# Table lamp

**Principle of button:** The reset button is used. Press the button, and after releasing the hand, the button cap will spring up and restore to its original position. It is connected to the circuit as a digital input, and can read two states. When pressed or not, the corresponding values are 0 and 1.



All sample programs of the course are in the folder as shown below.

|  |
| --- |
|  |

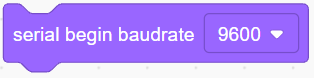
## Assemble the table lamp

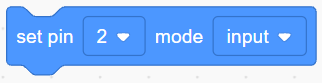
|  |  |
| --- | --- |
| Remove the red parts in the figure below  拆呼吸灯红色部分 | 拆完后 |
| 小台灯_空白视图 4 | 小台灯_空白视图 4_1 |
| 小台灯_空白视图 4_2 | 小台灯_空白视图 4_3 |
| 小台灯_空白视图 4_4 | 小台灯_空白视图 4_5 |
| 小台灯_空白视图 4_6 | 小台灯_空白视图 4_7 |
| 小台灯_空白视图 4_8 | 小台灯_空白视图 4_9 |
| 小台灯_空白视图 4_10 | 小台灯_空白视图 4_11 |
| 小台灯_空白视图 4_12 | 小台灯_空白视图 4_13 |
| 小台灯_空白视图 4_14 | 小台灯_空白视图 4_15 |
| 小台灯_空白视图 4_16 |  |
| The left button is connected to D2, the right button is connected is connected to D4, and the LED light is connected to D11. | |

## Read the status values of the buttons

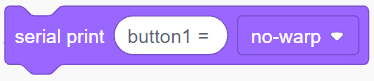
Read the status values when the two buttons are pressed and released.

1. **Program block**

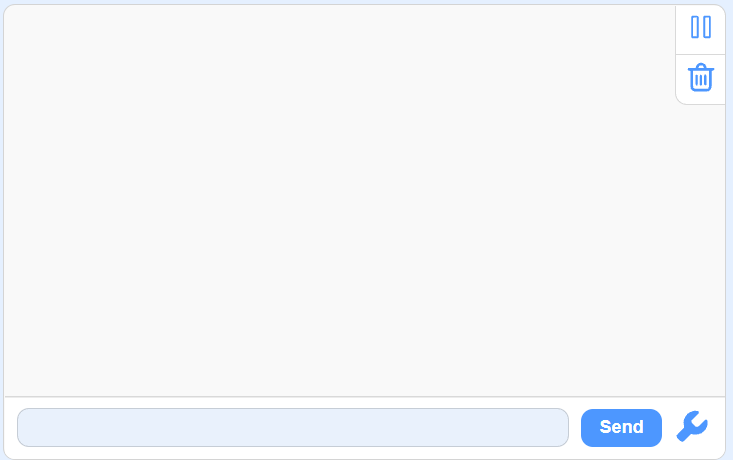
Set the serial baud rate for serial communication.

Set the pin as input mode, and the buttons are sensors for input.

Read the digital value of the pin, and the value obtained can only be 0 or 1.

The serial port prints the values with no line break at the end of the printing.

The serial port monitor of SnailBlock graphical software is in the lower right corner.

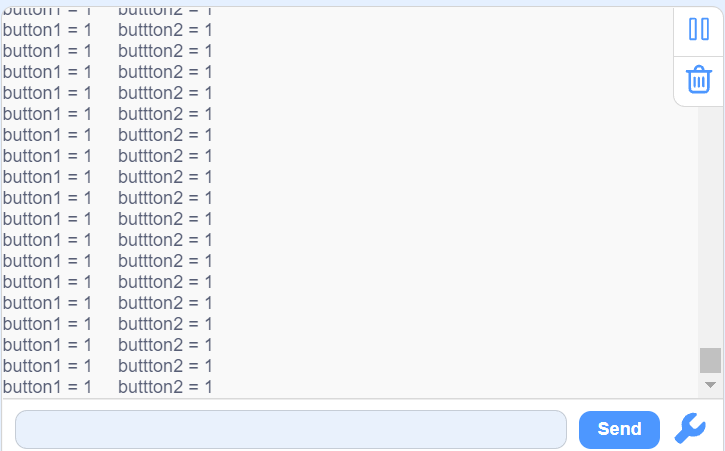


1. **Example code**

|  |
| --- |
|  |

**（2）Experiment phenomenon**

You can see the printed values in the serial port monitor area of SnailBlock. Press the button to observe whether the printed value changes to 0.



**Tips：**Modify the string printed in the Serial. Print() function and observe whether the modified string will be printed. The serial port monitor is a vary useful tool. It can visually display the printed values, and can be used to check where the code stops, and so on. Be good at using the serial port.

