Temperature control fan

1. Learning Objectives

In this course, we mainly learn how to make a temperature control fan through MakeCode graphical programming.

2. Building Blocks

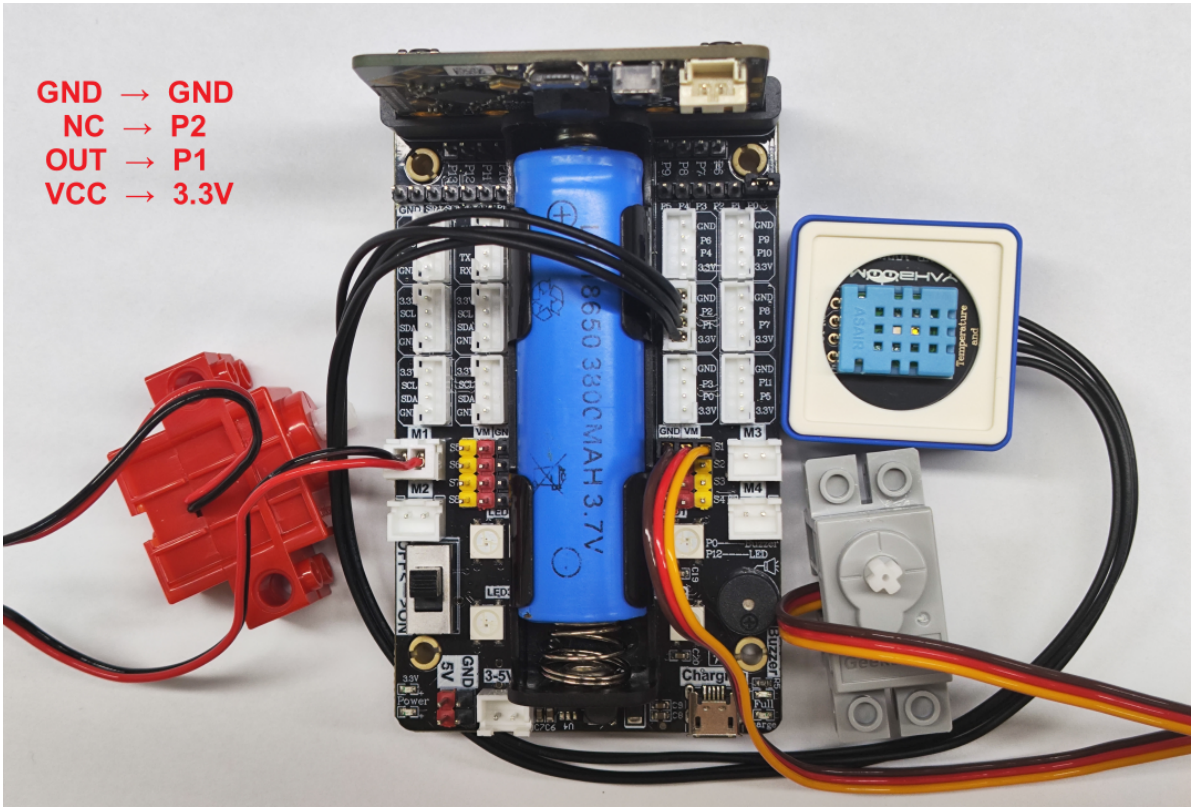
For the steps of building blocks, please refer to the installation drawings of [Assembly Course]--[Temperature-control fan] in the materials or the building block installation brochure.

3. Sensor Wiring

Insert the building block motor wiring into the M1 interface of the Super:bit expansion board, and the black wiring into the side close to the battery.

Insert the building block servo wiring into the S1 interface of the Super:bit expansion board, and the orange servo wiring into the yellow pin of S1.

Connect the temperature and humidity sensor to the P1P2 interface.



