

Code

Issues

Pull requests

Actions

Projects

Wiki

Security

Insights

...

master ▾

...

Digital-Electronics-2 / labs / lab2 / Readme.md



FilipPaul Update Readme.md



1 contributor

Raw

Blame

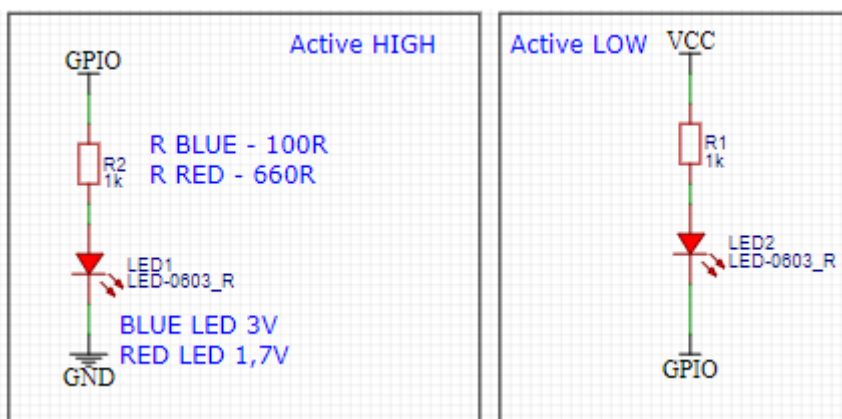


240 lines (202 sloc) 6.98 KB

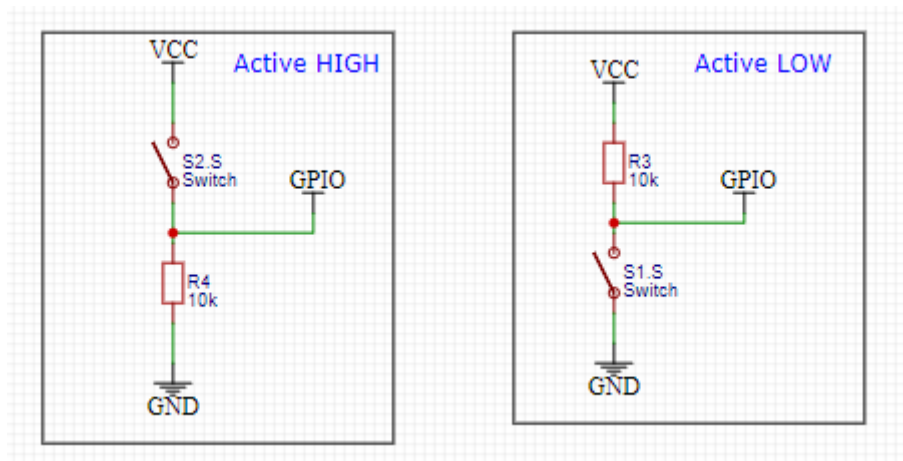
lab2

Preparation TASK

LEDs in common wiring



Pushbuttons in common wiring



computation of current limiting resistors

$$R = \frac{V_{SUPPLY} - V_{LED}}{I} = \frac{5 - 3}{20mA} = 100\Omega$$

LED color	Supply voltage	LED current	LED voltage	Resistor value
red	5 V	20 mA	1,7 V	660 Ω
blue	5 V	20 mA	3 V	100 Ω