## Buttons control the LED

Two buttons, one controls LED light on, the other controls LED light off.

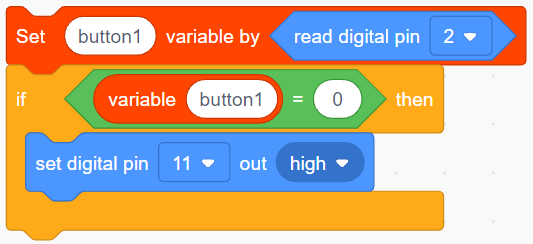
1. **Program block**

Judge the program block. If the condition is correct, execute the program block inside.

**（2）Example code**

|  |
| --- |
|  |

**（3）Program explanation**



Read the value of pin 2 (the button module is connected to pin 2) and assign it to the variable button 1.

Judge whether the variable button 1 is equal to 0. If it is equal to 0, set the LED on.

**（4）Experiment phenomenon**

Press the button on the left to turn on the LED, and then press the button on the right to turn off the LED.

**Tips：**Try to modify the code by yourself. Instead, click button 2 to turn on the LED, and then click button 1 to turn off the LED.

## Calculate the number of button clicks

To learn the table lamp function, the key step is to learn the button counting function firstly, which is to calculate the number of times the button is clicked. Click is to press and release the button once. There is problem that the buttons are jittery, which means that the buttons are pressed but not detected. We can reduce the error caused by the button jitter by judging that the buttons are pressed twice.

1. **Program block and idea**

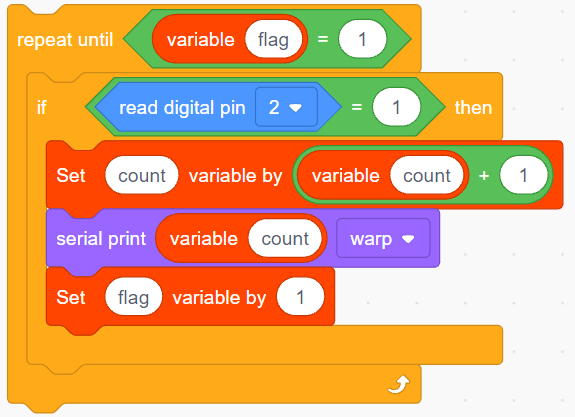
The program inside is repeated until its judgement condition is correct.

Programming idea: How can I count the times a button is clicked? After the button is pressed and then released, the button is clicked once.

（2）Example code

|  |
| --- |
|  |

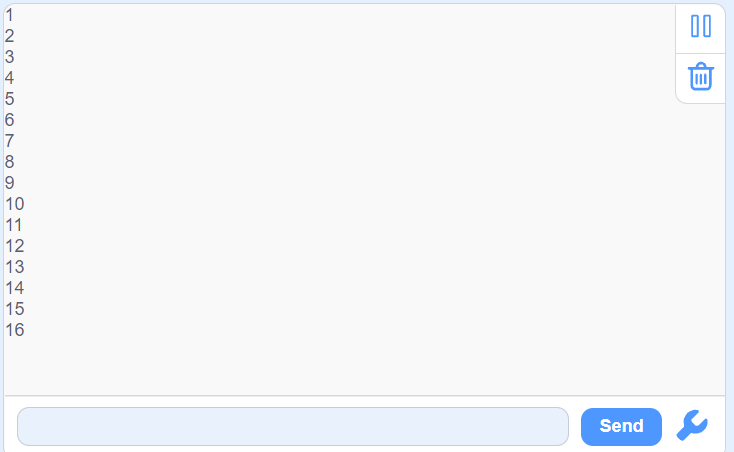
（3）Program explanation



The program is executed repeatedly. When the button is released, that is, pin 2 = 1, variable count = count + 1, and then set variable flag = 1 to meet the judgement condition of repeated execution block flag = 1, and stop repeating the program.

（4）Experiment operation and phenomenon

Click the button on the left, and the serial port monitor area prints the number of times the button has been clicked.



