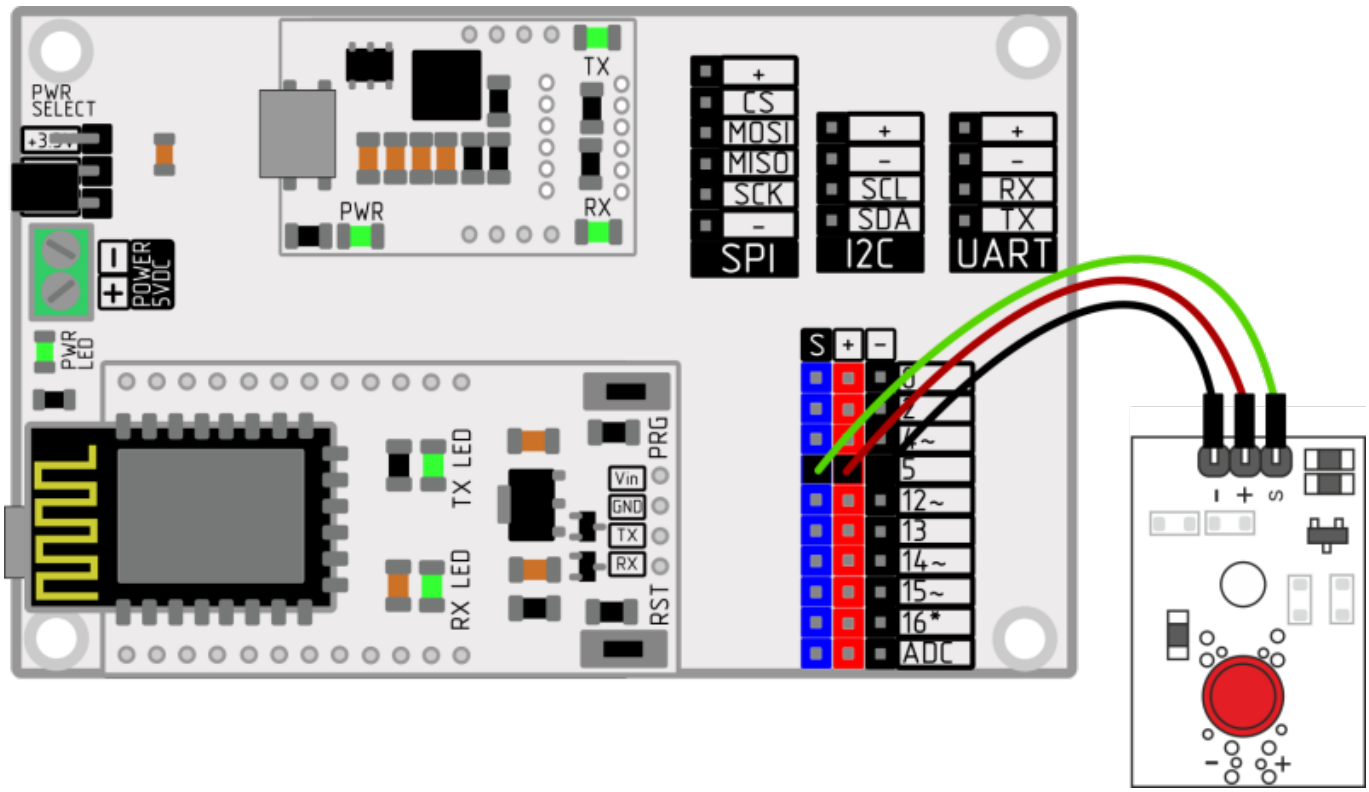
Урок 8. Циклы. Цикл While

В этом уроке мы будем использовать ту же схему подключения что и в прошлом уроке:



Несмотря на то что код из нашего предыдущего урока выглядит уже более опрятно, в нём очевиден ещё один изъян - это повторение строки blink несколько раз. Именно в этот момент вызывается наша функция. Но ведь сейчас нам нужно моргнуть только три раза, а потом мы хотим чтобы светодиод постоянно мигал. Что же делать тогда? Писать много раз подряд строку `blink()`?

Для случаев когда код нужно повторять несколько раз существуют специальные синтаксические конструкторы, они называются циклы. Цикл повторяет выполнение участка кода. В этом уроке мы познакомимся с циклом `while` (*от английского while - пока*).

Конструкция цикла `while` следующая: после ключевого слова `while` указывается выражение, которое будет проверяться перед каждым повторением. Если выражение истинно (оно равно `True`), то цикл выполняется. Если оно ложно (равно `False`), цикл прекращается.

Условные выражения записываются так же как и в математике, за несколькими исключениями. Они приведены в таблице:

Выражение	**Проверяемое условие**	**Пояснение**
`x > y`	Больше	Проверяет x больше чем y? Например, выражение `2 > 3` вернёт `False`, а выражение `3 > 0` вернёт `True`.
`x >= y`	Больше либо равно	Проверяет x больше или равно чем y? Например, выражение `2 >= 3` вернёт `False`, а выражение `3 >= 3` вернёт `True`.
`x < y`	Меньше	Проверяет x меньше чем y? Например, выражение `2 < 3` вернёт `True`, а выражение `3 < 0` вернёт `False`.
`x <= y`	Меньше либо равно	Проверяет x меньше или равно чем y? Например, выражение `2 <= 3` вернёт `True`, а выражение `3 <= 0` вернёт `False`.
`x == y`	Равно	Проверяет x равен ли y? Например, выражение `2 == 2` вернёт `True`, а выражение `3 == 2` вернёт `False`.
`x != y`	Не равно	Проверяет x не равен ли y? Например, выражение `2 != 2` вернёт `False`, а выражение `3 != 2` вернёт `True`.

Как и с другими конструкциями, чтобы было понятно что данная строка принадлежит циклу, она должна отделяться пробелами.

Для того чтобы цикл выполнялся постоянно, ему нужно указать что условие всегда верно. Например написать 2=2 (два всегда равно двум). Либо напрямую указать значение `True` (которое как раз и возвращается при проверке условия 2=2). Таким образом наш цикл будет выглядеть следующим образом:

```
while True:
    blink()
```

Здесь мы заменили наши вызовы функции blink на цикл while True. Таким образом цикл будет выполняться постоянно и светодиод будет постоянно мигать. Весь код нашего скрипта получит следующий вид:

```
# импорт модулей
from machine import Pin
import time

#переменные
led = Pin(5, Pin.OUT)
sec = 1

# мигание светодиодом
```

```
def blink():
    led.on()
    time.sleep(sec)
    led.off()
    time.sleep(sec)


# основной цикл программы
while True:
    blink()
```

Запомнить:

- Циклы нужны для повторения участков кода
- Перед запуском цикла while проверяется условие, если оно истинно, то цикл выполняется, если ложно, то цикл не выполнится
- Условные выражения записываются так же как и в математике

Предыдущий урок

Следующий урок