Lab results

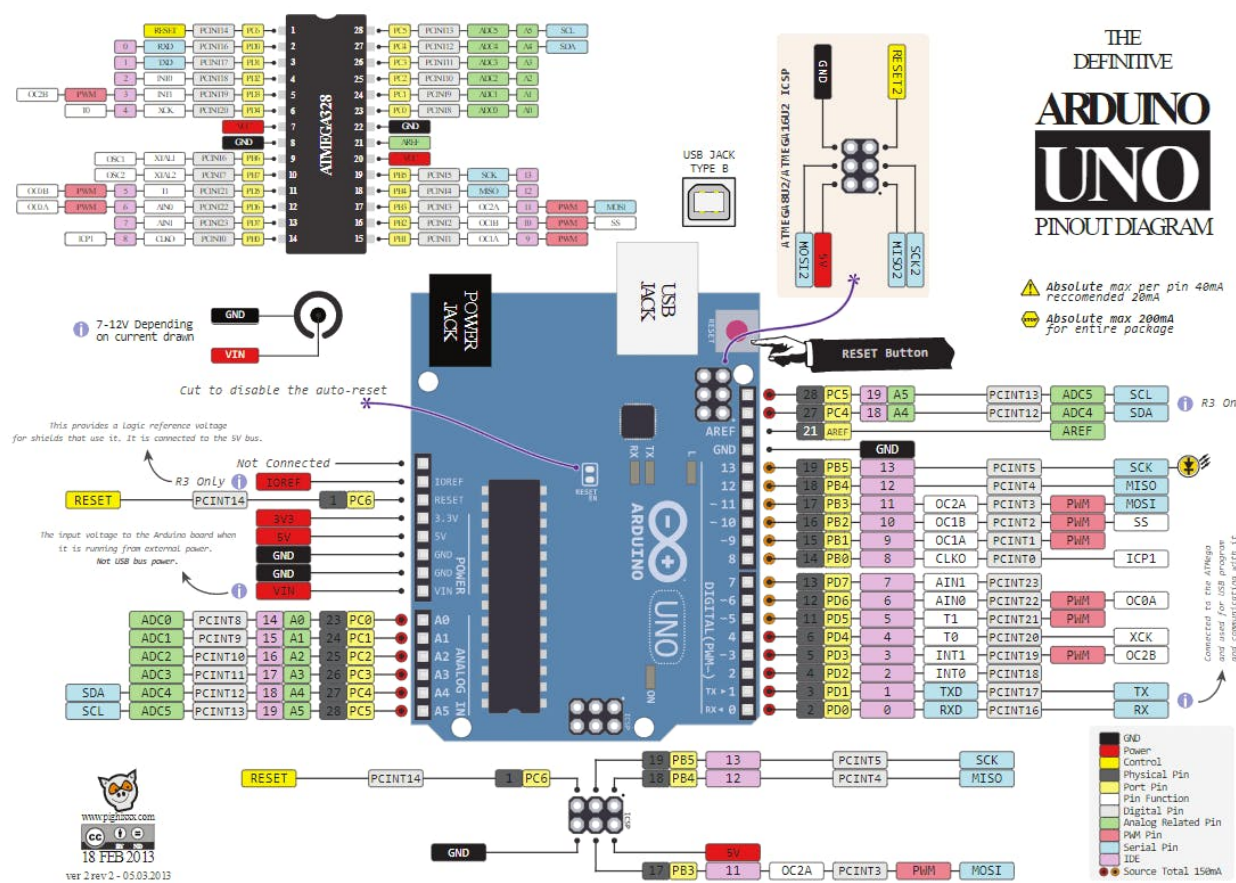
DDRB	Description
0	Input pin
1	Output pin

PORTB	Description
0	Output low value
1	Output HIGH value

DDRB	PORTB	PUD (in MCUCR)	Direction	Internal pull-up resistor	Description
0	0	X	input	no	Tri-state, high-impedance

DDRB	PORTB	PUD (in MCUCR)	Direction	Internal pull-up resistor	Description
0	1	0	input	yes	PBx will source current if ext.pulled low.
0	1	1	input	no	Tri-state, high-impedance
1	0	X	output	no	Output low (sink)
1	1	1	output	no	Output high (source)