4. Programming

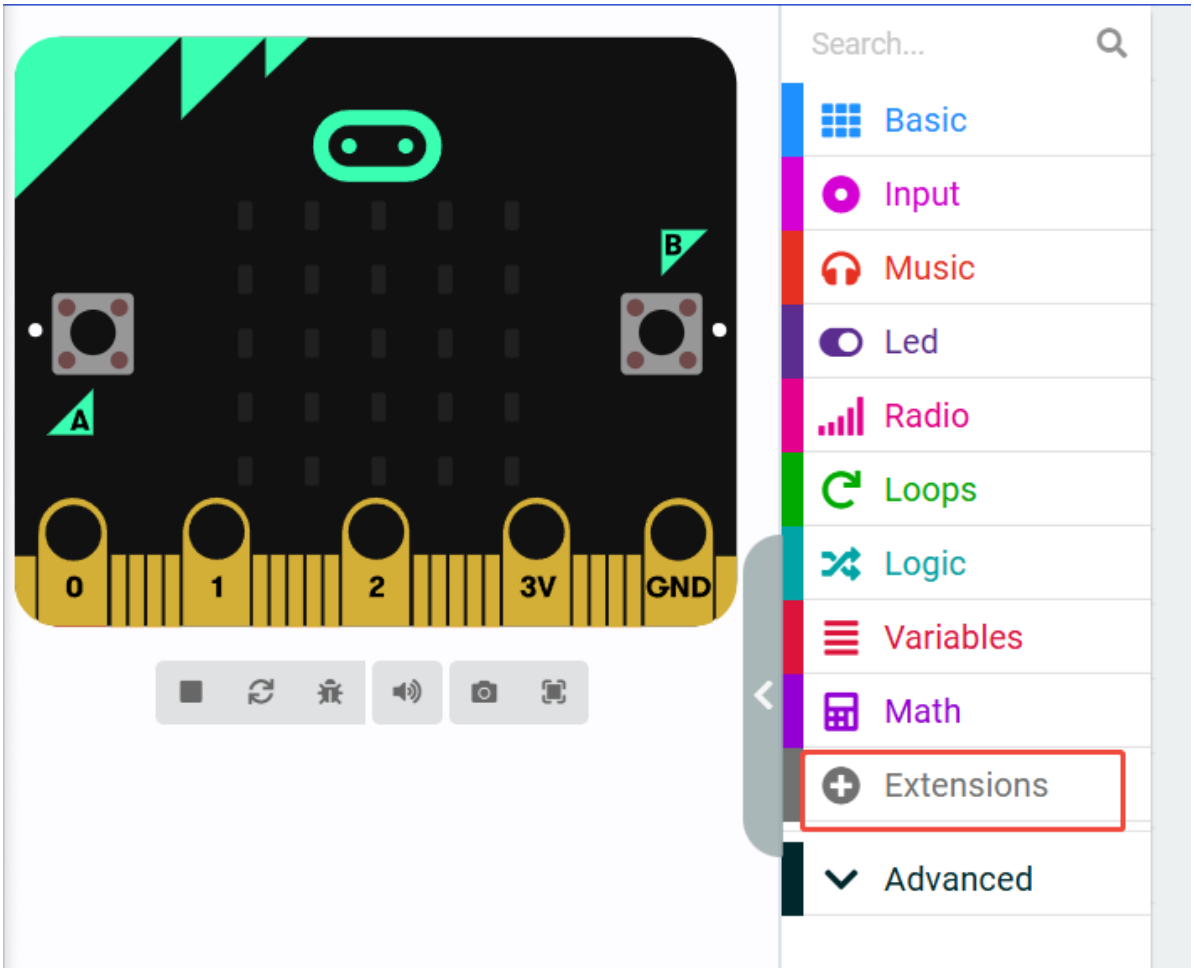
Method 1 Online Programming:

