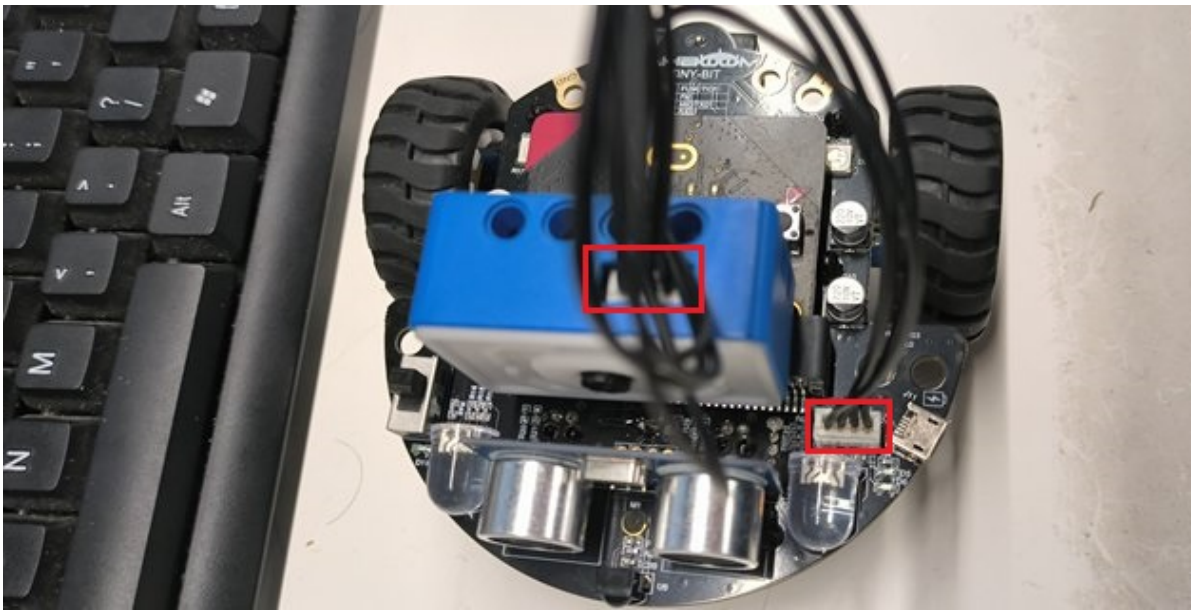
7. Face tracking

7.1 Experimental preparation

- tinybit car
- microbit
- wifi camera

7.2 Experimental installation

Connect the camera to the car, as shown below



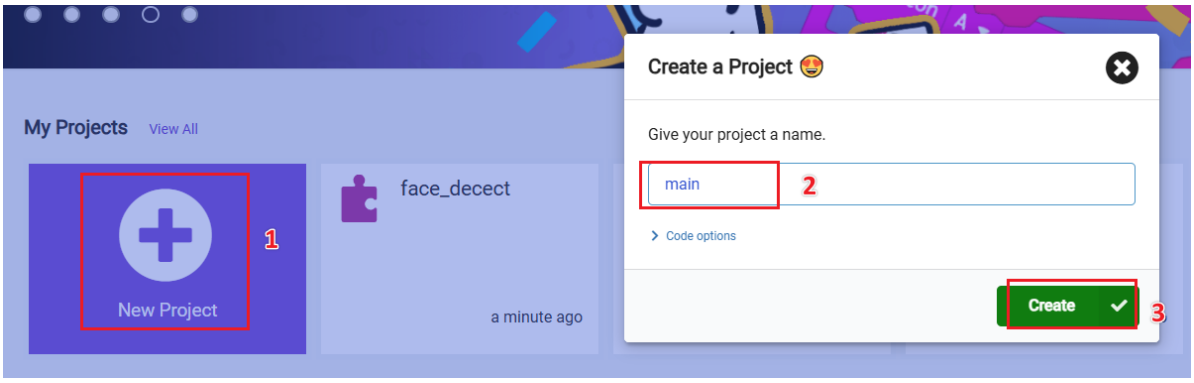
Note: The switch at the bottom of the car must be turned to TXD to work with the image transmission camera