Arduino UNO Schematics

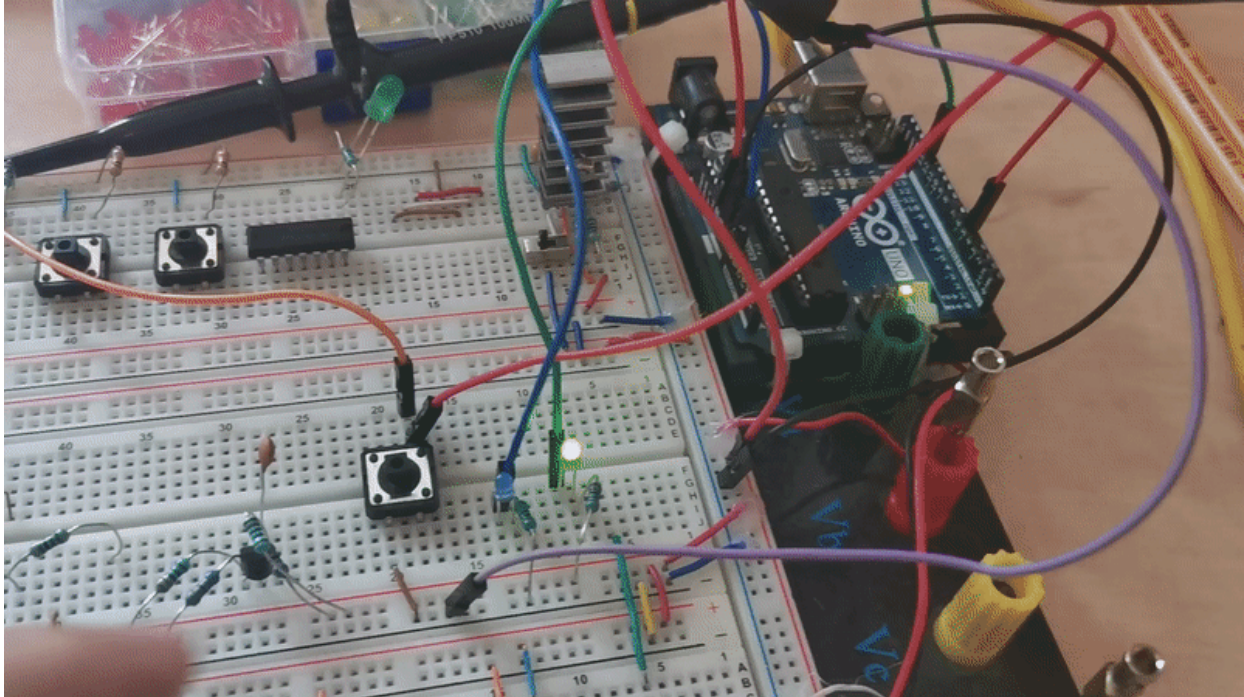


Port	Pin	Input/output usage?
A	x	Microcontroller ATmega328P does not contain port A
B	0	Yes (Arduino pin 8,CLKO,ICP1)
	1	Yes (Arduino pin 9,PWM)
	2	Yes (Arduino pin 10,SS,PWM)