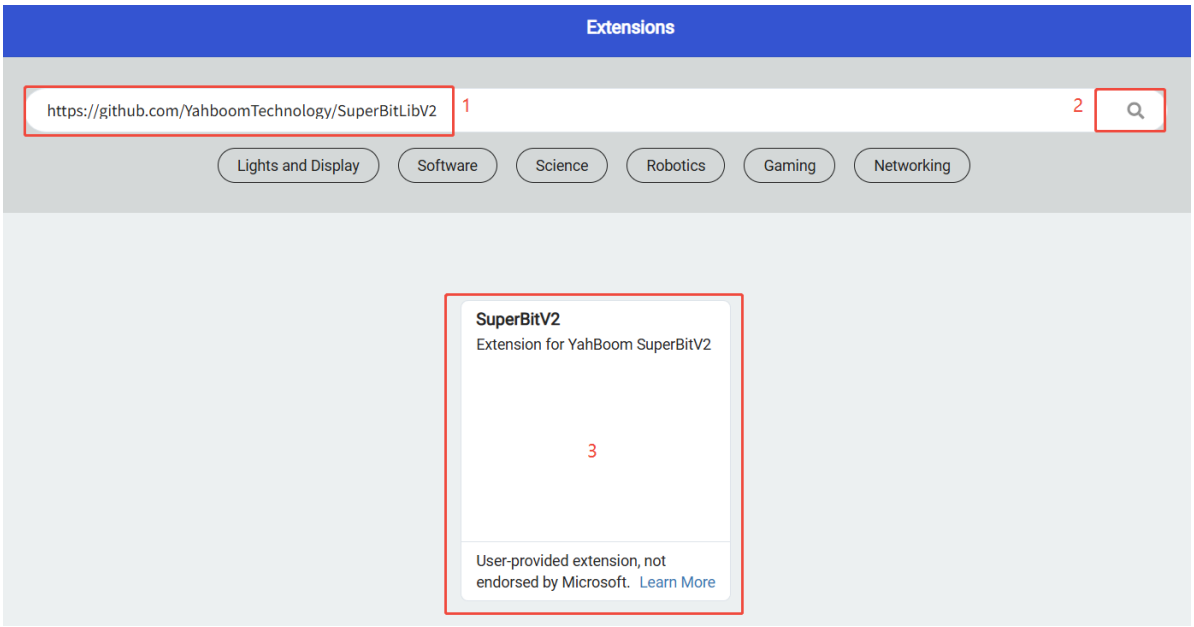
First, connect micro:bit to the computer via a USB data cable. The computer will pop up a U disk. Click the URL in the U disk: https://makecode.microbit.org/ to enter the programming interface. Then, add the Yahboom software package **https://github.com/YahboomTechnology/SuperBitLibV2** to start programming.

Method 2 Offline Programming:

Open the offline programming software MakeCode and enter the programming interface. Click [New] and add the Yahboom software package **https://github.com/YahboomTechnology/SuperBitLibV2** to start programming.

