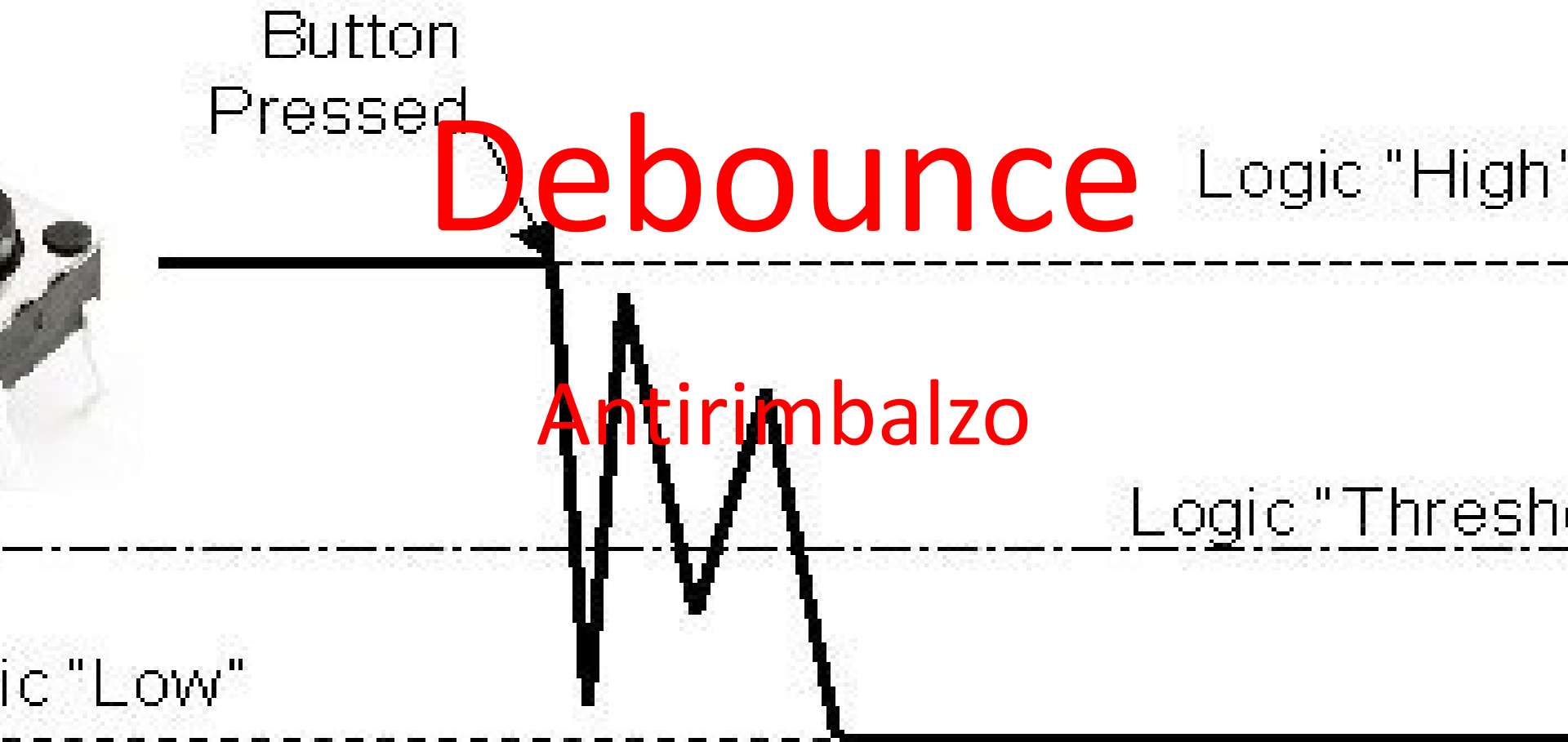


# Button "Bounce"



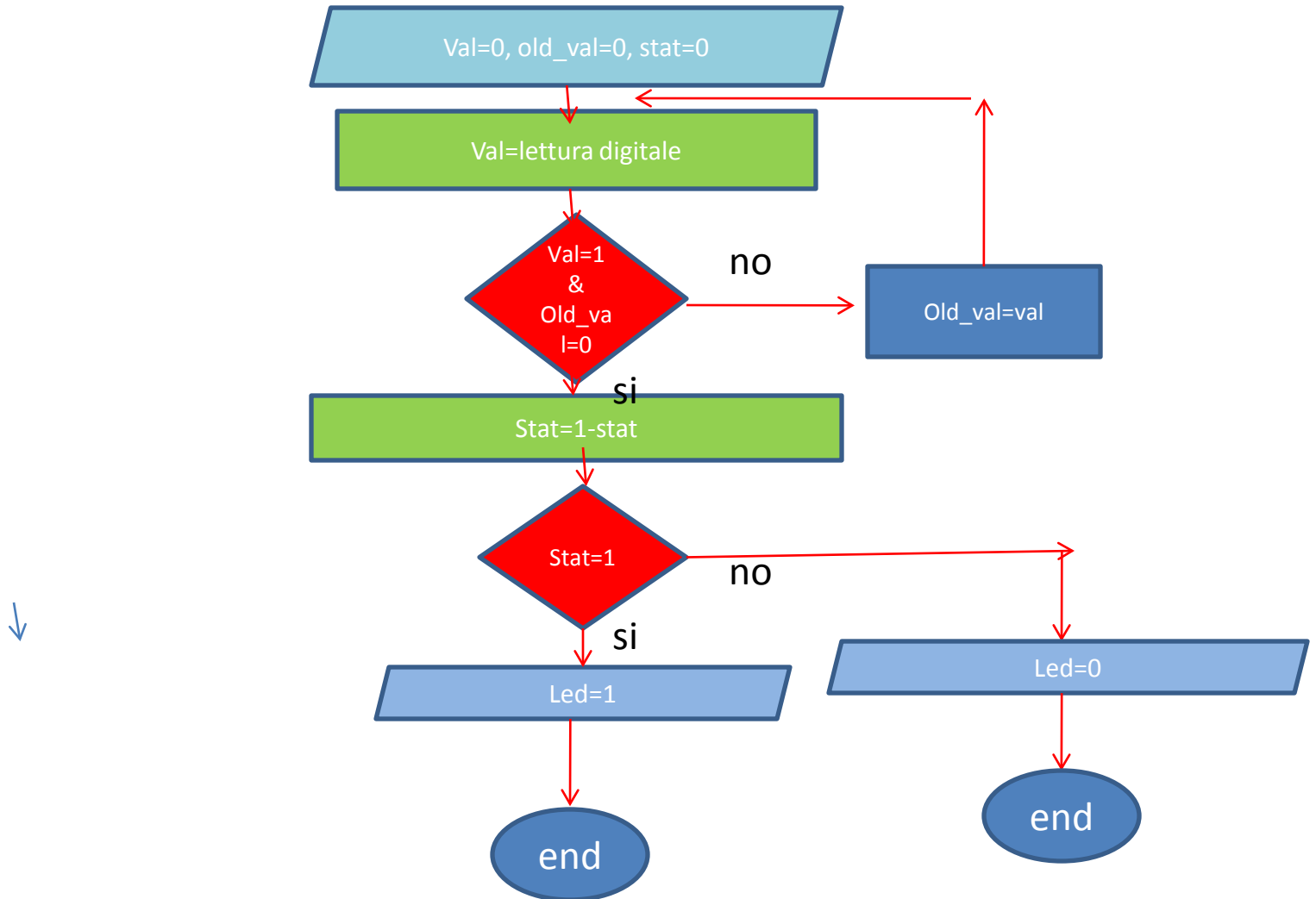
# Antirimbalzo



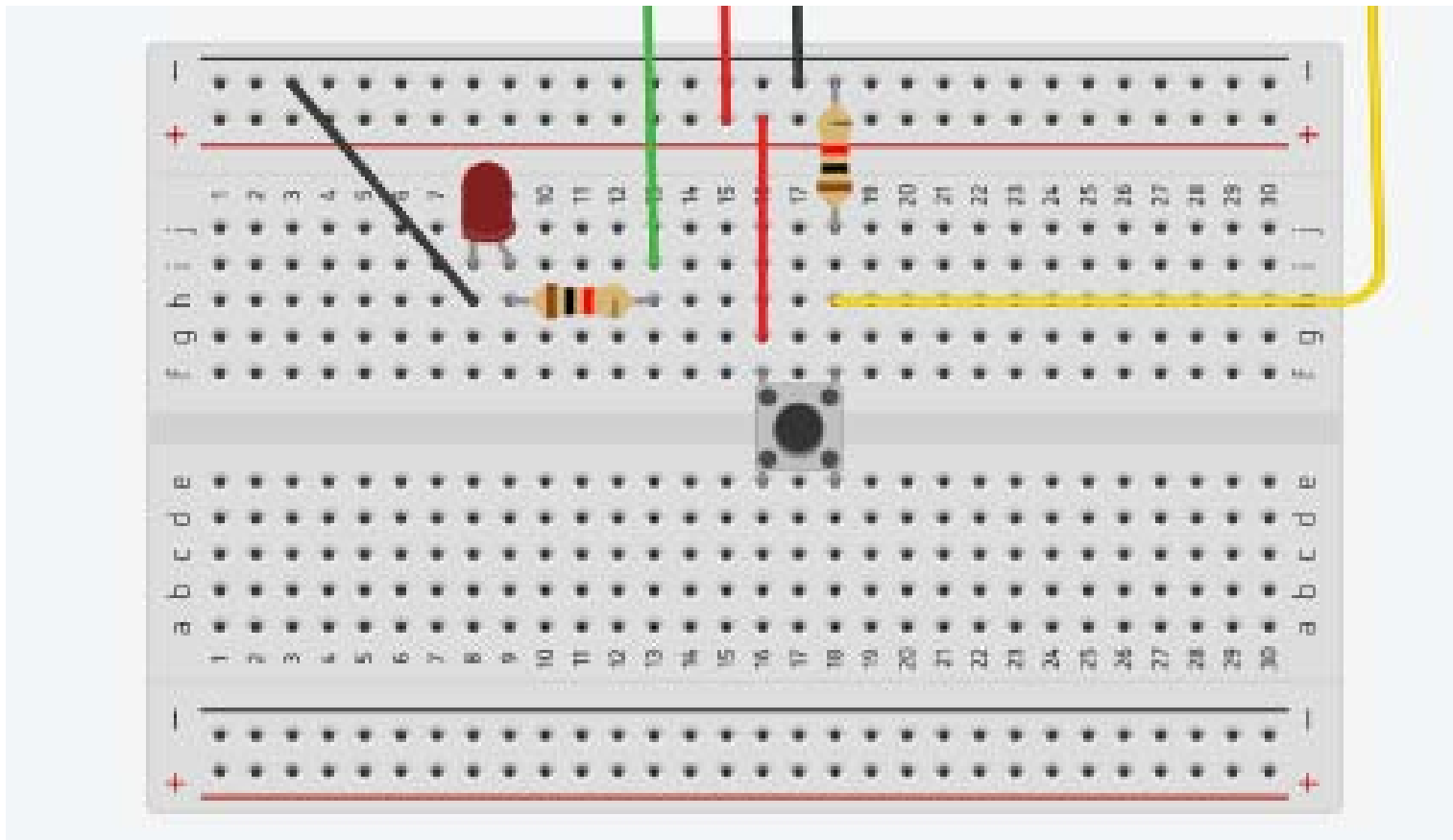
# Bounce, rimbalzo

- Quando viene pigiato un interruttore, si avrà sicuramente un cambio di stato
- Il cambio di stato non è sempre immediato ma possono esserci dei rimbalzi meccanici che creano un segnale impreciso detto flickering
- Molte volte il problema del rimbalzo viene risolto attraverso il sistema hardware
- Nel caso dei microcontrollori il problema viene risolto tramite software.