7.3 Import and briefly explain the microbit building blocks

7.3.1 Open the programming website

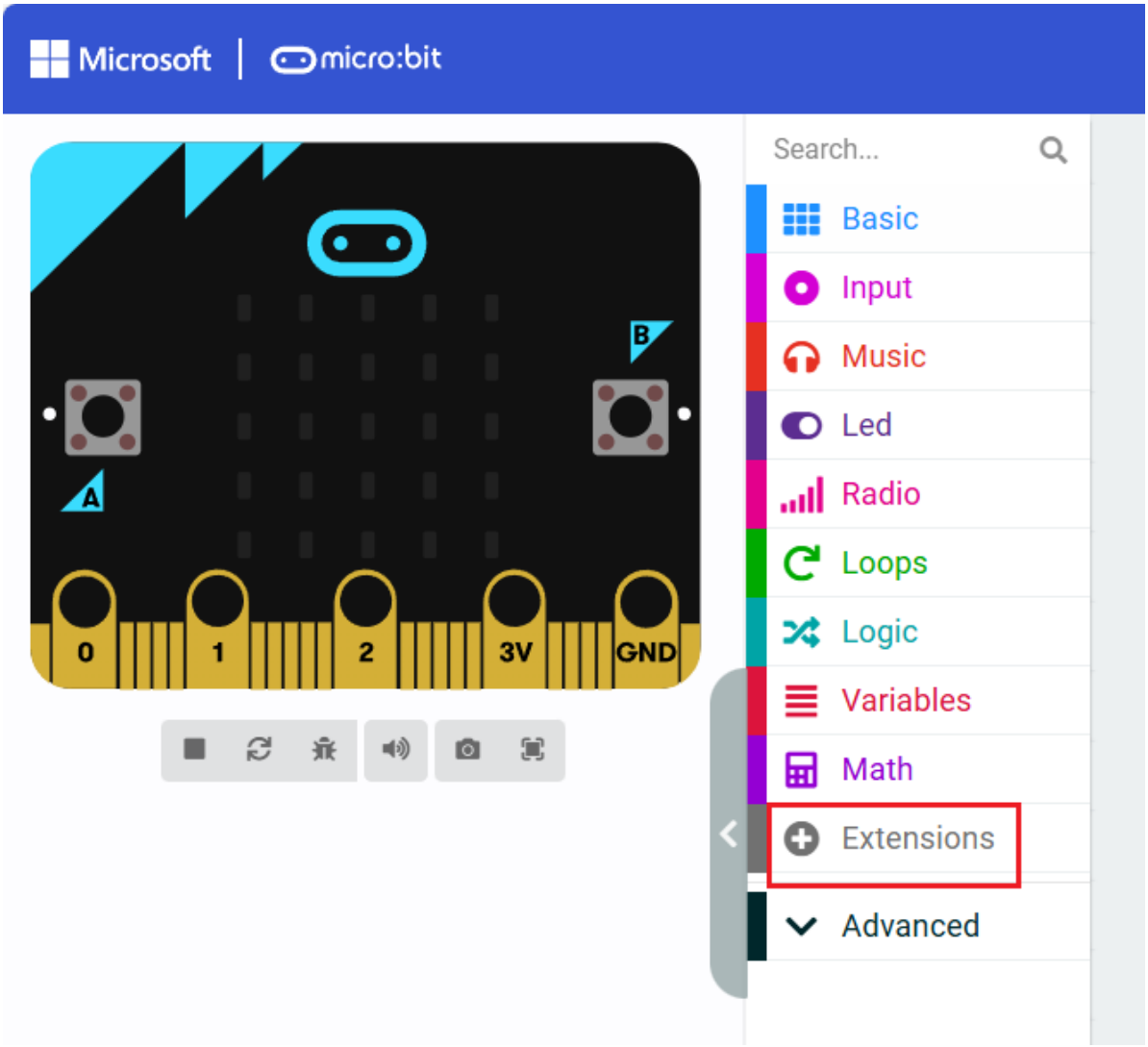
https://makecode.microbit.org/#

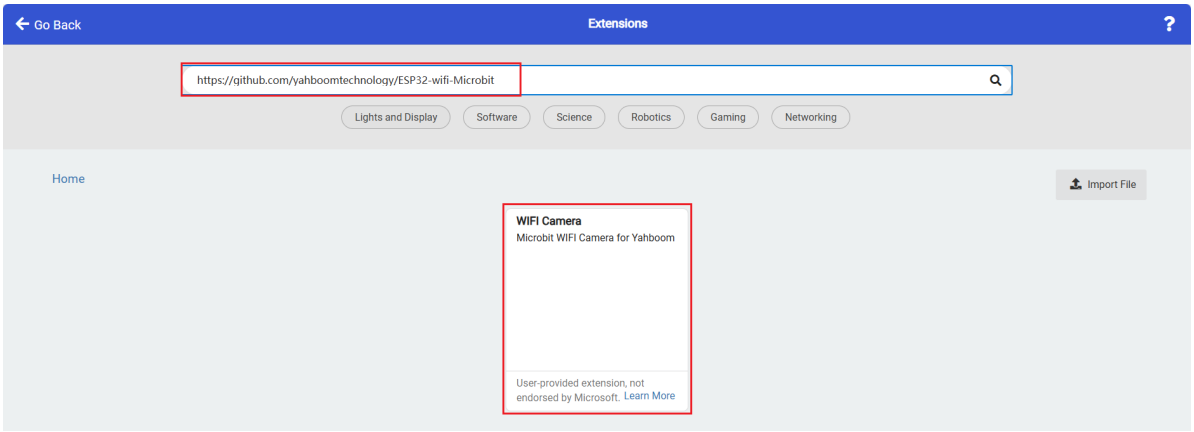
7.3.2 Create a new project



7.3.3 Add the camera building block

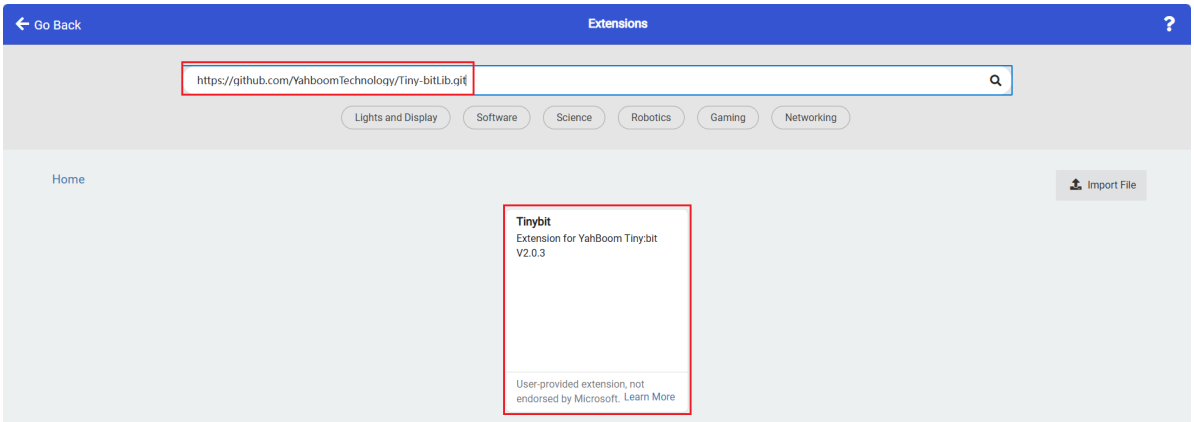
Website for the camera building block: https://github.com/yahboomtechnology/ESP32-wifi-Microbit





