Micro:bit handle control

1. Learning objectives

In this course, we mainly learn how to use MakeCode graphical programming to realize the control of Biped robot by micro:bit handle.

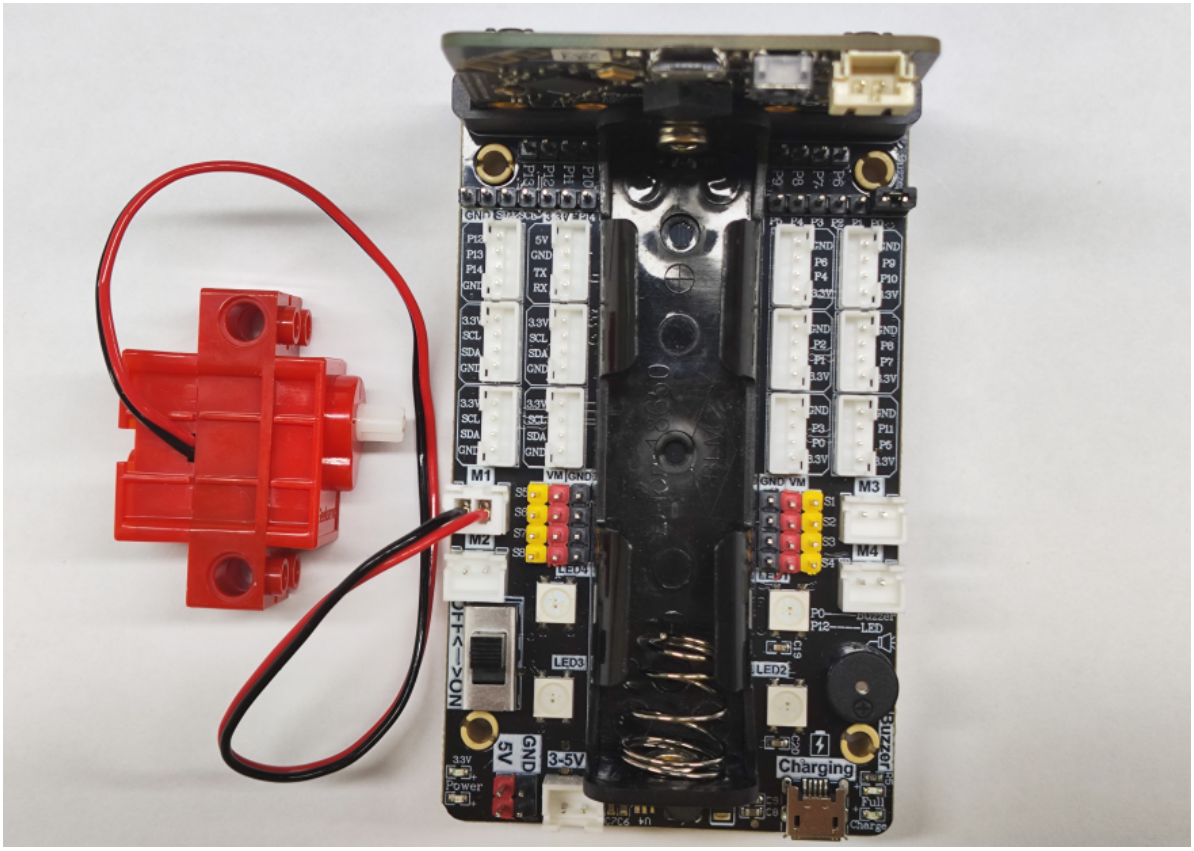
2. Building blocks

For the building blocks steps, please refer to the installation drawings of **[Assembly Course]--[Biped robot]** in the materials or the building blocks installation book.

3. Motor wiring

Insert the motor wiring on the left side of the car into the M1 interface of the Super:bit expansion board, with the black wire close to the battery side;

As shown below:



4. Programming

Method 1 Online programming:

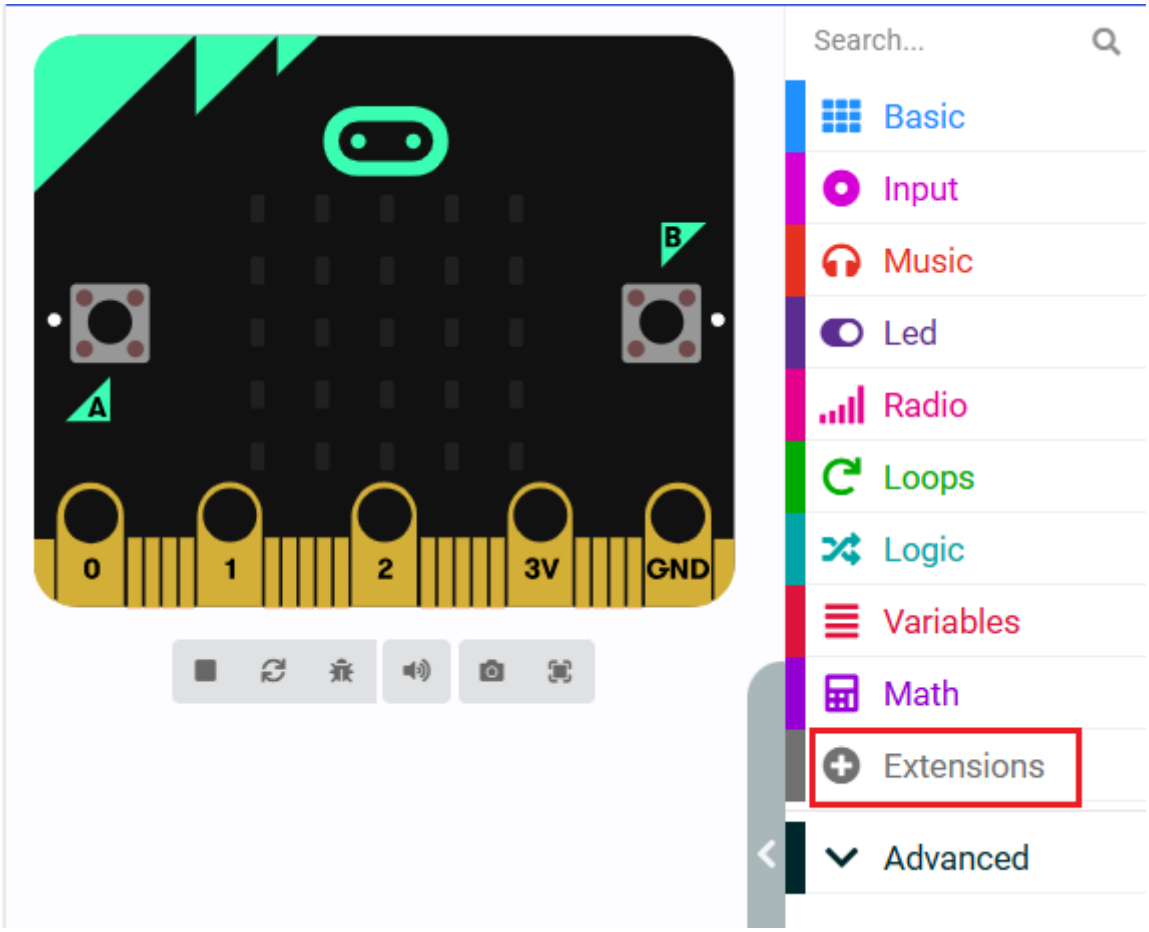
First, connect micro:bit to the computer via a USB data cable, and a U disk will pop up on the computer. Click the URL in the U disk: https://makecode.microbit.org/ to enter the programming interface. Then, add the Yahboom software package to start programming.

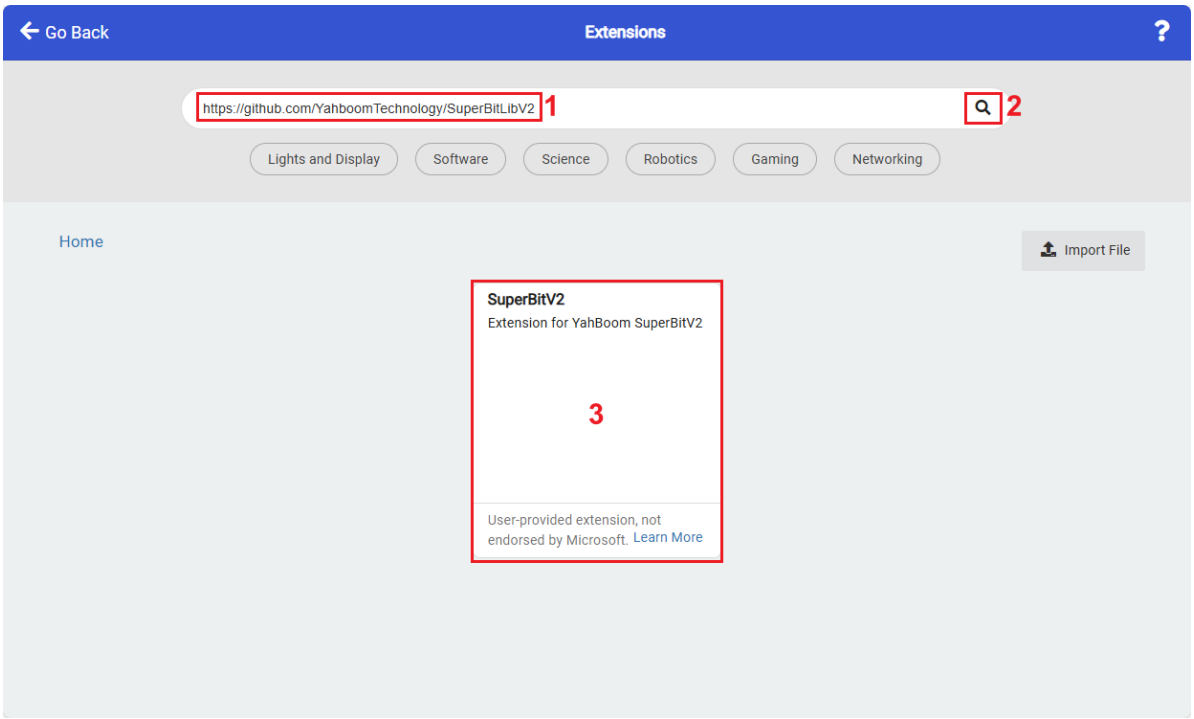
Method 2 Offline Programming:

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

superbit kit expansion package: https://github.com/YahboomTechnology/SuperBitLibV2
Handle expansion package: https://github.com/YahboomTechnology/GHBitLib