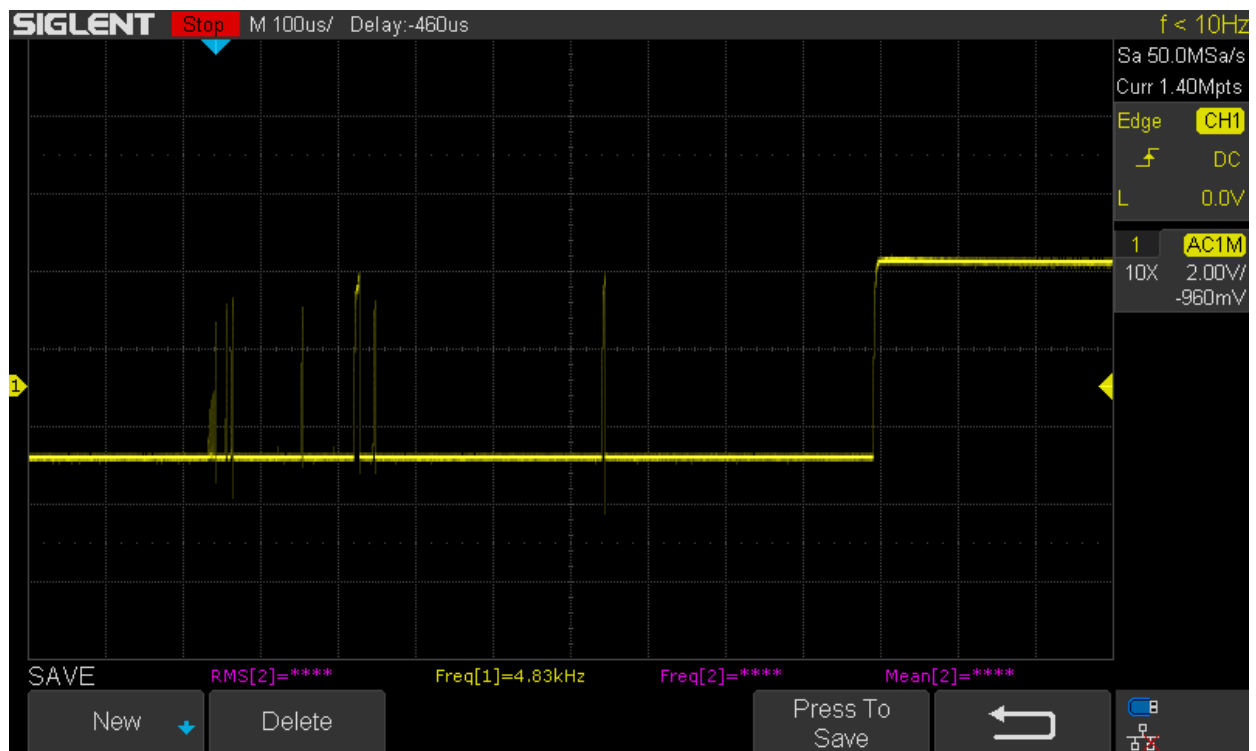
Port	Pin	Input/output usage?
	3	Yes (Arduino pin 11,MOSI,PWM)
	4	Yes (Arduino pin 12,MISO)
	5	Yes (Arduino pin 13, internal LED, SCK)
	6	NO(Arduino pin OSC1,XTAL1,)
	7	NO (Arduino pin OSC2,XTAL2,)
C	0	Yes (Arduino pin A0(14))
	1	Yes (Arduino pin A1(15))
	2	Yes (Arduino pin A2(16))
	3	Yes (Arduino pin A3(17))
	4	Yes (Arduino pin A4(18)),SDA
	5	Yes (Arduino pin A5(19)),SCL
	6	NO (Arduino pin RST)
	7	NO (Arduino pin Undefined)
D	0	Yes (Arduino pin RXD<-0)
	1	Yes (Arduino pin TXD(1)
	2	Yes (Arduino pin 2)
	3	Yes (Arduino pin 3,PWM)
	4	Yes (Arduino pin 4)
	5	Yes (Arduino pin 5, PWM)
	6	Yes (Arduino pin 6. PWM)
	7	Yes (Arduino pin 7, PWM)

Test circuit and code

the code is [here](#)



My own button bouncing



```

/*****
*
* Alternately toggle two LEDs when a push button is pressed.
* ATmega328P (Arduino Uno), 16 MHz, AVR 8-bit Toolchain 3.6.2
*
* Copyright (c) 2018-2020 Tomas Fryza
* Dept. of Radio Electronics, Brno University of Technology, Czechia