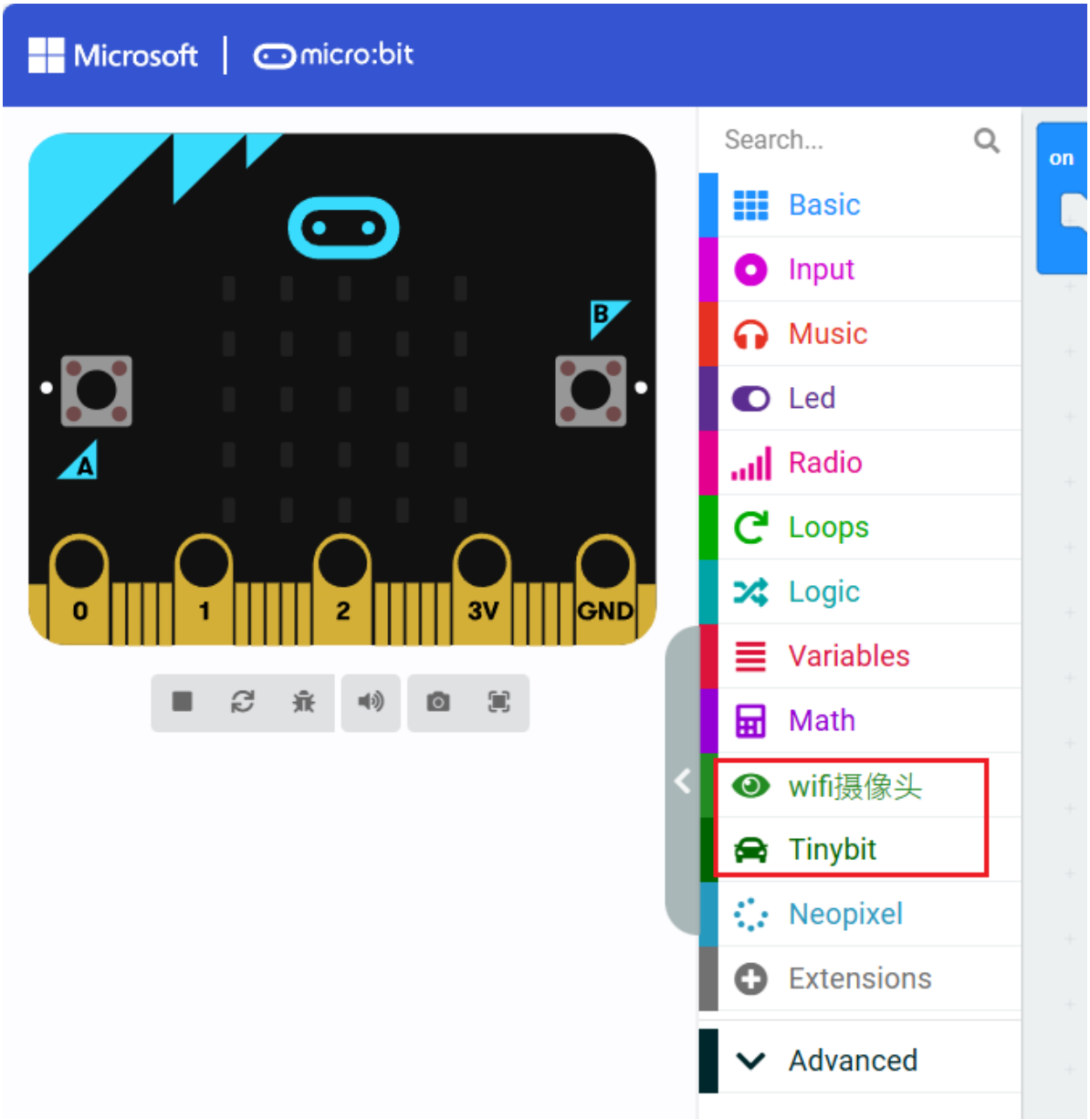
7.3.4 Add the building blocks of the car

The URL of the building blocks of the car: https://github.com/YahboomTechnology/Tiny-bitLib.git

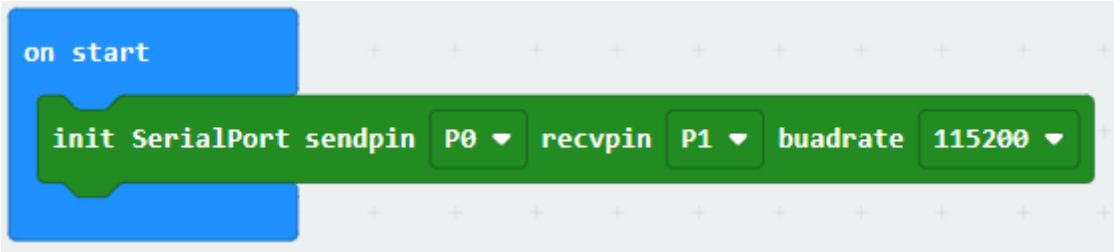


7.3.5 Introduction to the main building blocks

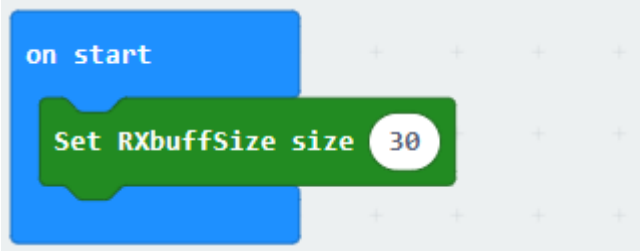
After successfully introducing the above building blocks, the result will be displayed as shown in the figure below



