

# To Blink or Not to Blink?

Paul MacDougal

[Triembed.org](http://Triembed.org)

Dec 12, 2016

# Blink

// the setup function runs once when you power the board

```
void setup() {  
  // initialize digital pin 13 as an output.  
  pinMode(13, OUTPUT);  
}
```

// the loop function runs over and over again forever

```
void loop() {  
  digitalWrite(13, HIGH); // turn the LED on (HIGH is the voltage level)  
  delay(1000);           // wait for a second  
  digitalWrite(13, LOW); // turn the LED off (LOW is the voltage level)  
  delay(1000);           // wait for a second  
}
```

- This talk is not about blinking LEDs
- It is about how to write software that plays well with others.
- The first thing we will do is move blink to a separate file, one that can be used in a project with other activities going on.

# Blink and Button

// the setup function runs once when you power the board

```
void setup()
{
  initBlink();
  initButton();
}
```

// the loop function runs over and over again forever

```
void loop() {
  handleBlink();
  handleButton();
}
```

# Button

```
#include <Arduino.h>
#include "pins.h"
```

```
void initButton()
{
    pinMode(BUTTON_PIN, INPUT_PULLUP);
    pinMode(BUZZER_PIN, OUTPUT);
    digitalWrite(BUZZER_PIN, LOW);
}
```

```
void handleButton()
{
    digitalWrite(BUZZER_PIN, HIGH ^ digitalRead(BUTTON_PIN));
}
```

# Blink 2

```
#include <Arduino.h>
#include "pins.h"

void initBlink()
{
    pinMode(LED_PIN, OUTPUT);
}

void handleBlink()
{
    digitalWrite(LED_PIN, HIGH);
    delay(1000);
    digitalWrite(LED_PIN, LOW);
    delay(1000);
}
```

# Rule #1

- Don't call `delay()`
- Try to get out of `handleFoo()` quickly

# Blink 3

```
#include <Arduino.h>
#include "pins.h"

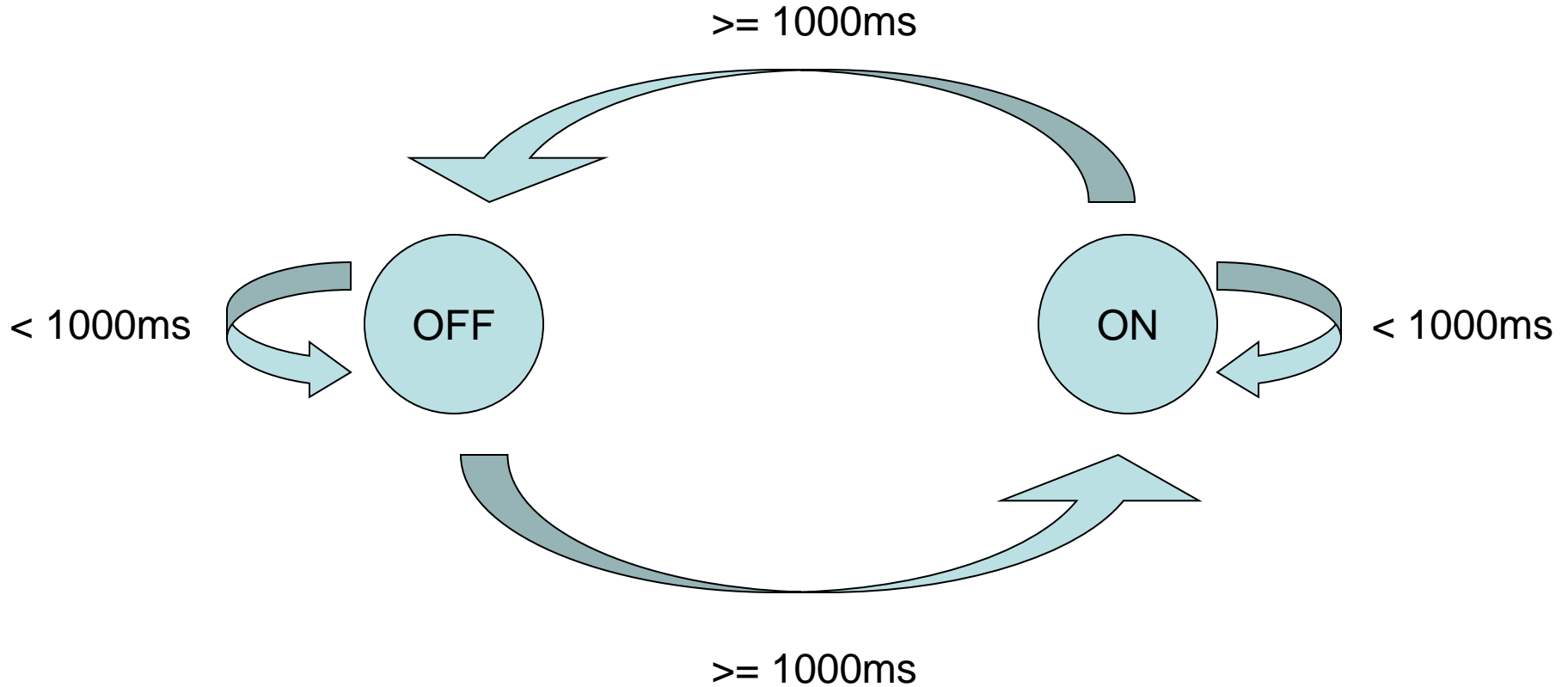
uint32_t timeChanged;
uint8_t ledState;

void initBlink()
{
    pinMode(LED_PIN, OUTPUT);
    ledState = HIGH;
    digitalWrite(LED_PIN, ledState);
    timeChanged = millis();
}

void handleBlink()
{
    uint32_t now = millis();
    if (now - timeChanged >= 1000) {
        ledState = ledState ^ HIGH;
        digitalWrite(LED_PIN, ledState);
        timeChanged = now;
    }
}
```



