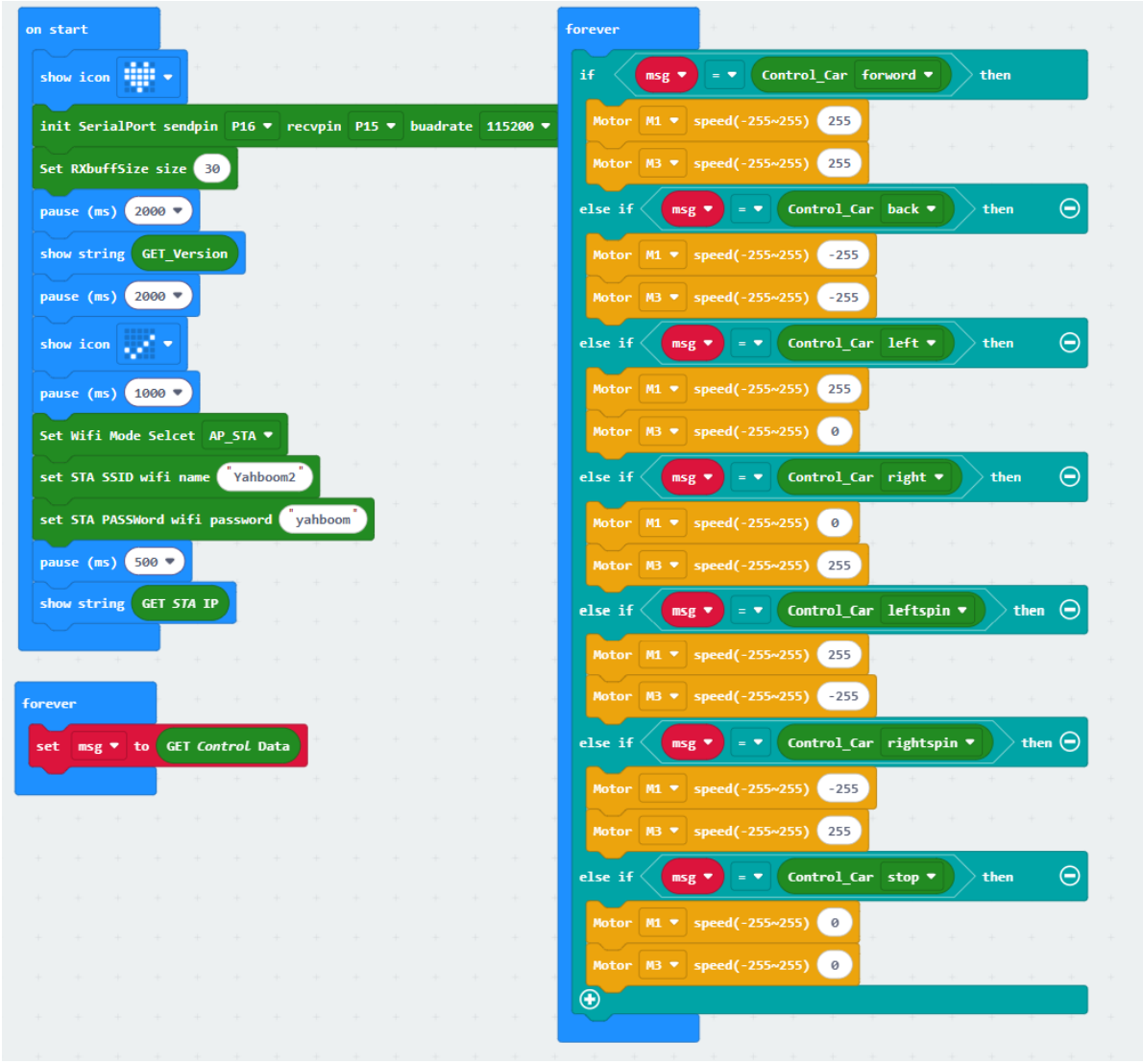
3. Use your mobile phone to connect to the hotspot (the default hotspot name is Yahboom_ESP32_WIFI, without a password). Open YahboomCam, enter the camera IP and fill in the default IP address 192.168.4.1 displayed by microbit, and then click Login.