# Flow chart



# Circuito interruttore-led



# Programma Raspberry

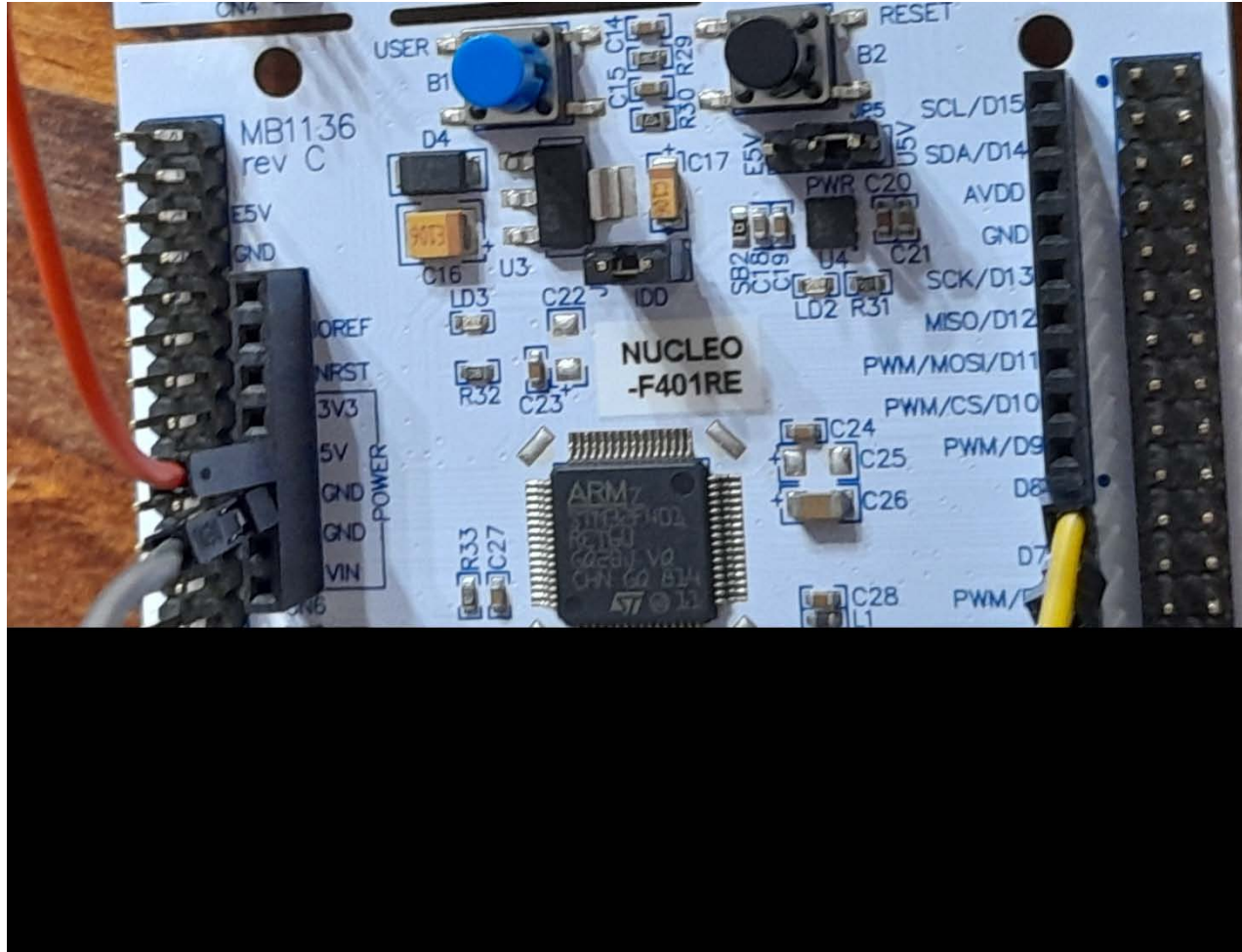
```
import RPi.GPIO as gpio
import time
gpio.setwarnings(False)
gpio.setmode(gpio.BCM) #il nome del pin non è sequenziale ma del #costruttore
gpio.setup(18, gpio.OUT) #pin 18 (nome BCM 12 come sequenza) OUTPUT #(led)
gpio.setup(23,gpio.IN) #pin 21 (nome BCM 23 come sequenza) INPUT
old=0
stat=0
while True:
    val=gpio.input(23)    #inva= variabile di lettura
    if val==1 & old==0:    #se inva =0 oppure false
        stat=1-stat
    old=val
    if val==0:
        gpio.output(18,False) #il led non si accende
    if val==True:         #se inva=1 o true
        gpio.output(18,True)  #il led si accende
```

# Programma Arduino

```
✓ ↩ 📄 ⬆ ⬇
antirimbalzo
#define led 10
#define B 7
int val = 0;
int old_val = 0;
int stat = 0;
void setup() {
  pinMode(led, OUTPUT);
  pinMode(B, INPUT);
}
void loop() {
  val = digitalRead(B);
  if ((val == HIGH) && (old_val == LOW)){
    stat = 1 - stat;
    delay(15);
  }
  old_val = val;
  if (stat == 1) {
    digitalWrite(led, HIGH);
  }
  else {
    digitalWrite(led, LOW);
  }
}
```