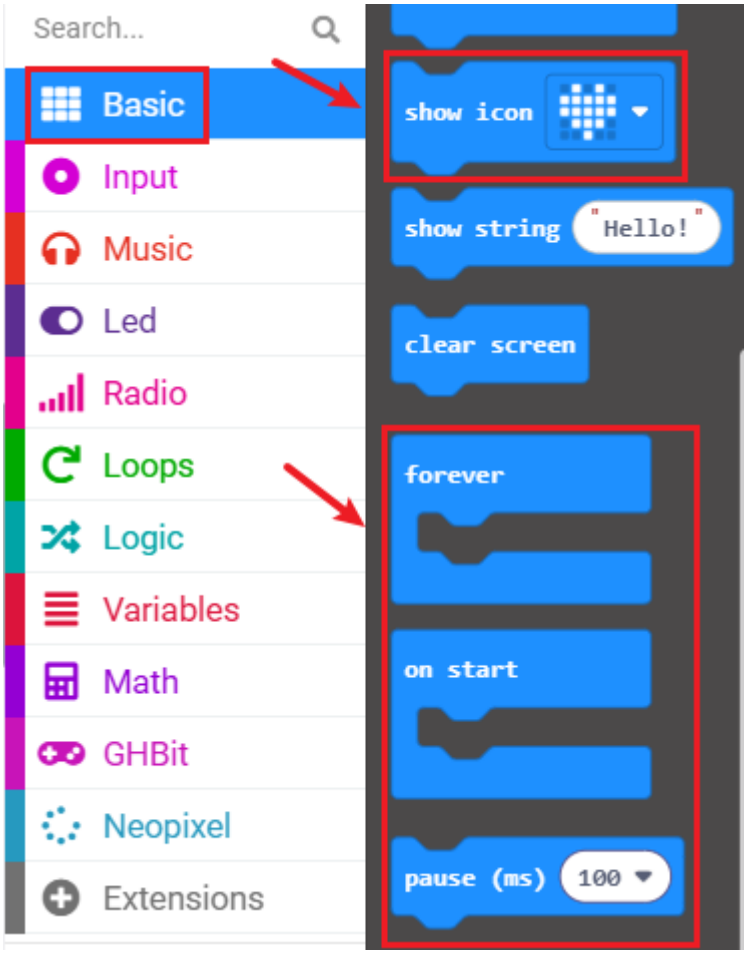
4.1 Add expansion package





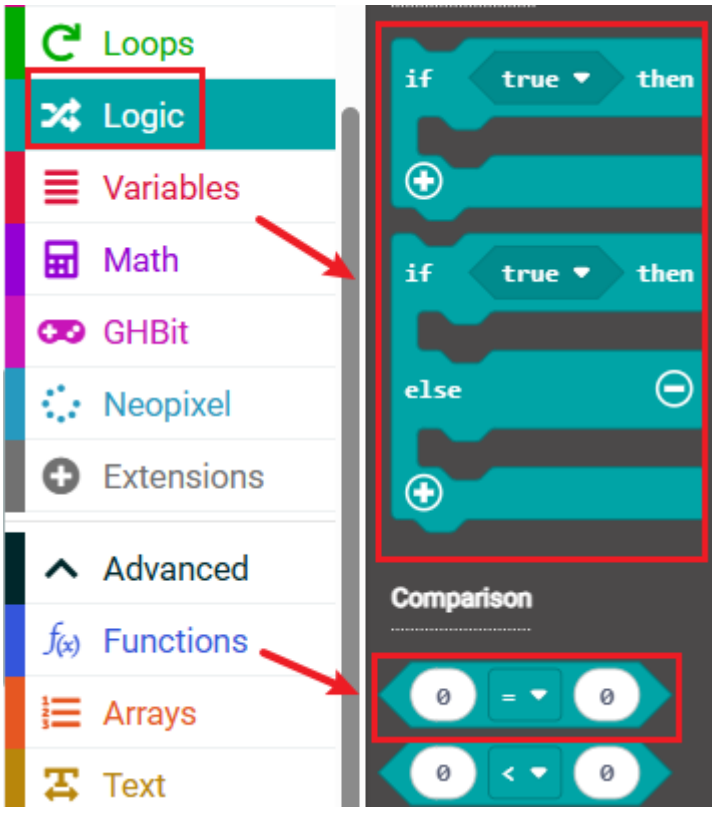
4.2 Building blocks used

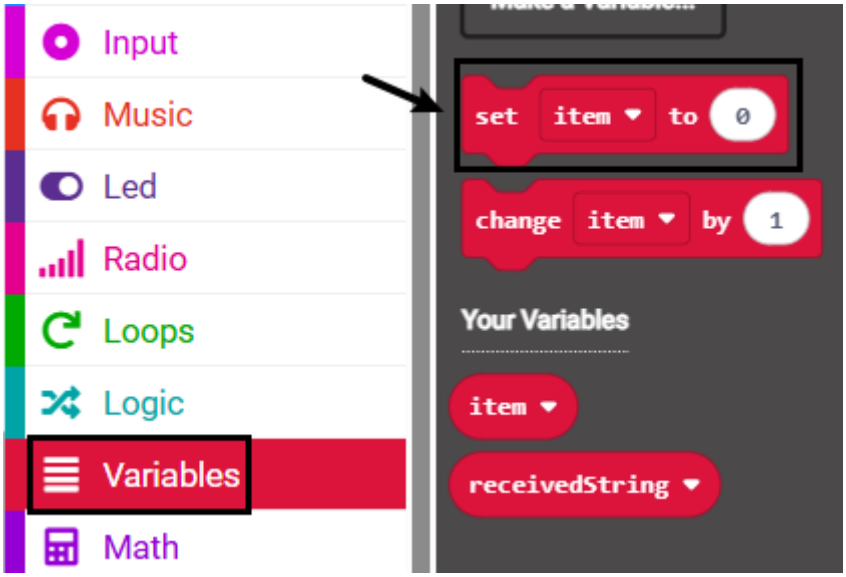
The location of the building blocks required for this programming is shown in the figure below.