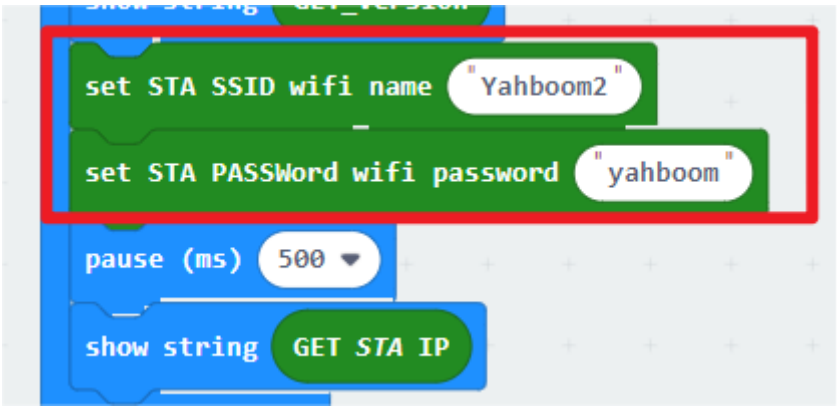
4. The page after logging in is as follows. You can control the car remotely by clicking the control button on the interface, and do operations such as forward, backward, left turn, right turn, etc. (the servo cannot control the car), and the real-time image of the camera is also displayed on the app page. The camera can be controlled by the horizontal and vertical control switches of the camera.

6. Extension: STA mode connection

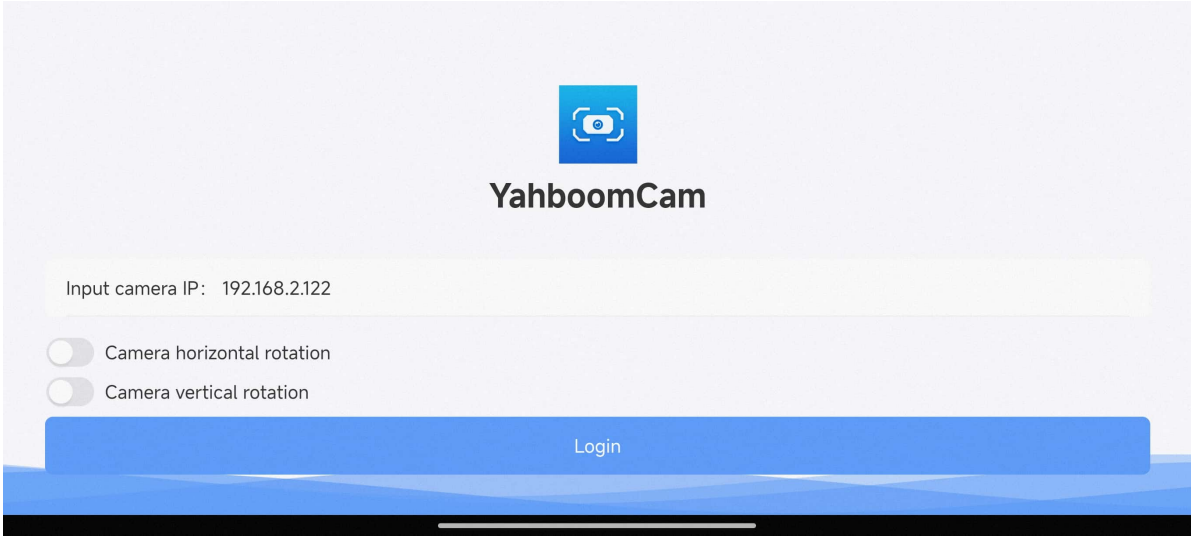
1. Open the URL https://makecode.microbit.org/# in the browser
2. Then drag the **microbit-wifi_superkit_STA.hex** file provided in this experiment into the browser that opens the URL, and the program diagram of this project source code will be automatically opened



3. Set the WiFi name and password to be connected. The WiFi name and password must be filled in correctly according to the WiFi you want to connect to, otherwise there is no way to correctly connect to the WiFi and display the IP address. If the connection is successful, the IP address of sta_ip:192.168.x.x will be displayed. If the connection is unsuccessful, sta_ip:null will be displayed. You need to check whether the WiFi name and address filled in the program are correct.



4. Connect your phone to the WIFI set above, and make sure that the phone and the robot are connected to the same WiFi. Open YahboomCam, enter the camera IP and fill in the sta_ip address displayed by microbit (pay attention to the content displayed by microbit, and it will start to display "sta_ip: 192.168.x.x" after you see the icon tick), then click to log in. (The IP in the picture is only an example, the specific IP should be based on the microbit display)



5. After logging in, you can see the following page. You can control the car remotely by clicking the control button on the interface, such as forward, backward, left turn, right turn, etc. (the servo cannot be controlled by the car), and the real-time image of the camera is also displayed on the app page. You can control the camera through the horizontal and vertical control switches of the camera.