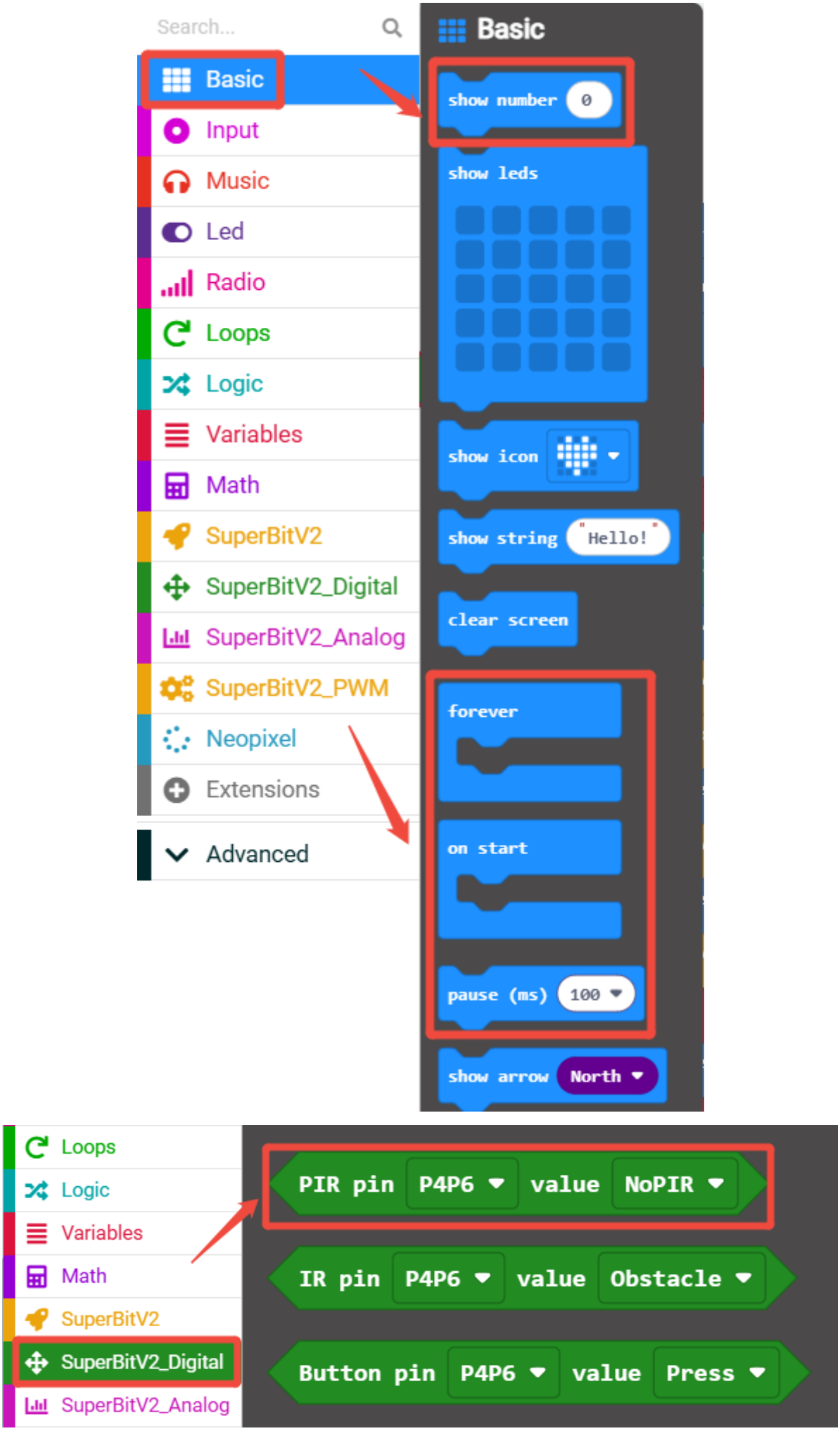
4.1 Adding extension packs



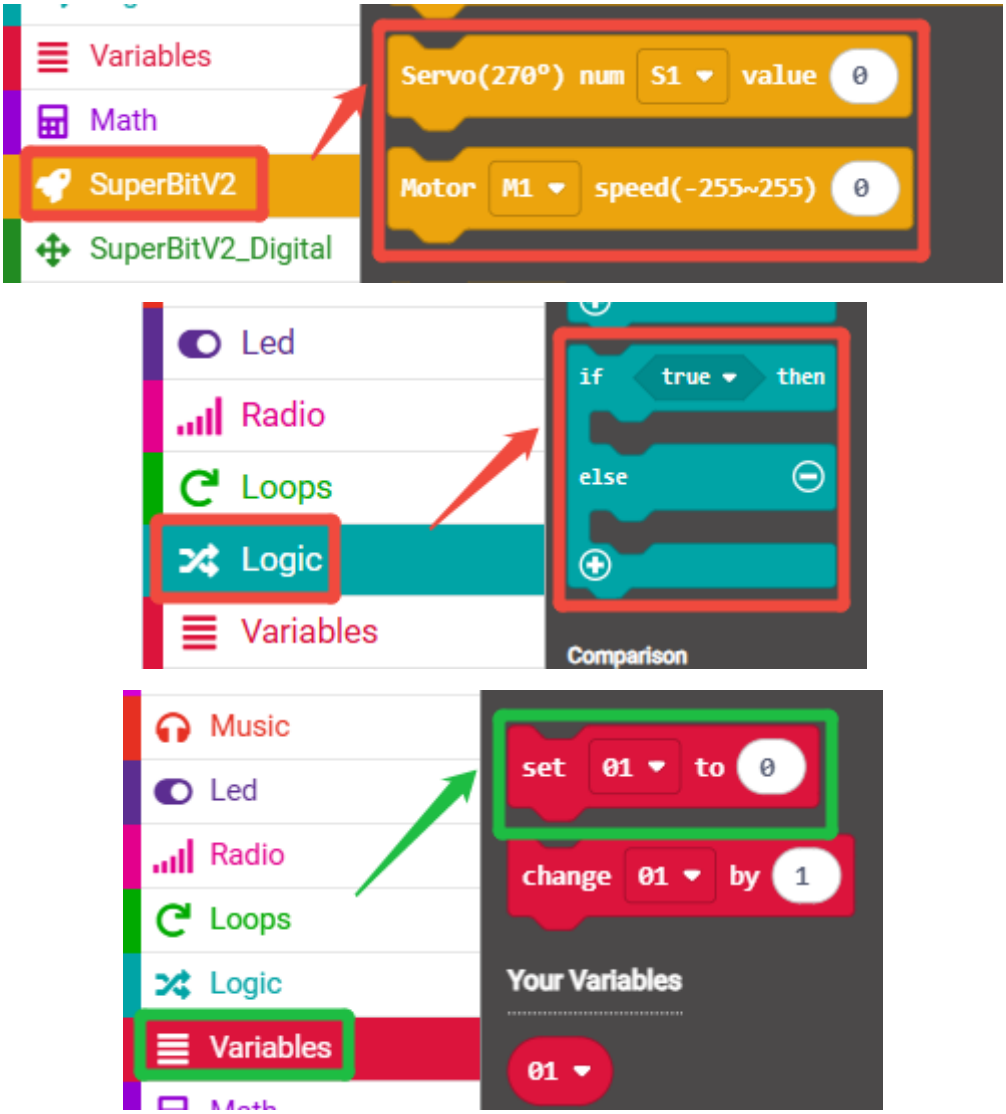


4.2 Building blocks used

The locations of the building blocks required for this programming are shown in the figure below.