- **Serial port initialization building block** This is used to define the pin for serial communication. When communicating with the wifi camera, the baud rate is 115200 by default and cannot be changed

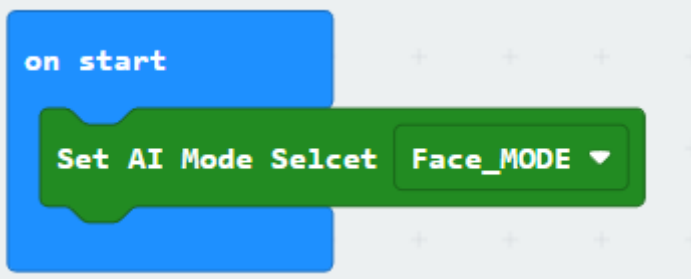


- **Set the size of the serial port receiving buffer** This block is used to define the size of a packet of data that can be accepted for transparent transmission, for example

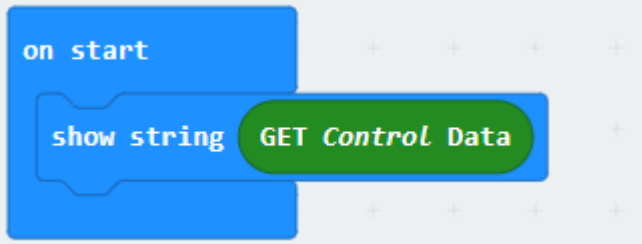


This defines that the maximum size of a packet is 30 characters. Exceeding this will result in incomplete data reception. **This value cannot be less than 25, otherwise the IP information will also be incomplete**

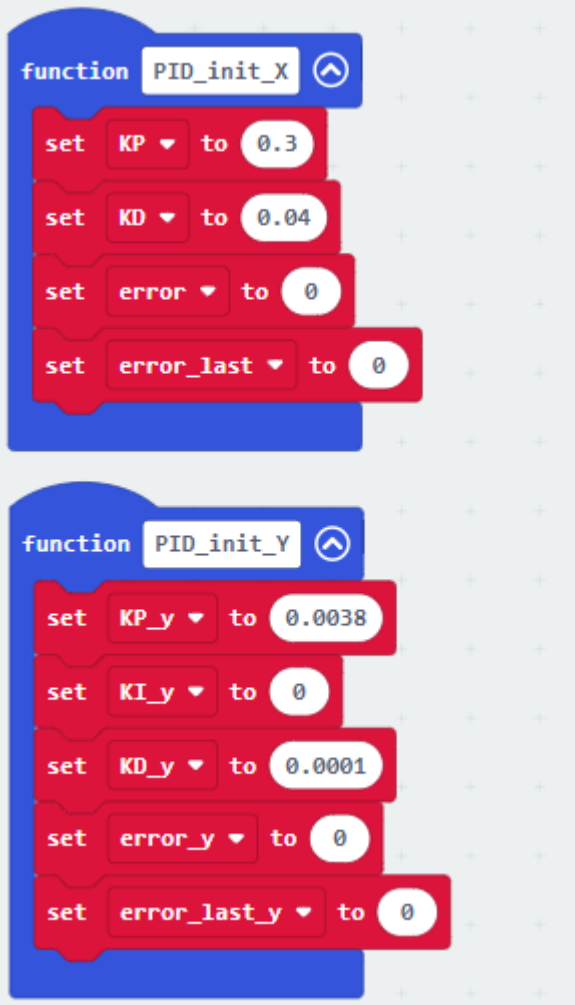
- **Set AI mode** This block is used to set the current AI visual play mode, which can be set to different modes



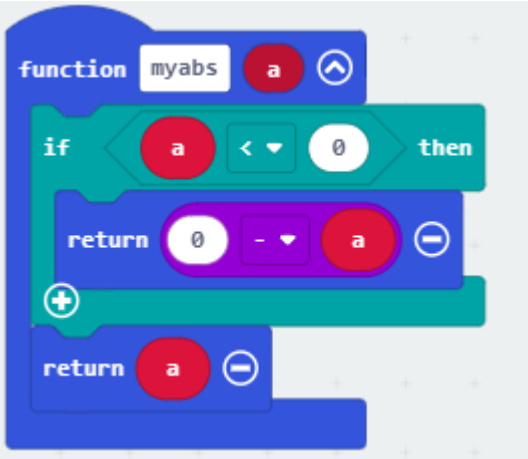
- **Block for obtaining transparent data** This block is mainly used to obtain information sent by the host computer to microbit, and transmits information through the wifi camera as an intermediate



- **Initialize PID** Initialize the parameters of the PID controller in the x-axis and y-axis directions