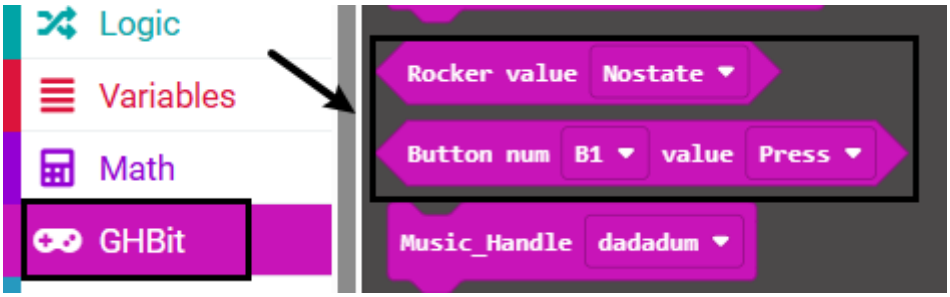

Search...
Basic
Input
Music
Led
Radio
Loops
Logic
Variables
Math
GHBit
Neopixel
Extensions
show icon
show string
"Hello!"
clear screen
forever
on start
pause (ms)
100


Basic
Input
more
Music
Led
Radio
on button
A
pres
on
shake

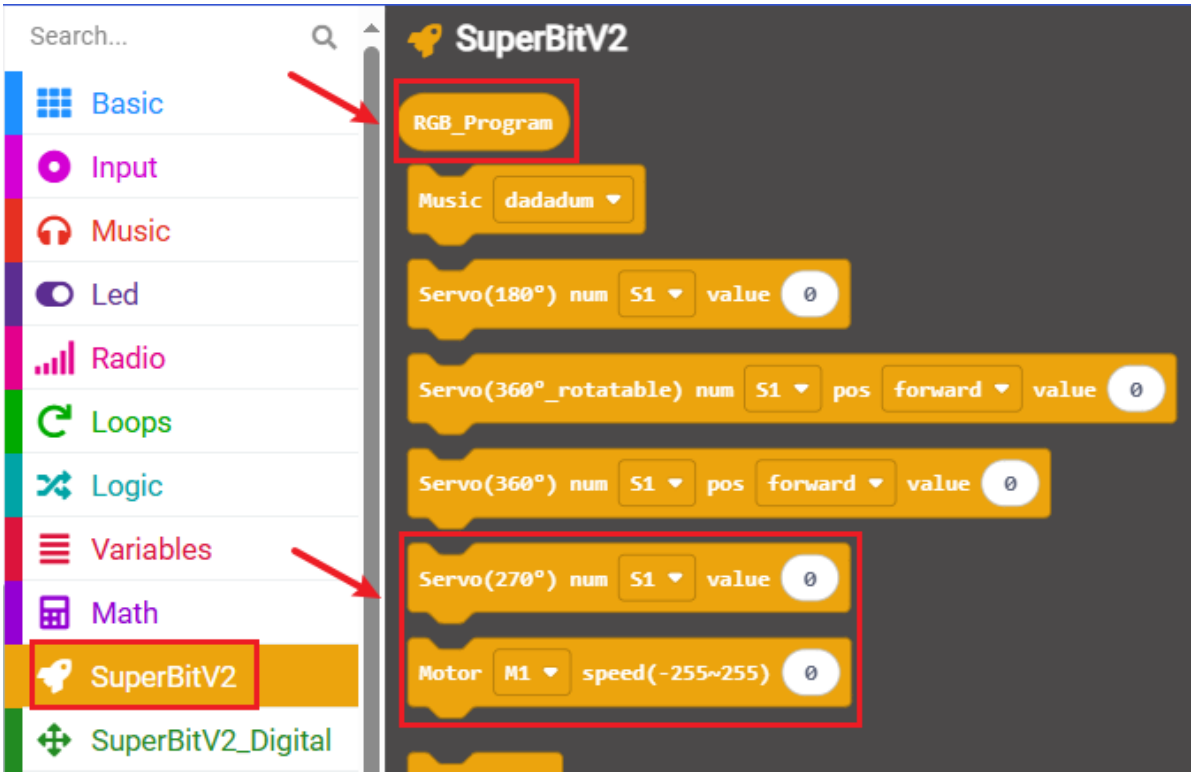

Led
Radio
Loops
Logic
Variables
Math
GHBit
Neopixel
Extensions
Advanced
Functions
Arrays
Text
Game
Images
Text
length of "Hello"
join "Hello" "World"
parse to number "123"
split "this" at " "
"this" includes " "
"this" find index of " "
"this" is empty
substring of "this" from 0 of length 10
compare "this" to " "


Loops
Logic
Variables
Math
GHBit
Neopixel
Extensions
Advanced
Functions
Arrays
Text
if true then
if true then
else
Comparison
0 = 0
0 < 0