## Table lamp

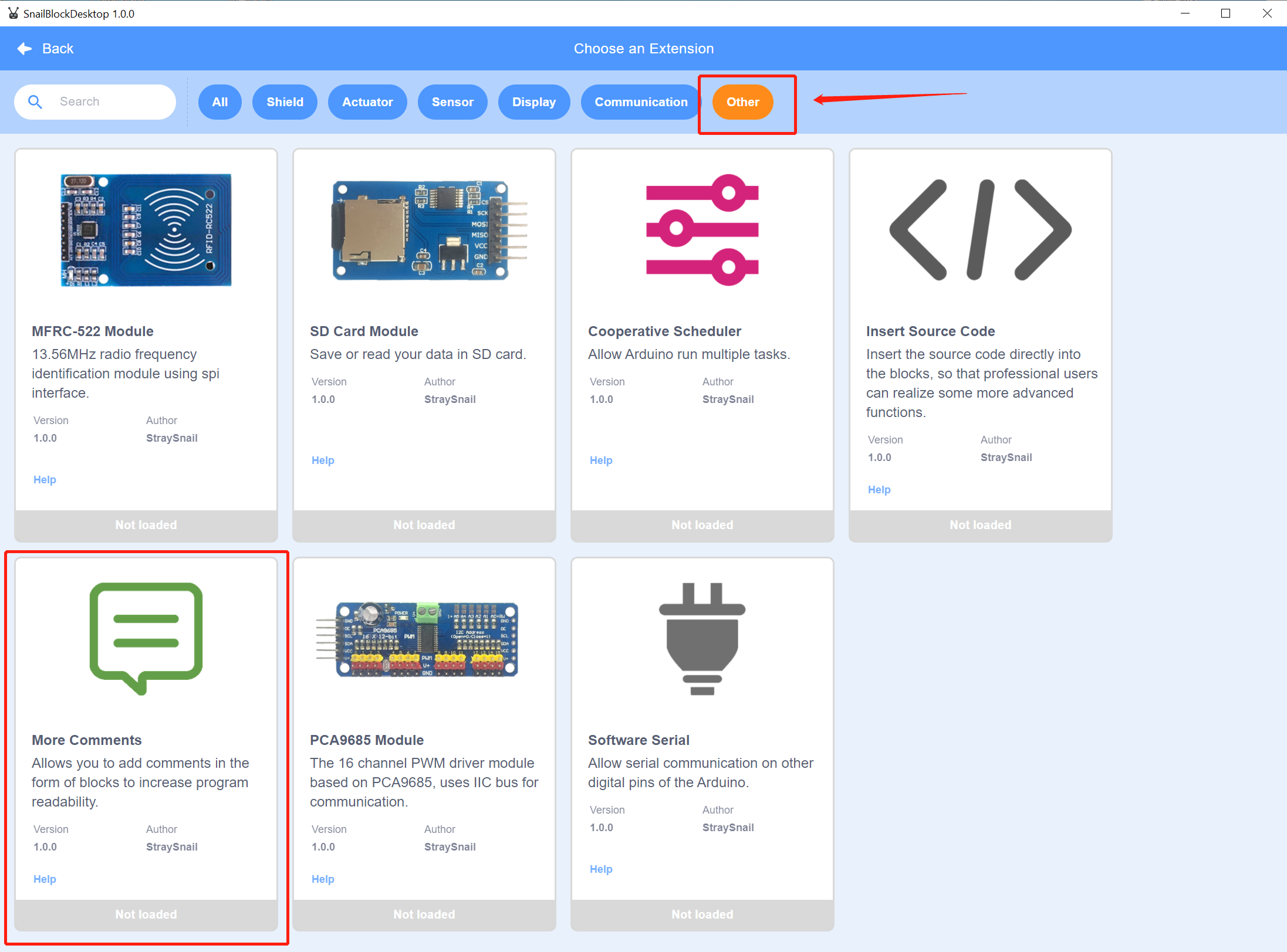
Many simple table lamps will light up when you press the button, and turn off when you press the button again.



