Labs/01-tools

Github Digital-electronic-2

Binary operators

symbol	meaning
&	AND
|	OR
^	XOR
~	NOT
<<	Binary left shift
>>	Binary right shift

Truth table

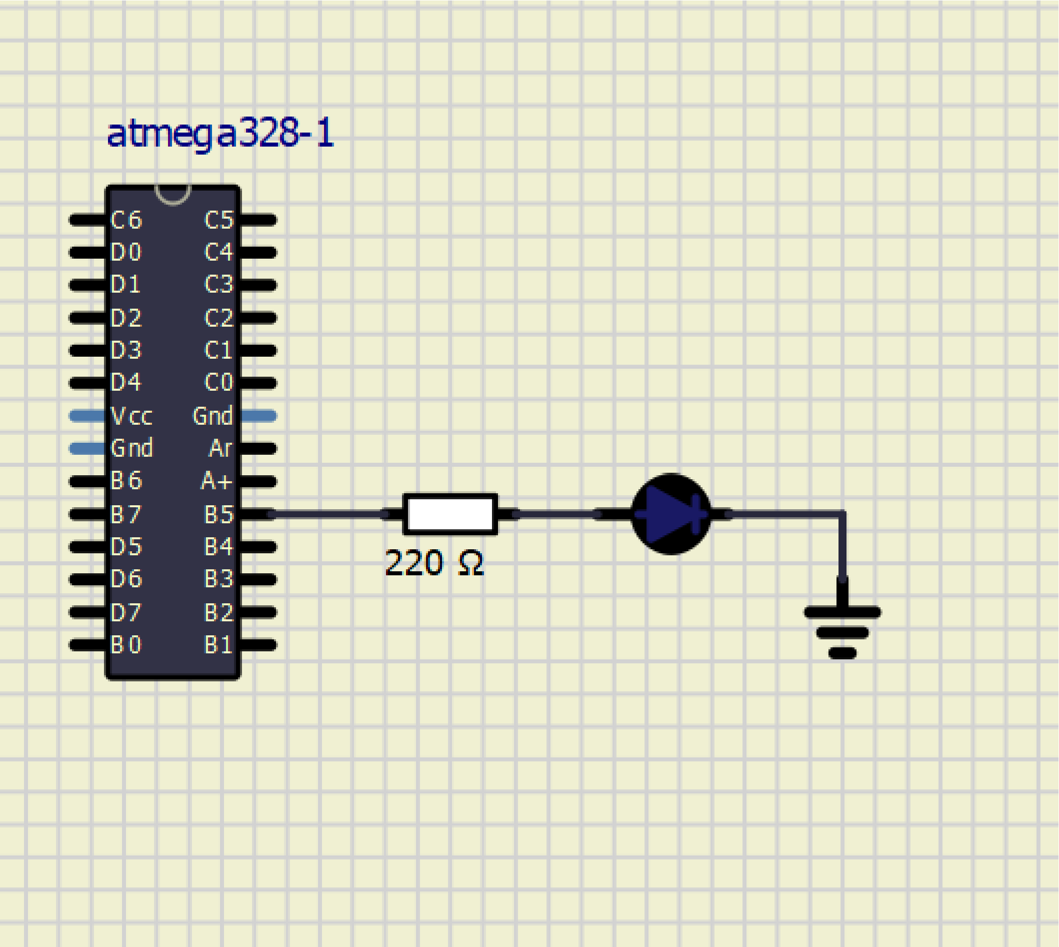
A	B	A & B	A | B	A ^ B	~A
0	0	0	0	0	1
0	1	0	1	1	1
1	0	0	1	1	0
1	1	1	1	0	0

Morse code

```
#define LED_GREEN   PB5 // AVR pin where green LED is connected
#define SHORT_DELAY 250 // Delay in milliseconds
#define DOT_DELAY   200
#define DASH_DELAY  600 // dash is 3 times longer than dot
#ifndef F_CPU           // Preprocessor directive allows for conditional
                        // compilation. The #ifndef means "if not defined".
# define F_CPU 16000000 // CPU frequency in Hz required for delay
#endif                  // The #ifndef directive must be closed by #endif

#include <util/delay.h> // Functions for busy-wait delay loops
#include <avr/io.h>     // AVR device-specific IO definitions

void dot() {
    _delay_ms(DOT_DELAY);               //Wait for pause
    PORTB = PORTB | (1<<LED_GREEN);     //Turn on for dot
    _delay_ms(DOT_DELAY);               //Wait for dot
    PORTB = PORTB & ~(1<<LED_GREEN);    //Turn off
```

```c
}

void dash() {
    _delay_ms(DOT_DELAY);               //Wait for pause
    PORTB = PORTB | (1<<LED_GREEN);     //Turn on for dash
    _delay_ms(DASH_DELAY);              //Wait for dash
    PORTB = PORTB & ~(1<<LED_GREEN);    //Turn off
}

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);

    // Infinite loop
    while (1)
    {
        dash();                  //D
        dot();
        dot();
        _delay_ms(DOT_DELAY);   //Pause
        dot();                  //E
        _delay_ms(DOT_DELAY);   //Pause
        dot();                  //2
        dot();
        dash();
        dot();
        dot();
        _delay_ms(DASH_DELAY);  //End of a word
    }

    // Will never reach this
    return 0;
}
```

Scheme


atmega328-1
C6
D0
D1
D2
D3
D4
Vcc
Gnd
B6
B7
D5
D6
D7
B0
C5
C4
C3
C2
C1
C0
Gnd
Ar
A+
B5
B4
B3
B2
B1
220 Ω