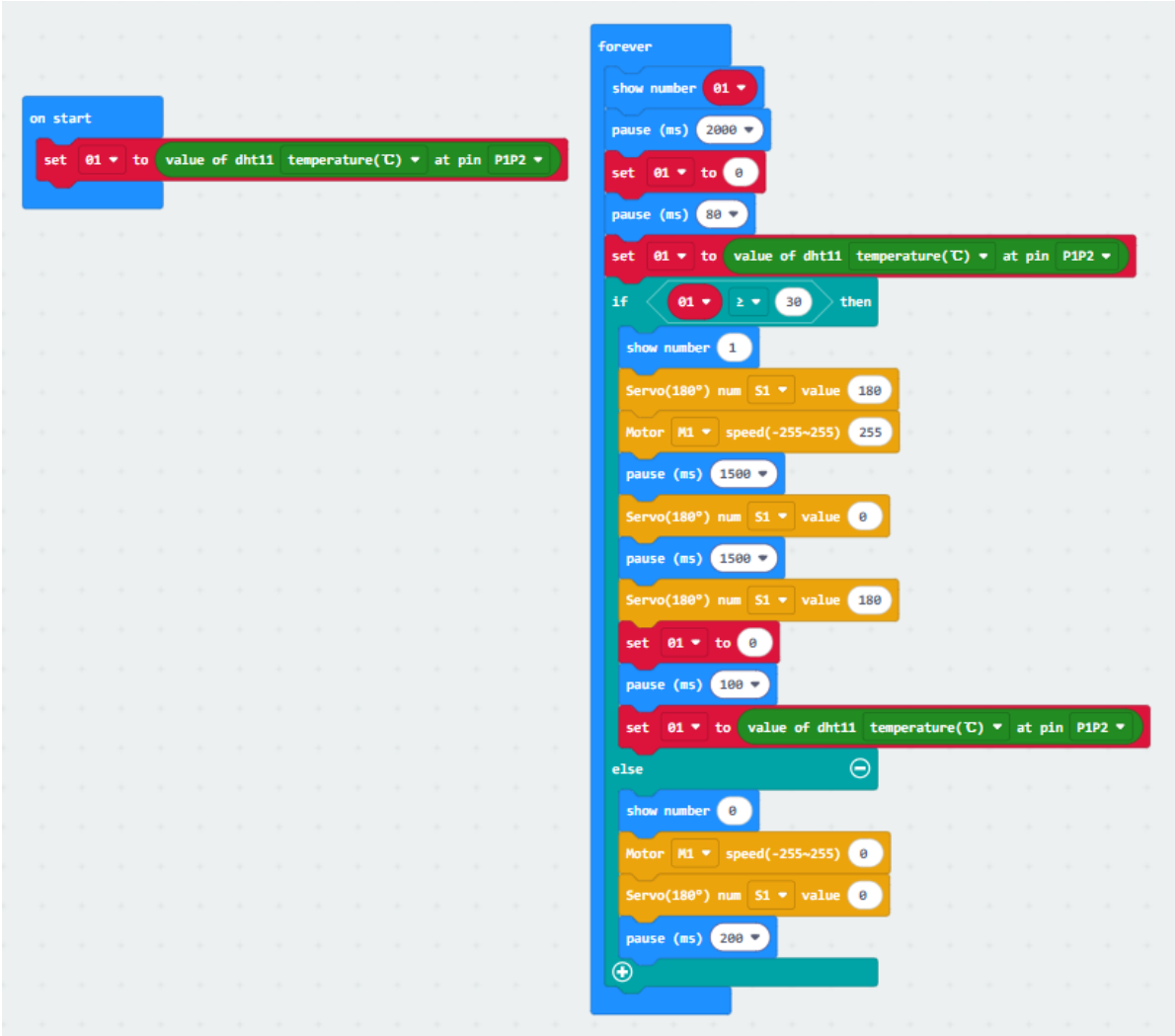

Search...
Basic
Input
Music
Led
Radio
Loops
Logic
Variables
Math
SuperBitV2
SuperBitV2_Digital
SuperBitV2_Analog
SuperBitV2_PWM
Neopixel
Extensions
Advanced
Basic
show number 0
show leds
show icon
show string "Hello!"
clear screen
forever
on start
pause (ms) 100
show arrow North

Loops
Logic
Variables
Math
SuperBitV2
SuperBitV2_Digital
SuperBitV2_Analog
PIR pin P4P6 value NoPIR
IR pin P4P6 value Obstacle
Button pin P4P6 value Press



4.3 Combining blocks

The summary program is shown in the figure below.