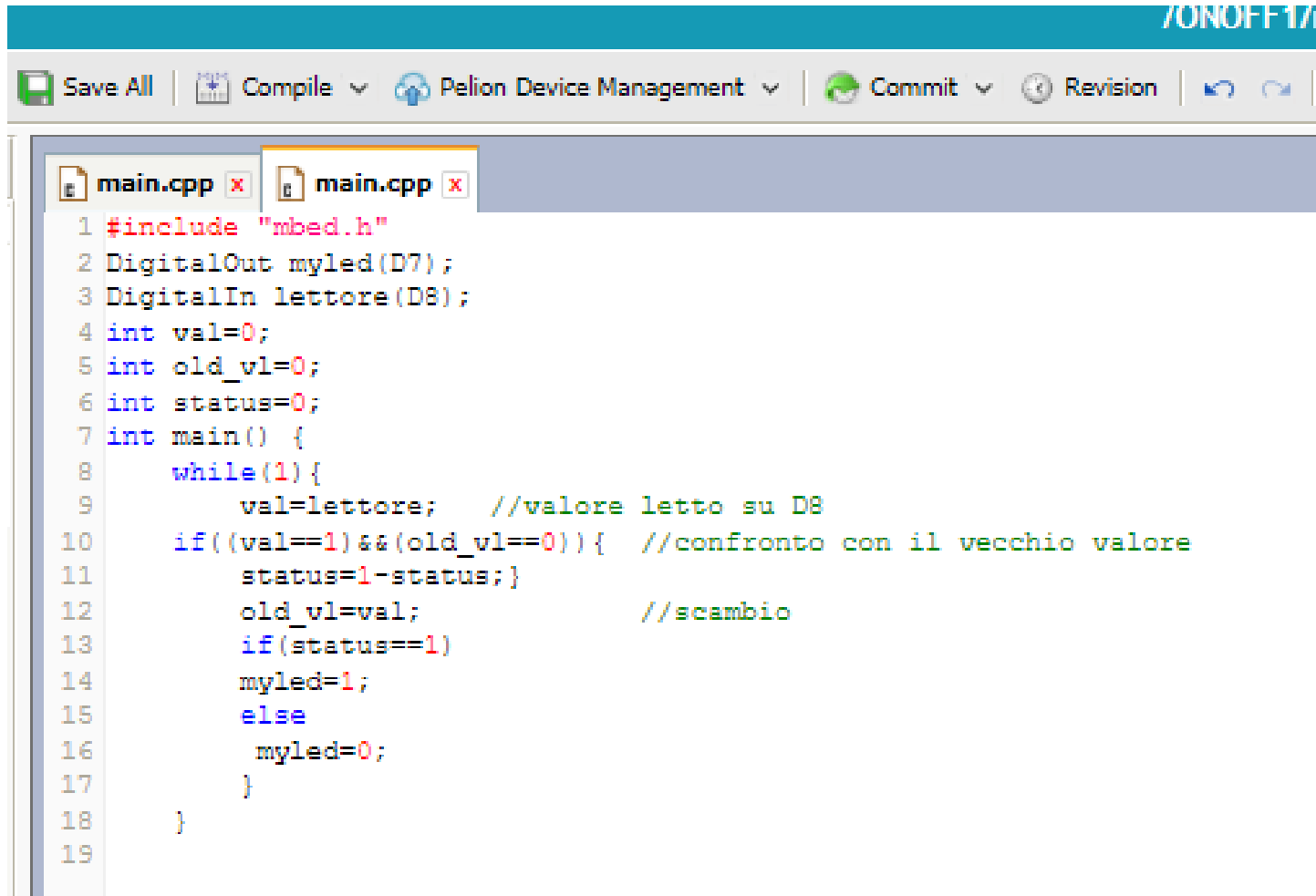
---

# STM32





# Programma STM32



```
1 #include "mbed.h"
2 DigitalOut myled(D7);
3 DigitalIn lettore(D8);
4 int val=0;
5 int old_val=0;
6 int status=0;
7 int main() {
8     while(1){
9         val=lettore; //valore letto su D8
10        if((val==1)&&(old_val==0)){ //confronto con il vecchio valore
11            status=1-status;}
12        old_val=val; //scambio
13        if(status==1)
14            myled=1;
15        else
16            myled=0;
17    }
18 }
19
```