

Lab 1: Git version-control system, AVR tools

Elena Arjona Bustos

1. The link to my repository is:

<https://github.com/elenaab16/Digital-Electronics2>

2. Blink example.

- $|$: represents the logic OR.

0	0	0
0	1	1
1	0	1
1	1	1

- $\&$: represents the logic AND.

0	0	0
0	1	0
1	0	0
1	1	1

- \wedge : represents the logic XOR.

0	0	0
0	1	1
1	0	1
1	1	0

- \sim : represents one's complement.

0	1
1	0

- \ll : represents left shifting.

$0010 \ll 2 \rightarrow 1000$

3. Morse Code application.

```
/*  
  
* MorseCode.c  
  
*  
  
* Created: 29/09/2020 13:24:45  
  
* Author : Elena Arjona Bustos  
  
*/  
  
  
/* Defines -----*/  
  
#define LED_GREEN PB5 // AVR pin where green LED is connected  
  
#define SHORT_DELAY 500 // Delay in milliseconds  
  
#define LONG_DELAY 1000 // Delay in milliseconds  
  
  
#ifndef F_CPU  
  
#define F_CPU 16000000 // CPU frequency in Hz required for delay  
  
#endif  
  
  
/* Includes -----*/  
  
#include <util/delay.h> // Functions for busy-wait delay loops  
  
#include <avr/io.h> // AVR device-specific IO definitions  
  
  
#include <avr/io.h>  
  
  
  
/* Functions -----*/
```

```
/**
```

```
* Long wait function. Turns on the LED for the long period of time and then,
```

```
* it turn it off.
```

```
*/
```

```
void long_wait(void);
```

```
/**
```

```
* Short wait function. Turns on the LED for the short period of time and then,
```

```
* it turn it off.
```

```
*/
```

```
void short_wait(void);
```

```
/**
```

```
* Main function where the program execution begins. Toggle one LED
```

```
* and use function from the delay library.
```

```
*/
```

```
int main(void)
```

```
{
```

```
    // Set pin as output in Data Direction Register
```

```
    // DDRB = DDRB or 0010 0000
```

```
    DDRB = DDRB | (1<<LED_GREEN);
```

```
    // Set pin LOW in Data Register (LED off)
```

```
// PORTB = PORTB and 1101 1111  
PORTB = PORTB & ~(1<<LED_GREEN);
```

```
while (1)
```

```
{
```

```
    // Letter D -> _ . .
```

```
    long_wait();
```

```
    short_wait();
```

```
    short_wait();
```

```
    // Letter E -> - .
```

```
    long_wait();
```

```
    short_wait();
```

```
    // Number 2 -> . . _ _ _
```

```
    short_wait();
```

```
    short_wait();
```

```
    long_wait();
```

```
    long_wait();
```

```
    long_wait();
```

```
}
```

```
}
```

```
void long_wait(void){
```

```
    // Set pin HIGH in Data Register (LED on)
```

```
    // PORTB = PORTB and 1101 1111
```

```
    PORTB = PORTB | (1<<LED_GREEN);
```

```
    // Pause several milliseconds
```

```
    _delay_ms(LONG_DELAY);
```

```
    // Set pin LOW in Data Register (LED off)
```

```
    // PORTB = PORTB and 1101 1111
```

```
    PORTB = PORTB & ~(1<<LED_GREEN);
```

```
    // Pause several milliseconds
```

```
    _delay_ms(SHORT_DELAY);
```

```
}
```

```
void short_wait(void){
```

```
    // Set pin HIGH in Data Register (LED on)
```

```
    // PORTB = PORTB and 1101 1111
```

```
    PORTB = PORTB | (1<<LED_GREEN);
```

```
// Pause several milliseconds  
  
_delay_ms(SHORT_DELAY);  
  
// Set pin LOW in Data Register (LED off)  
  
// PORTB = PORTB and 1101 1111  
PORTB = PORTB & ~(1<<LED_GREEN);
```

```
// Pause several milliseconds  
  
_delay_ms(SHORT_DELAY);
```

```
}
```