- **myabs function** Calculate the absolute value of an integer



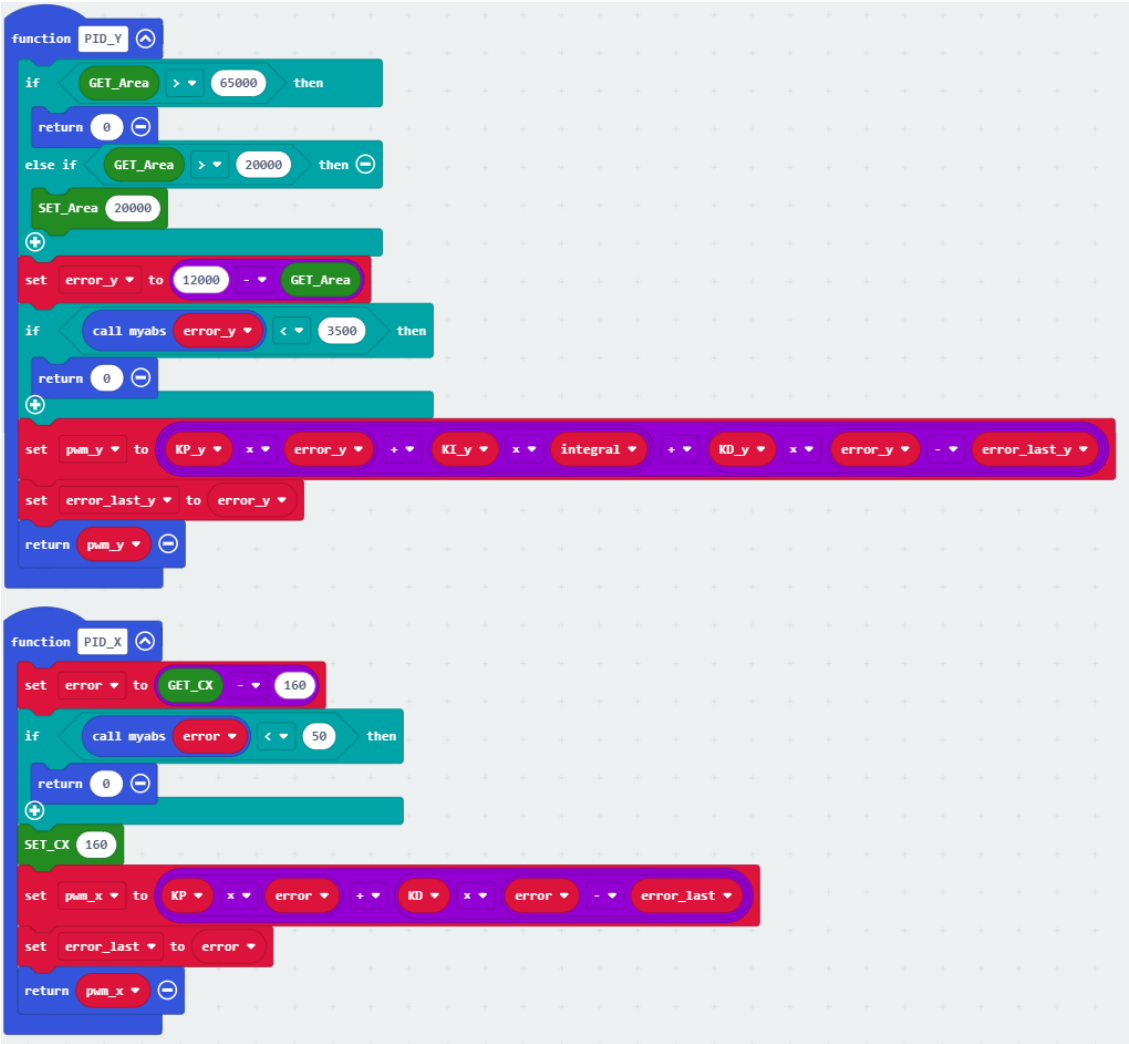
- **PID output** Calculate the PID control output in the x-axis and y-axis directions, that is, the adjustment amount of the car in the horizontal and vertical directions
    - PID_Y: If the area of the recognition box is greater than 65000, it returns 0 and does not make any adjustments; if the area of the recognition box is greater than 20000, the area is set to 20000;

      Calculate the error error_y between the area of the recognition box and the expected area; if the error is less than 3500, return 0, which means the error is small enough and no adjustment is required;

      Calculate the output pwm_y of the PID controller, and then use error_last_y to store the error value.
    - PID_X: Calculate the error error between the center point position of the recognition box in the X direction and the expected position 160;

      If the absolute value of the error is less than 50, return 0 without adjustment;

      Set the position of the center point of the recognition box to 160, then calculate the output pwm_x of the PID controller, and use error_last to store the last error value.

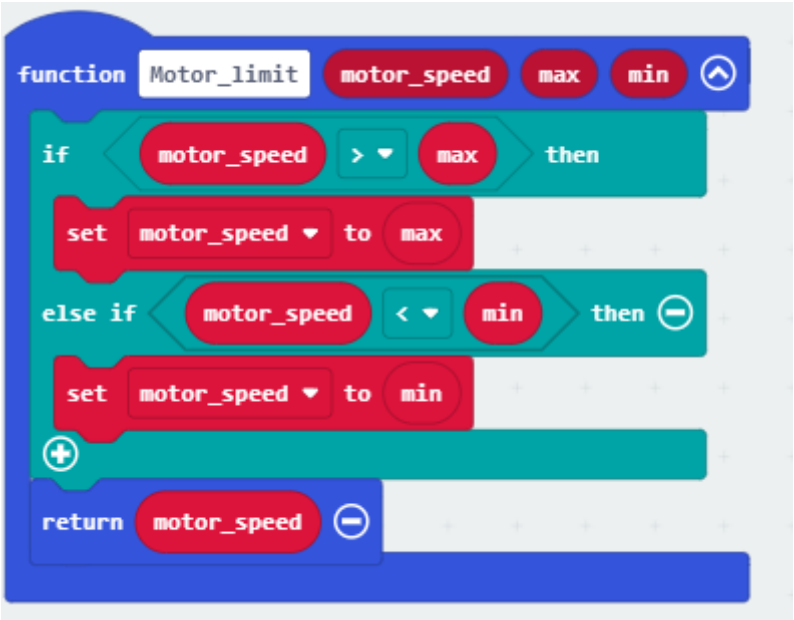


- **Speed Limit** Limit the motor speed to ensure that it does not exceed the maximum value max or fall below the minimum value min