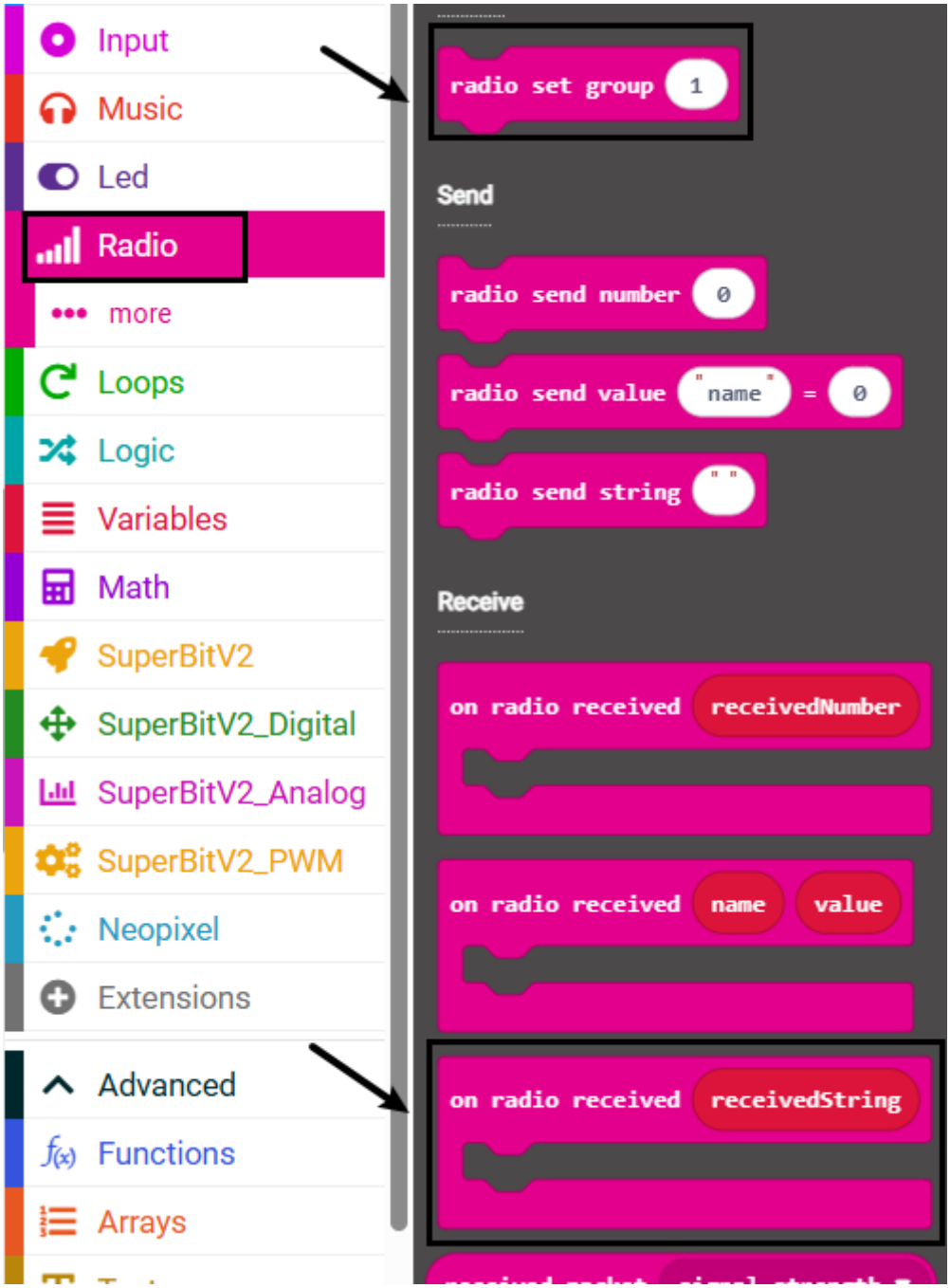

Input
Music
Led
Radio
Loops
Logic
Variables
Math
set item to 0
change item by 1
Your Variables
item
receivedString


Logic
Variables
Math
GHBit
Rocker value Nostate
Button num B1 value Press
Music_Handle dadadum


Basic
Input
Music
Led
Radio
Loops
on start
pause (ms) 100
show arrow North


Search...
Basic
Input
Music
Led
Radio
Loops
Logic
Variables
Math
SuperBitV2
SuperBitV2_Digital
SuperBitV2
RGB_Program
Music dadadum
Servo(180°) num S1 value 0
Servo(360°_rotatable) num S1 pos forward value 0
Servo(360°) num S1 pos forward value 0
Servo(270°) num S1 value 0
Motor M1 speed(-255~255) 0

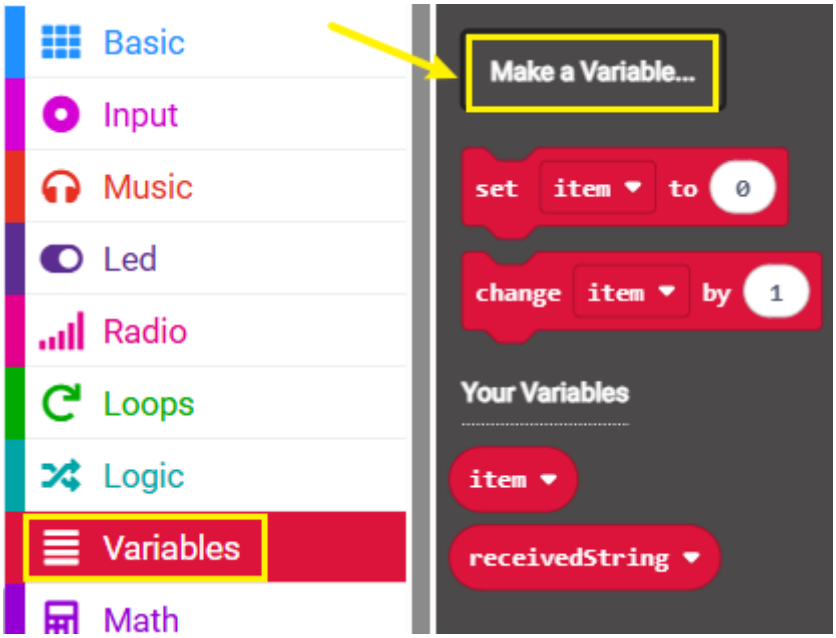

SuperBitV2
SuperBitV2_Digital
SuperBitV2_Analog
SuperBitV2_PWM
Neopixel
more
Extensions
strip show rainbow from 1 to 360
strip show color red
strip show bar graph of 0 up to 255
strip show
strip clear