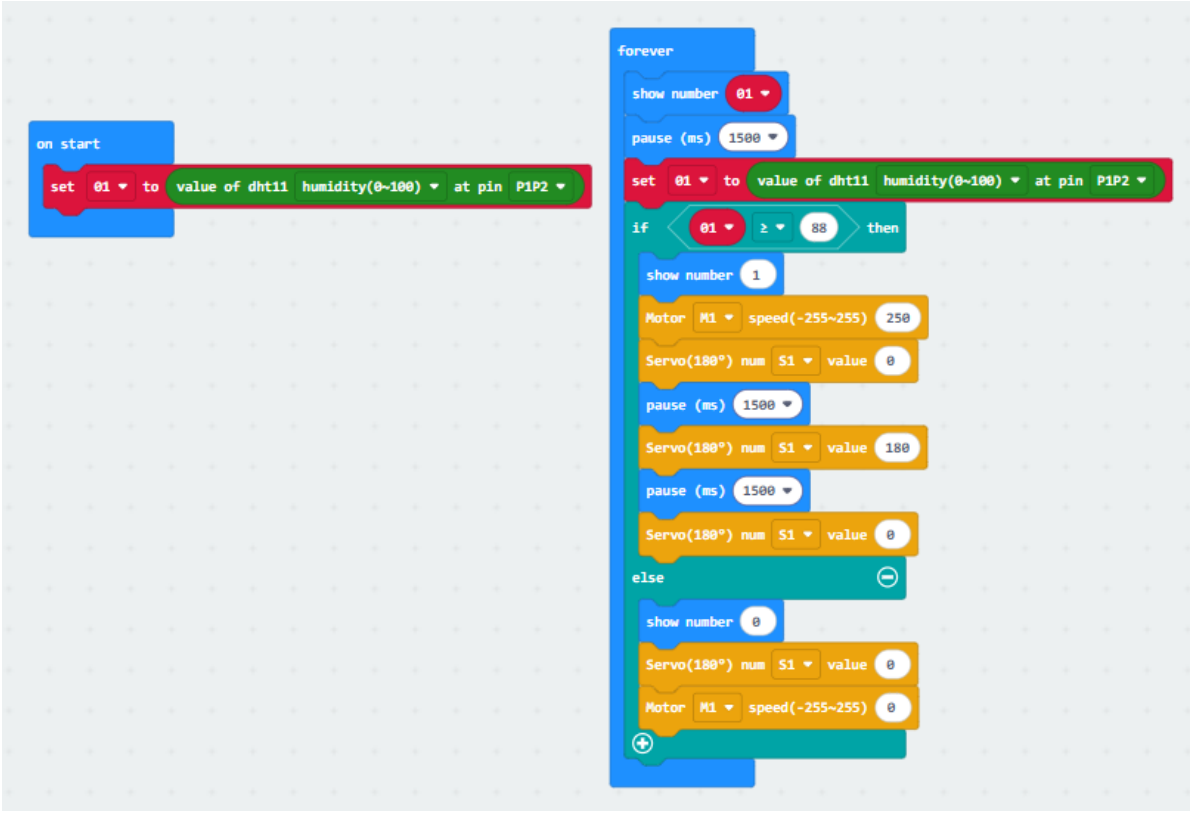
You can also directly open the **Temperature-control-fan.hex** file provided in this experiment and drag it into the browser that opens the URL, and the program diagram of this project source code will be automatically opened.

Since the temperature and humidity module is affected by the temperature and humidity of the environment, the obtained values will also be different. Please modify the sensor threshold according to your actual environment.

5. Experimental phenomenon

After the program runs successfully, the current ambient temperature will be printed on the microbit motherboard. If you hold the temperature and humidity module in your hand to warm it up, after a period of time, when the temperature reaches 30 degrees Celsius, the motherboard dot matrix displays 1, the fan starts to rotate, and the servo turns from 180 degrees to 0 degrees and then returns to 180 degrees. When the temperature is lower than 30 degrees, the fan stops rotating, the servo returns to 0 degrees, and the motherboard dot matrix displays 0.

Note: Temperature detection will have a slow heating and cooling process, and the effect is slightly lower than the real-time performance of humidity detection. If humidity control is required, you can use the case source code provided by the network disk under the same wiring conditions, and continue to breathe on the temperature and humidity module for a few seconds to meet the triggering conditions. The code summary of humidity detection is as follows:


on start
set 01 to value of dht11 humidity(0~100) at pin P1P2
forever
show number 01
pause (ms) 1500
set 01 to value of dht11 humidity(0~100) at pin P1P2
if 01 ≥ 88 then
show number 1
Motor M1 speed(-255~255) 250
Servo(180°) num S1 value 0
pause (ms) 1500
Servo(180°) num S1 value 180
pause (ms) 1500
Servo(180°) num S1 value 0
else
show number 0
Servo(180°) num S1 value 0
Motor M1 speed(-255~255) 0