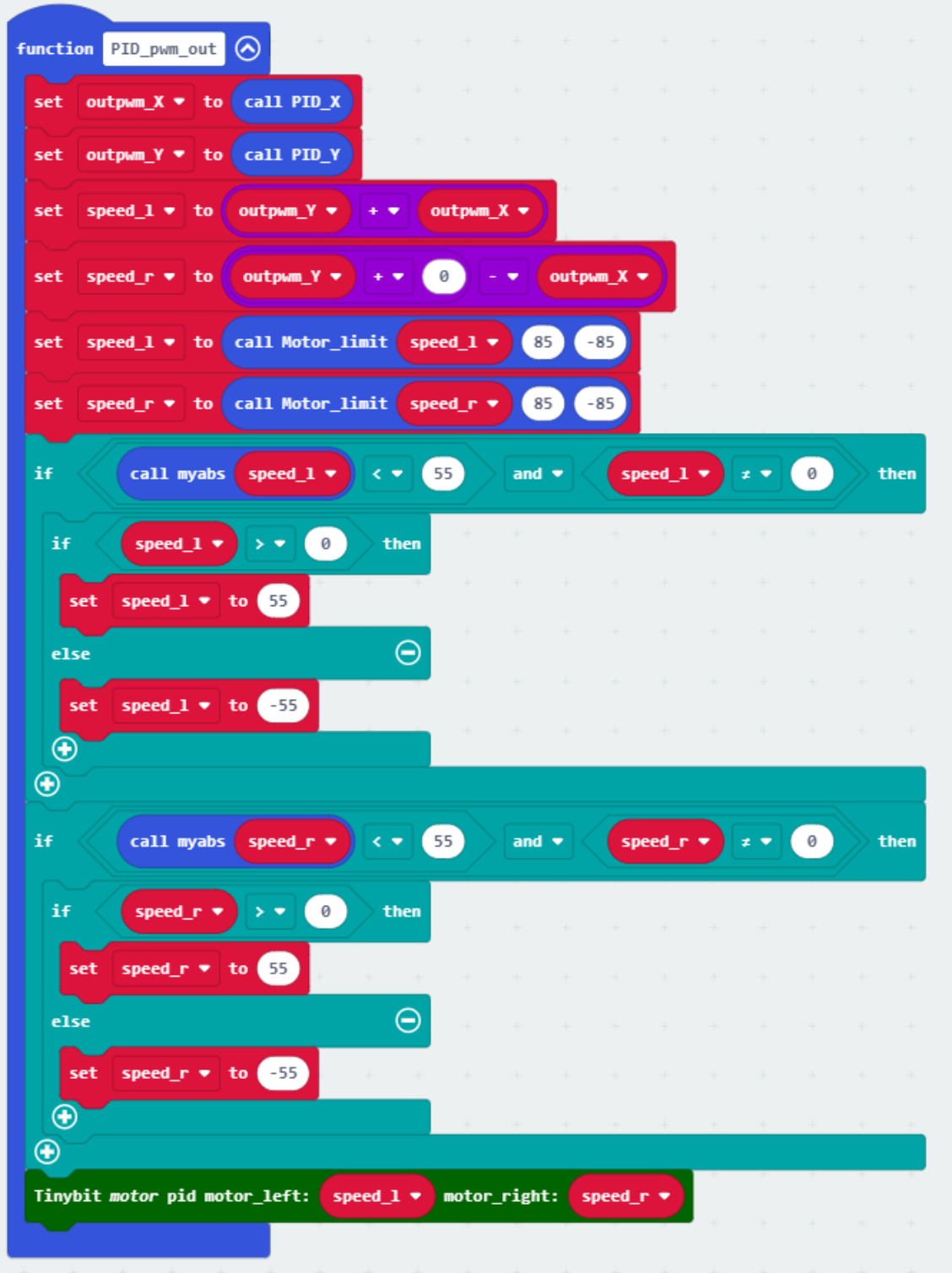
  If motor_speed is greater than max, set motor_speed to max;

  If motor_speed is less than min, set motor_speed to min;

  Returns the motor_speed value after limiting.



- **Motor Control** Calculates the PID control outputs for the x- and y-axis directions and calculates the speeds of the left and right motors based on these outputs. Limit the motor speed to between -85 and 85. If the absolute value of the speed is less than 55 and not 0, set it to 55 or -55