```

```

* This work is licensed under the terms of the MIT license.
*
*****/

/* Defines -----*/
#define LED_GREEN    PB5      //arduino 13 AVR pin where green LED is connected
#define LED_BLUE     PC0      // arduino 6
#define SWITCH       PD5      // arduino 5
#define BLINK_DELAY  250
#ifndef F_CPU
#define F_CPU 16000000        // CPU frequency in Hz required for delay
#endif
bool toggle_state = 1;

/* Includes -----*/
#include          // Functions for busy-wait delay loops
#include          // AVR device-specific IO definitions
#include          // needed for platformio

/* Functions -----*/

void toggle(){
toggle_state = !toggle_state;
_delay_ms(BLINK_DELAY);

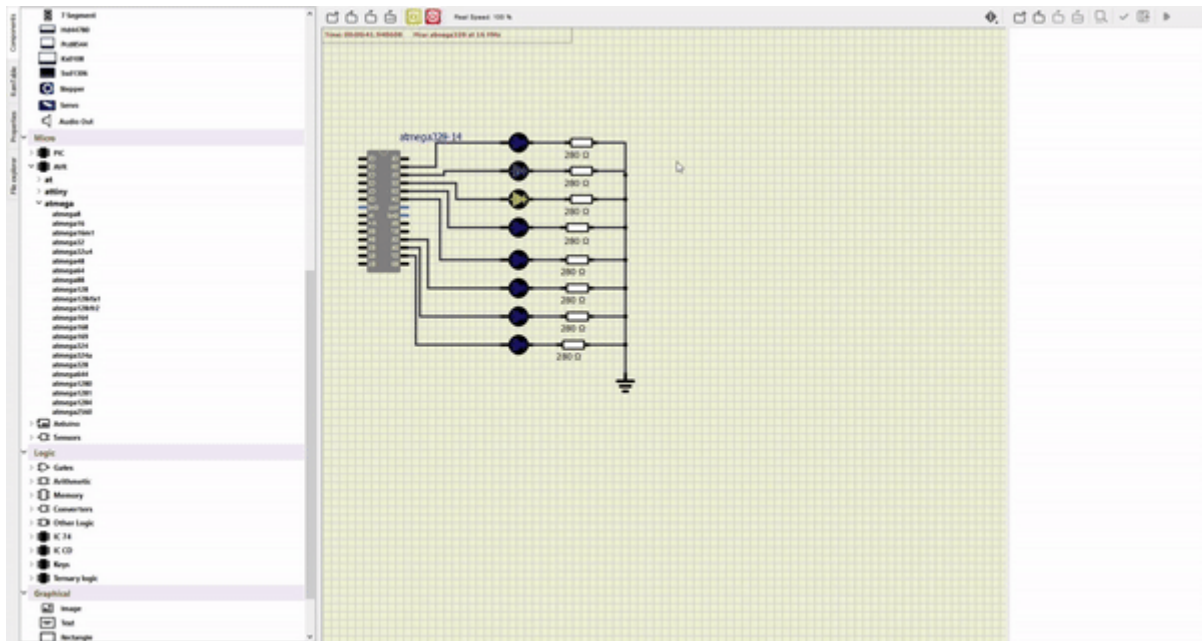
}
/**

* Main function where the program execution begins. Toggle two LEDs
* when a push button is pressed.
*/
int main(void)

{
    Serial.begin(9600);
    /*PUSHTBUTTON*/
    DDRD = DDRD & ~(0<

```

Knight Rider



```
<pre><code class="language-pascal">
#include <Arduino.h>
#define analog_pin = A0
#ifndef F_CPU
#define F_CPU 16000000 // CPU frequency in Hz required for delay
#endif
uint8_t analogValue = 100;
int led_detect = 0;
bool flag = 0;
void KnightRider()
{
    if (flag == 0){
        if ( led_detect == 7)
        {
            PORTD = PORTD & ~(1<<(led_detect-1));
            PORTD = PORTD | (1<<led_detect);
            led_detect = led_detect+1;
            flag = 1;
        }
        else if((led_detect < 8) && (led_detect != 0) ){
            PORTD = PORTD & ~(1<<(led_detect-1)); // turn pre led off
            PORTD = PORTD | (1<<led_detect); // turn next led on
            led_detect = led_detect +1;
        }
        else if(led_detect == 0){
            PORTD = PORTD | (1<<led_detect);
            led_detect = led_detect+1;
        }
    }
    else if (flag == 1) {
        if (led_detect == 1)
        {
            PORTD = PORTD & ~(1<<led_detect);
            PORTD = PORTD | (1<<(led_detect-1));
            led_detect = led_detect-1;
            flag = 0;
        }
        else if(led_detect == 8){
            PORTD = PORTD | (1<<led_detect);
            led_detect = led_detect-1;
        }
        else{
            PORTD = PORTD & ~(1<<led_detect);
            PORTD = PORTD | (1<<(led_detect-1));
        }
    }
}
```

```
        led_detect = led_detect - 1;
    }
}

int main(void){
    for (int i = 0; i < 7+1; i++){

        // Set pin as output in Data Direction Register...
        DDRD = DDRD | (1<<i);
        // ...and turn LED off in Data Register
        PORTD = PORTD & ~(1<<i);

    }

    while (1)
    {
        //mydelay = analogRead(0) //default 0-1023ms
        // there can be stored value potentiometer value, delay
        KnightRider();
        _delay_ms(100);
    }

    // Will never reach this
    return 0;
}
</code></pre>
```