Input
Music
Led
Radio
more
Loops
Logic
Variables
Math
SuperBitV2
SuperBitV2_Digital
SuperBitV2_Analog
SuperBitV2_PWM
Neopixel
Extensions
Advanced
Functions
Arrays
radio set group 1
Send
radio send number 0
radio send value "name" = 0
radio send string ""
Receive
on radio received receivedNumber
on radio received name value
on radio received receivedString

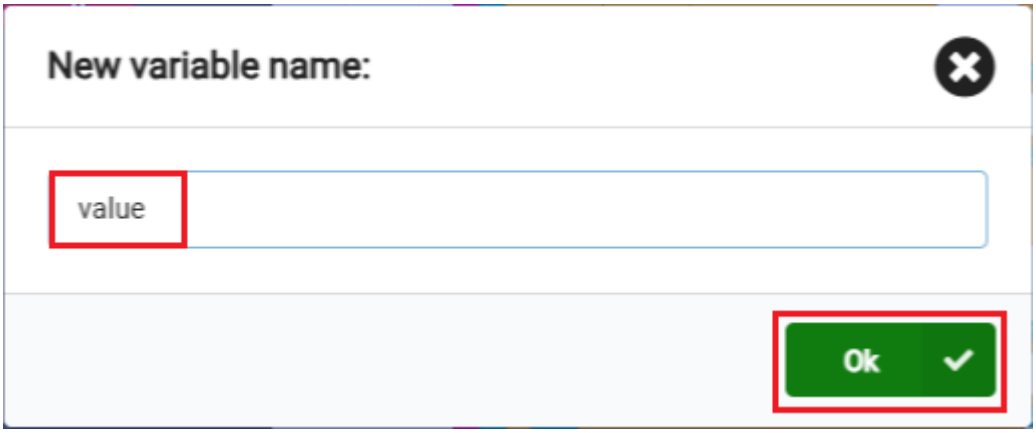

Search...
Basic
Input
Music
Led
Radio
more
more
radio set transmit power 7
radio set transmit serial number true
radio set frequency band 0
radio raise event
from source MICROBIT_ID_BUTTON_A

4.3 Add new variables

① Find the [Variable] option in the building block bar ---- [Set variable]