# Blink State Machine



# Blink 4

```
#include <Arduino.h>
#include "pins.h"

#define STATE_OFF 0
#define STATE_ON 1

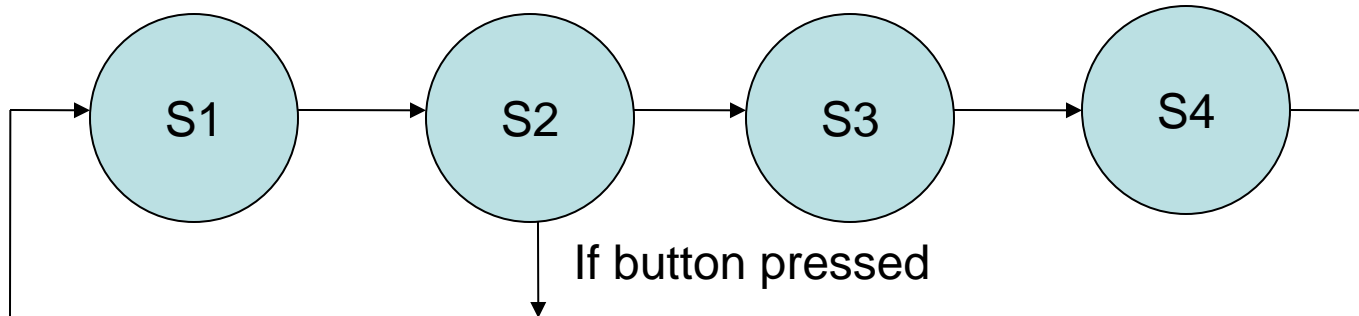
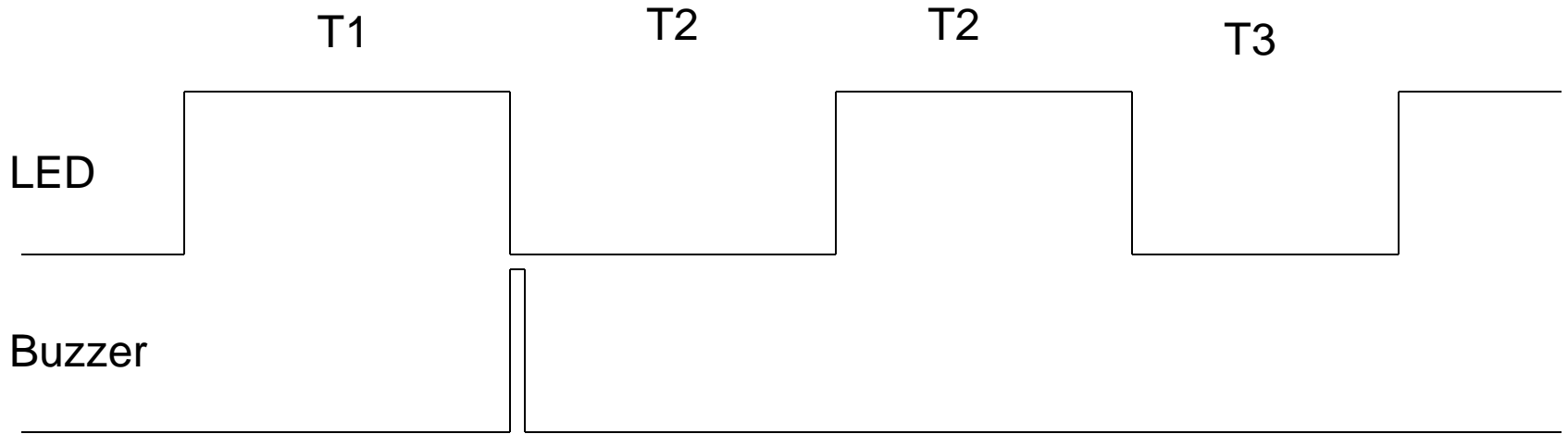
#define TIME_ON 1000
#define TIME_OFF 500

uint32_t timeChanged;
uint8_t state;

void initBlink()
{
    pinMode(LED_PIN, OUTPUT);
    state = STATE_ON;
    digitalWrite(LED_PIN, HIGH);
    timeChanged = millis();
}

void handleBlink()
{
    if (STATE_ON == state) {
        uint32_t now = millis();
        if (now - timeChanged >= TIME_ON) {
            digitalWrite(LED_PIN, LOW);
            state = STATE_OFF;
            timeChanged = now;
        }
    } else if (STATE_OFF == state) {
        uint32_t now = millis();
        if (now - timeChanged >= TIME_OFF) {
            digitalWrite(LED_PIN, HIGH);
            state = STATE_ON;
            timeChanged = now;
        }
    }
}
```