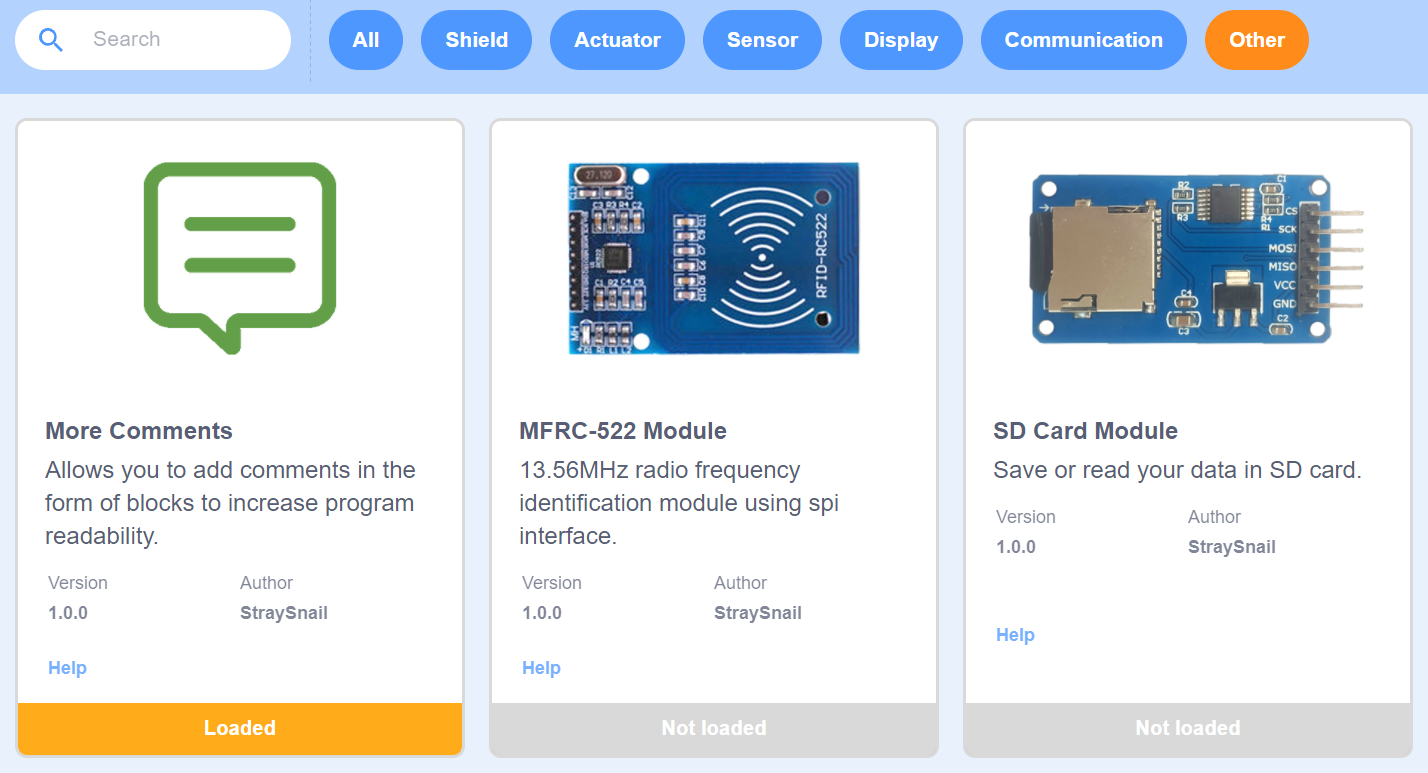
1. **Program block**

The program comment block is in the plug-in, so you need to add a plug-in.

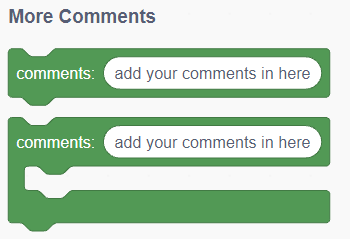
Click the bottom left corner of SnailBlock. Open the plug-in interface, click “Other”, and then click “More comment plugins”.



Clicked to display “Loaded”.



After clicking “Back”, you can see more comment program blocks added.



The comment blocks are for the convenience of interpreting the program. Anything written in the comment block will not affect the normal operation of the program.

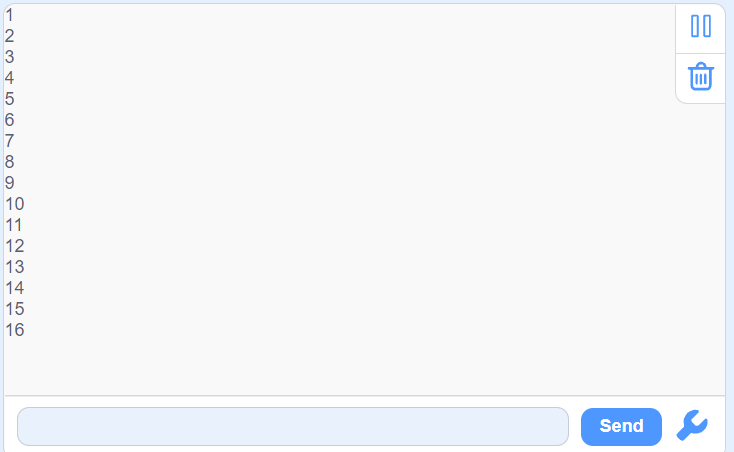
The operation block, finding the remainder, ts to find the remainder in mathematics. Usually, we use to find the remainder of 2, and the resulting value is 0 in the double and 1 in the singular.

1. Example code

|  |
| --- |
|  |

（2）Experiment phenomenon

Press the button on the left to turn on the LED. Press the button again to turn off the LED. Open the serial port monitor of Arduino IDE, click the button, and you can see that the serial port has printed the number of times the button was clicked.



**Tips:** In the example code, you can get 0 or 1 by calculating the number of times the button is pressed, and then calculate the remainder of 2. This method is vary common. Try to write it several times to get familiar with this usage.