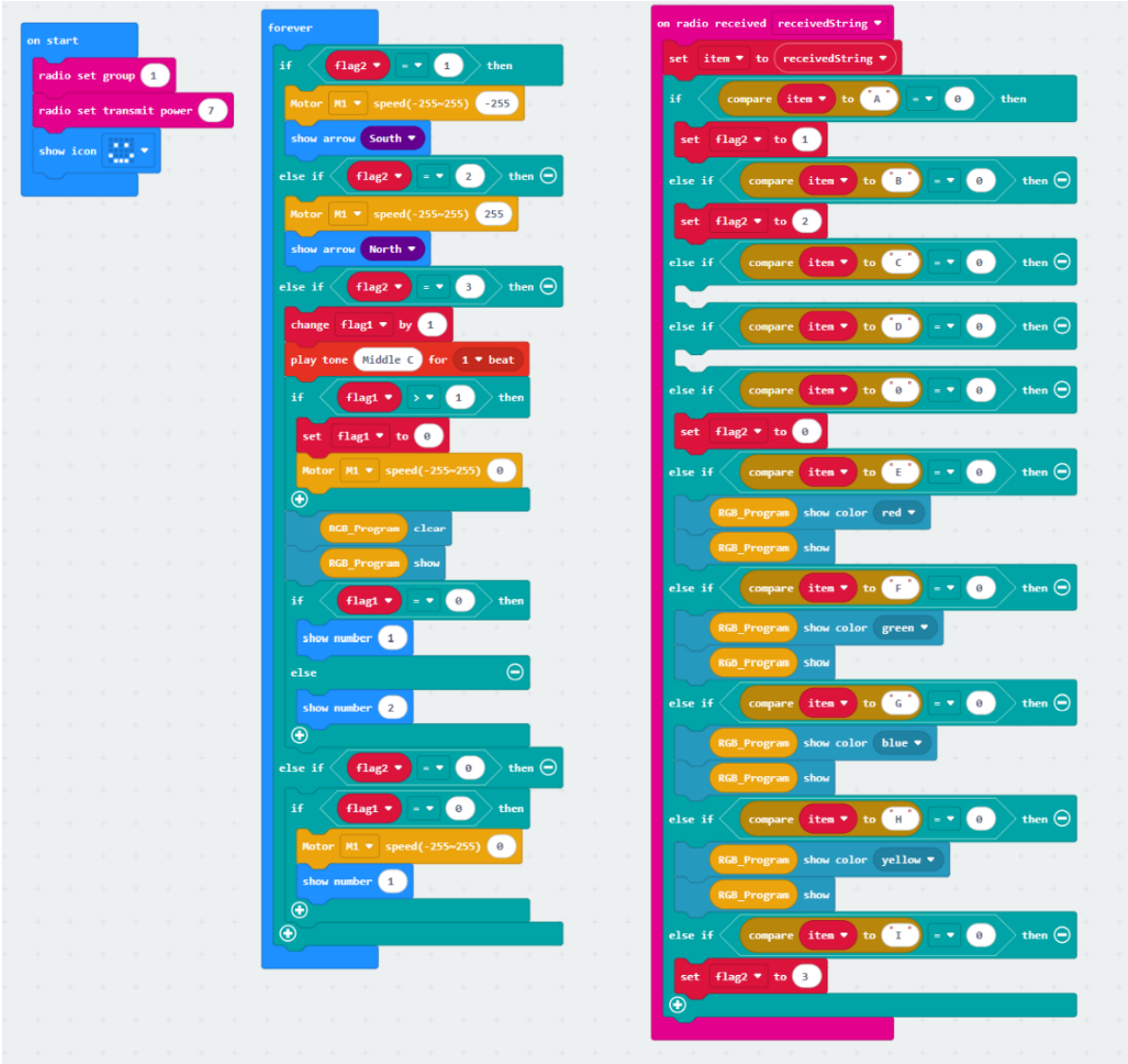


② Enter the variable name to complete the new variable.