- **Method to open the source code provided by the tutorial**

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

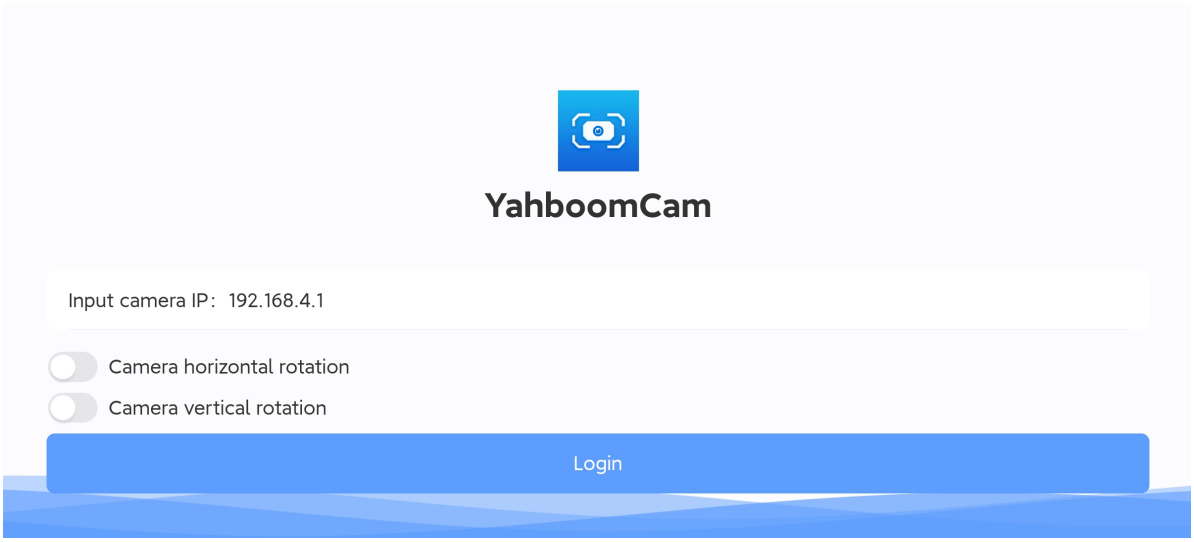
7.4 Experimental effect

1. Turn on the switch of the car, wait for the pattern corresponding to the mode setting to be displayed on the microbit, and then you can connect to the APP. After successful startup, the

camera module will generate a hotspot. The default IP address of the hotspot AP mode is 192.168.4.1 (this tutorial recommends using a simple self-heating hotspot AP mode for connection. To connect to WIFI using STA mode, see the extended content of STA mode connection below)

2. Download 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" or in the App Store to find "YahboomCam".

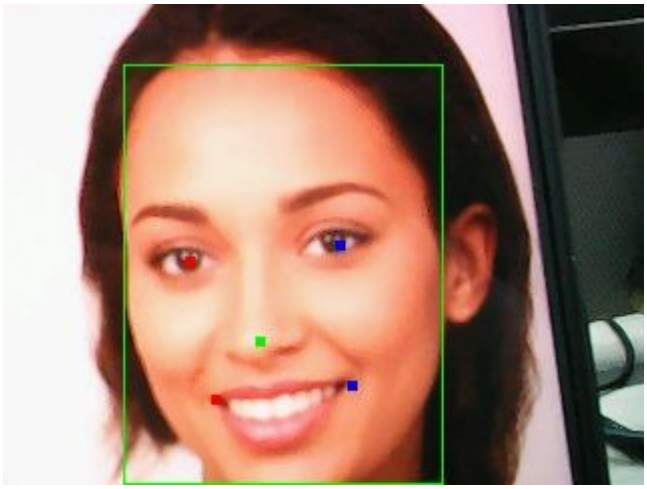
3. Use a mobile phone to connect to the hotspot (the default hotspot name is Yahboom_ESP32_WIFI, without a password). Open YahboomCam, enter the camera IP and fill in the default IP address 192.168.4.1 displayed by microbit, and then click Login.



4. The page after login is as follows, and you can see the camera screen.

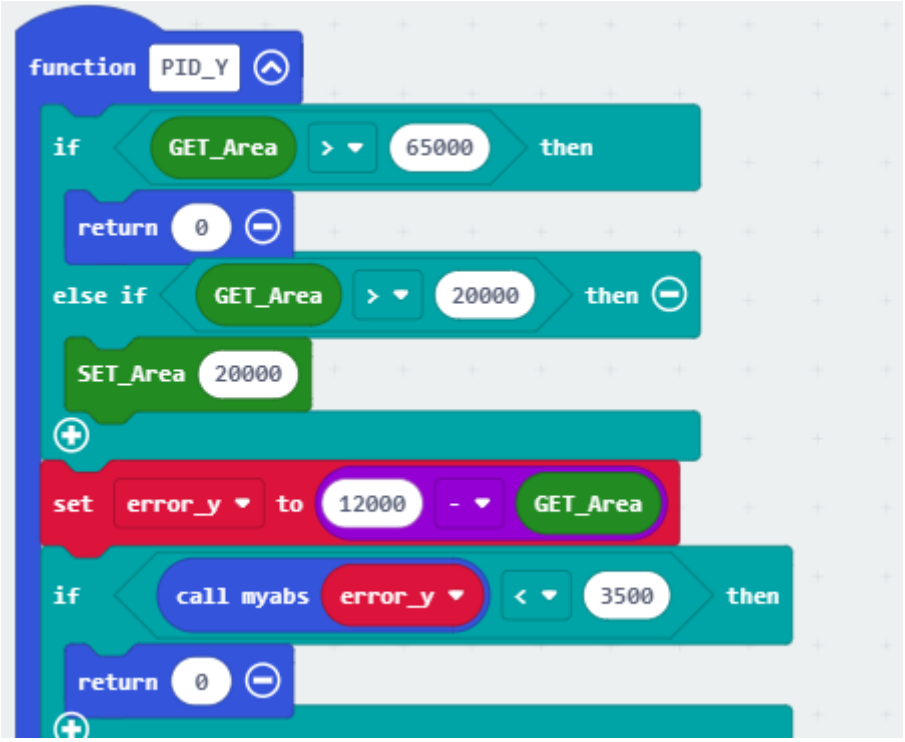
After detecting the face, the car tracks the face. If it cannot be detected, the car stops.