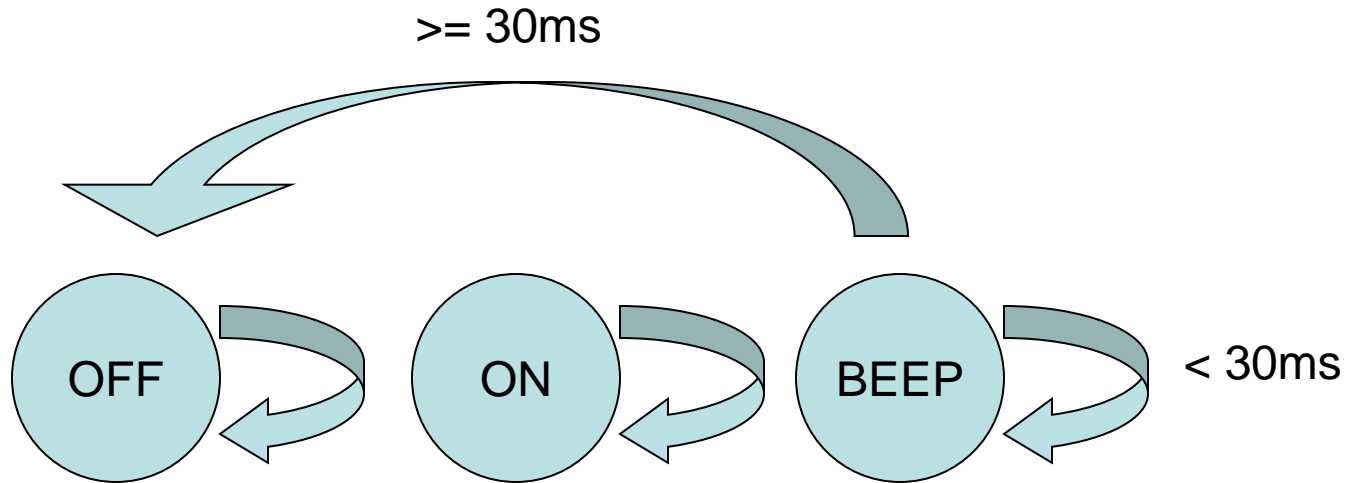
# Something more complex



# State table

State	LED	Buzzer	Delay	Next
S1	On		T1	S2
S2	Off	Beep	T2	S3 or S1
S3	On		T2	S4
S4	Off		T3	S1

# Buzzer State Machine



```

#include <Arduino.h>
#include "pins.h"

#define BEEP_TIME 30 // ms

uint8_t buzzerState;
uint8_t lastBuzzerState;
uint32_t enteredState;

void initBuzzer()
{
    pinMode(BUZZER_PIN, OUTPUT);
    digitalWrite(BUZZER_PIN, LOW);
    buzzerState = BSTATE_OFF;
    lastBuzzerState = BSTATE_OFF;
}

void handleBuzzer()
{
    if (BSTATE_OFF == buzzerState) {
        if (buzzerState != lastBuzzerState) {
            digitalWrite(BUZZER_PIN, LOW);
            lastBuzzerState = buzzerState;
        }
    }
}

```

```

    } else if (BSTATE_ON == buzzerState) {
        if (buzzerState != lastBuzzerState) {
            digitalWrite(BUZZER_PIN, HIGH);
            lastBuzzerState = buzzerState;
        }
    }

    } else if (BSTATE_BEEP == buzzerState) {
        if (buzzerState != lastBuzzerState) {
            enteredState = millis();
            digitalWrite(BUZZER_PIN, HIGH);
            lastBuzzerState = buzzerState;
        } else {
            uint32_t now = millis();
            if (now - enteredState >= BEEP_TIME) {
                buzzerState = BSTATE_OFF;
            }
        }
    }
}
}
}

```

# Blink 5

```
#include <Arduino.h>
#include "pins.h"
```

```
#define STATE_S1 0
#define STATE_S2 1
#define STATE_S3 2
#define STATE_S4 3
```

```
#define T1 1000
#define T2 500
#define T3 3000
```

```
uint32_t timeChanged;
uint8_t state;
```

```
void initBlink()
{
    pinMode(LED_PIN, OUTPUT);
    state = STATE_S1;
    digitalWrite(LED_PIN, HIGH);
    timeChanged = millis();
}
```

# Blink 5 (continued)

```
void handleBlink()
{
    if (STATE_S1 == state) {
        uint32_t now = millis();
        if (now - timeChanged >= T1) {
            digitalWrite(LED_PIN, LOW);
            state = STATE_S2;
            buzzerState =
BUZZER_STATE_BEEP; // beep
the buzzer as we enter S2
            timeChanged = now;
        }
    } else if (STATE_S2 == state) {
        uint32_t now = millis();
        if (now - timeChanged >= T2) {
            digitalWrite(LED_PIN, HIGH);
            if (buttonPressed) {
                state = STATE_S1;
            } else {
                state = STATE_S3;
            }
            buttonPressed = false;
            timeChanged = now;
        }
    }
```



# Blink 5 (continued)

```
} else if (STATE_S3 == state) {  
    uint32_t now = millis();  
    if (now - timeChanged >= T2) {  
        digitalWrite(LED_PIN, LOW);  
        state = STATE_S4;  
        timeChanged = now;  
    }  
}
```

```
} else if (STATE_S4 == state) {  
    uint32_t now = millis();  
    if (now - timeChanged >= T3) {  
        digitalWrite(LED_PIN, HIGH);  
        state = STATE_S1;  
        timeChanged = now;  
    }  
}  
}
```