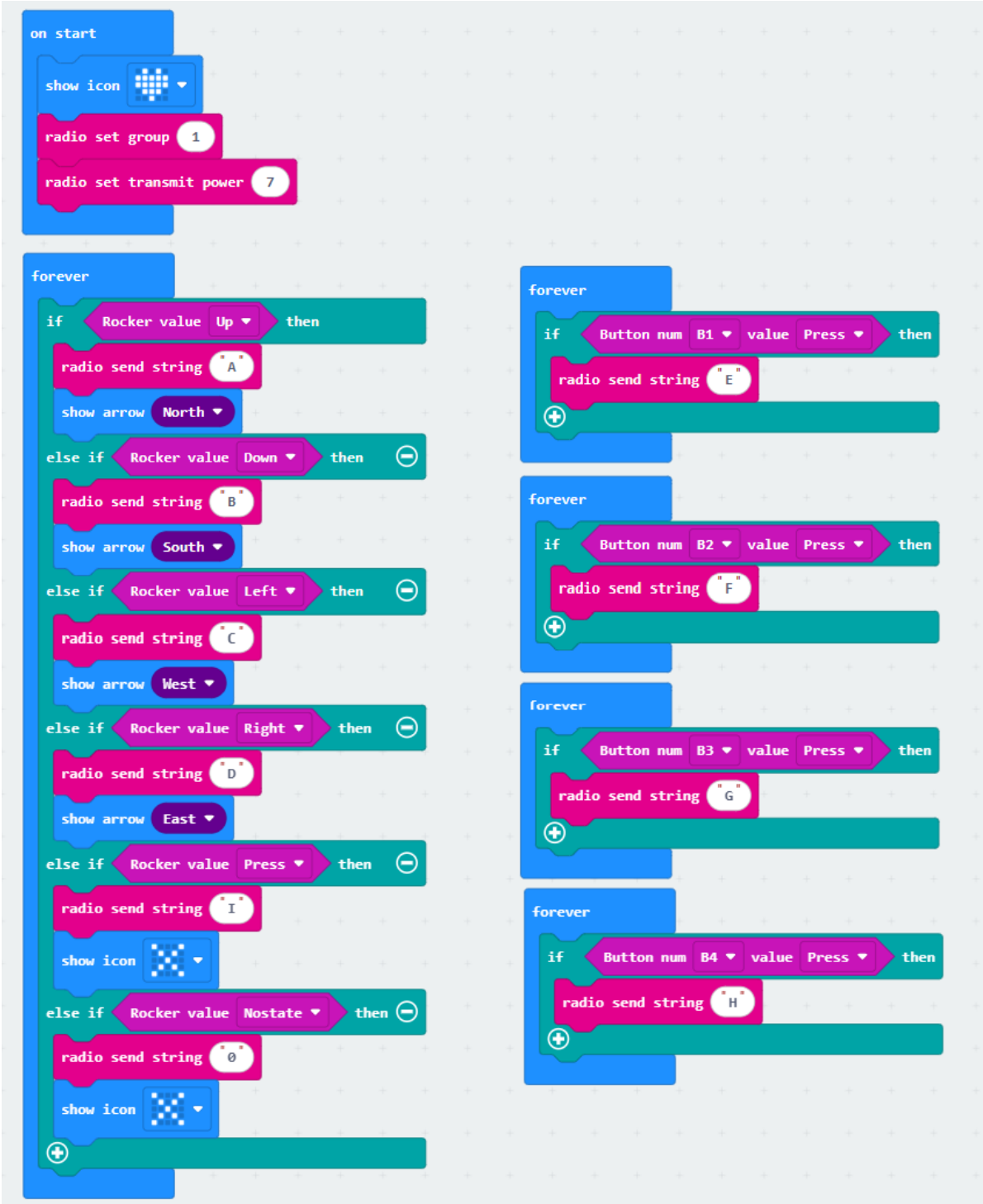


4.4 Combined blocks

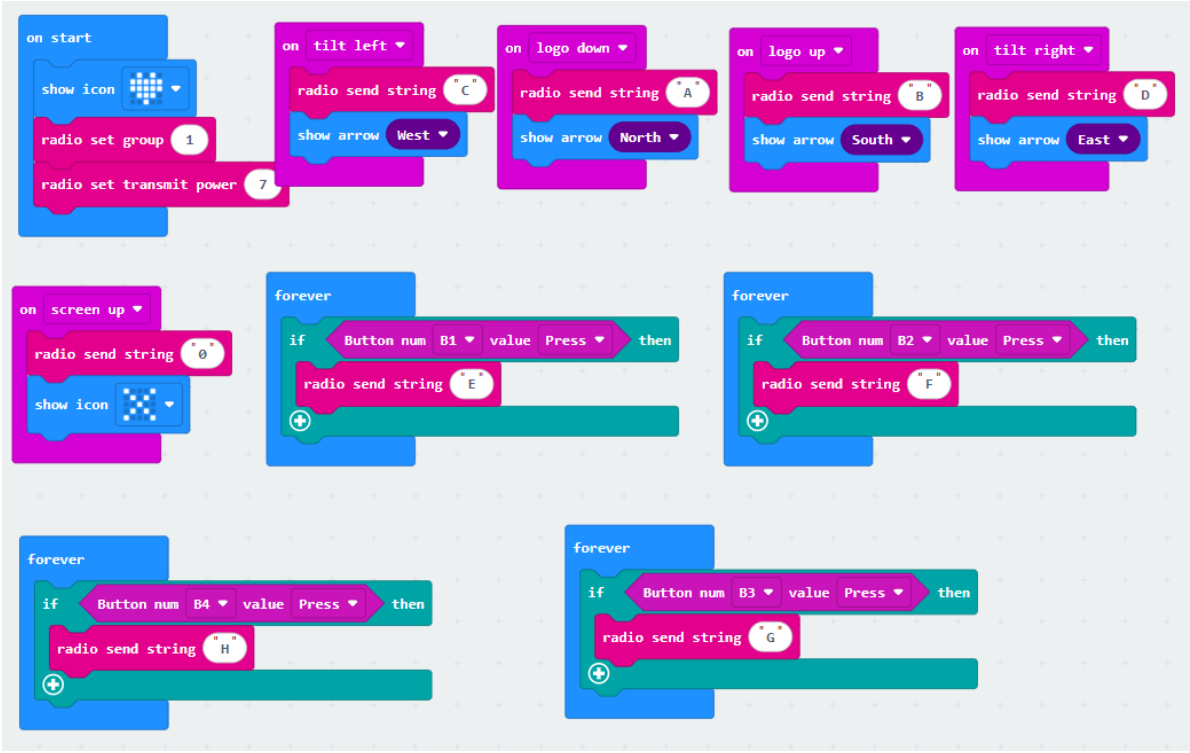
The microbit summary program on the **Biped robot** is shown in the figure below



The **Handle rocker control** program is as follows



The **Handle gravity control** program is as follows



You can also directly open the program provided in this experiment **microbit-handle-control-Biped-robot.hex**、**microbit-Handle-rocker-control.hex**、**microbit-Handle-gravity-control.hex** Drag the file into the browser with the URL opened, and the program diagram of the source code of this project will be automatically opened

5. Experimental phenomenon

We need to download the program of Biped robot to the micro:bit mainboard of Biped robot, turn on the power switch of Biped robot, and we can see that a heart pattern will be initialized on the micro:bit dot matrix.

Download the remote control program of the handle to the micro:bit mainboard of the handle, turn on the power switch of the handle, and we can see that a heart pattern will be initialized on the micro:bit dot matrix, and then an "X" pattern will be displayed, indicating that the handle is in the default state and no data is sent.

The two will automatically complete the pairing, and then we can start to remotely control the Biped robot.

The functions of the handle are as follows.

Handle rocker control:

After the joystick is successfully paired with the Biped robot, we can see the number 1 displayed on the micro:bit dot matrix of the Biped robot, indicating that it is in mode 1.

In mode 1:

- Push the joystick forward to control the Biped robot to move forward, and it will stop when you release your hand;
- Push the joystick backward to control the Biped robot to move backward, and it will stop when you release your hand;
- Press the red button to turn on the red RGB light;
- Press the green button to turn on the green RGB light;
- Press the yellow button to turn on the yellow RGB light;
- Press the blue button to turn on the blue RGB light.

We can press the joystick to switch to mode 2, and we can see the number 2 displayed on the micro:bit dot matrix of the Biped robot, indicating that it is in mode 2.

In mode 2:

- Push the joystick forward to control the Biped robot to move forward, and it will keep moving forward when you release your hand;
- Push the joystick backward to control the Biped robot to move backward, and it will keep moving backward when you release your hand;
- Press the red button to light up the red RGB light and stop the moving Biped robot;
- Press the green button to light up the green RGB light;
- Press the yellow button to light up the yellow RGB light;
- Press the blue button to light up the blue RGB light.

Each time you press the joystick, it will switch back and forth between mode 1 and mode 2, and the RGB light will go out.

Handle gravity control:

After the handle and Biped robot are successfully paired, we can see the number 1 displayed on the micro:bit dot matrix of the Biped robot.

- Tilt the handle forward to control the Biped robot to move forward, and stop when placed horizontally;
- Tilt the handle backward to control the Biped robot to move forward, and stop when placed horizontally;
- Press the red button to light up the red RGB light;
- Press the green button to light up the green RGB light;
- Press the yellow button to light up the yellow RGB light;
- Press the blue button to light up the blue RGB light.

(Note: There is only one mode for handle gravity control, and you cannot switch to mode 2)