Face:



Note: You need to open the screen to start recognition. If the screen is closed in the middle, the recognition will also be closed.

5. If you want to adjust the car's following distance, you can adjust the value of the identification box area in the PID_Y function (if necessary, you can fine-tune the PID, but it is best not to adjust it)

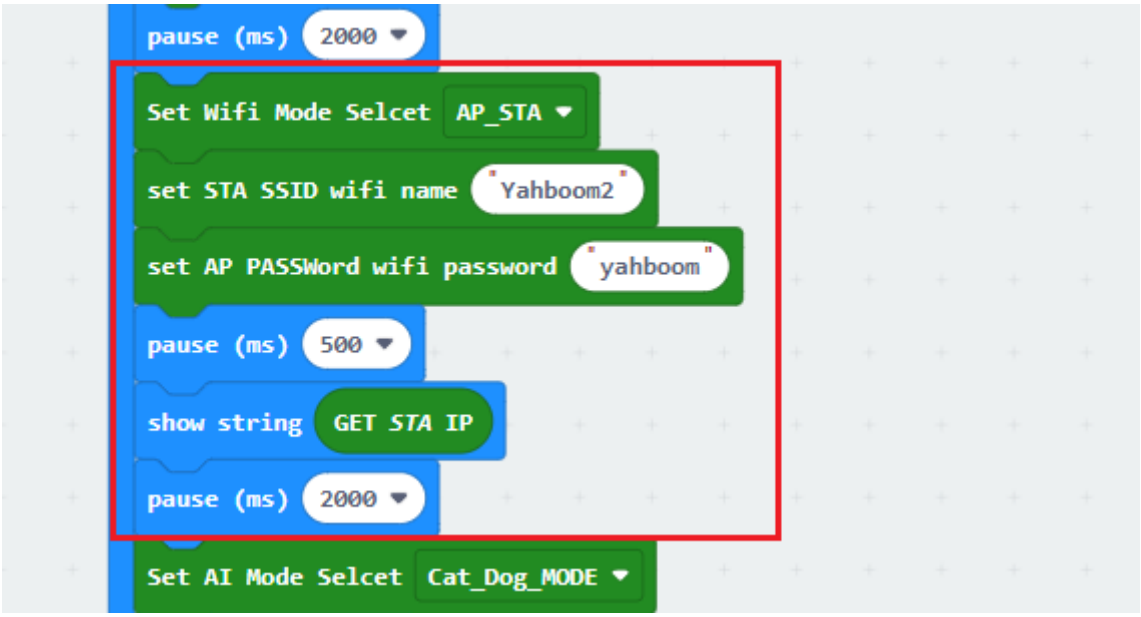


The larger the value, the smaller the following distance;

The smaller the value, the larger the following distance.

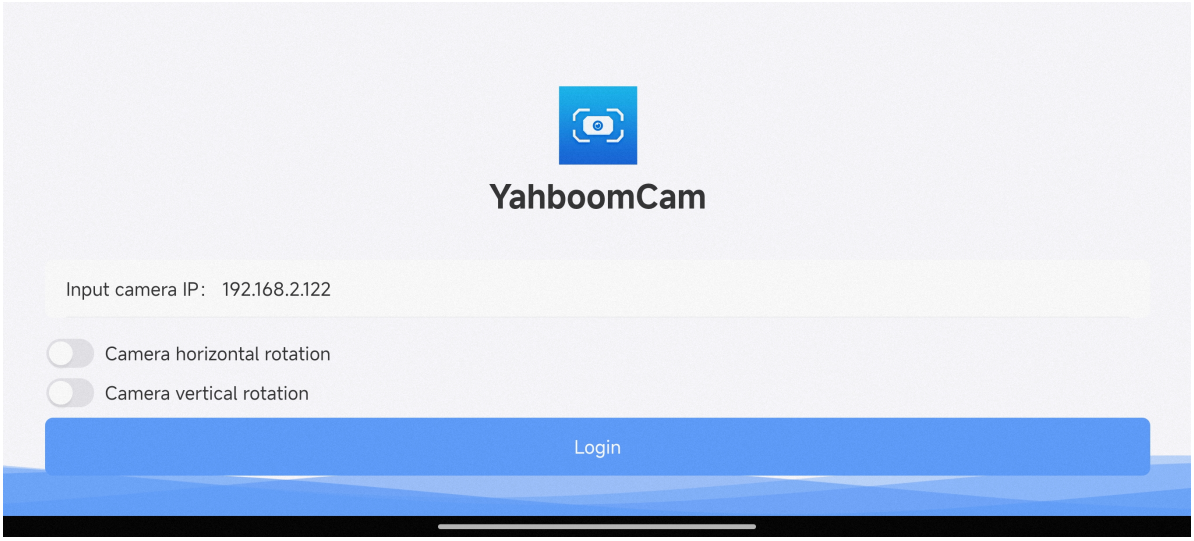
7.5 Extension: STA mode connection

1. Change the source code, drag the .hex file provided in this experiment into the browser that opens the URL, set the WiFi name and password to connect, and 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 connect to the WiFi correctly 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.



2. Connect the mobile phone to the WIFI set above, and make sure that the mobile 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, after seeing 1, 9, 2 floating continuously, it means the IP address is being

displayed), then click to log in. (The IP in the picture is only an example, the specific IP should be based on the microbit display)



3. After logging in, you can see the following page, and you can see the camera screen.