⑂ main ▾   **Digital-electronics-2** / Labs / 02-leds / README.md   Go to file   ···

🟣 **xkocum00** Update README.md   Latest commit 8807a76 2 minutes ago   🕐 History

👥 **1 contributor**

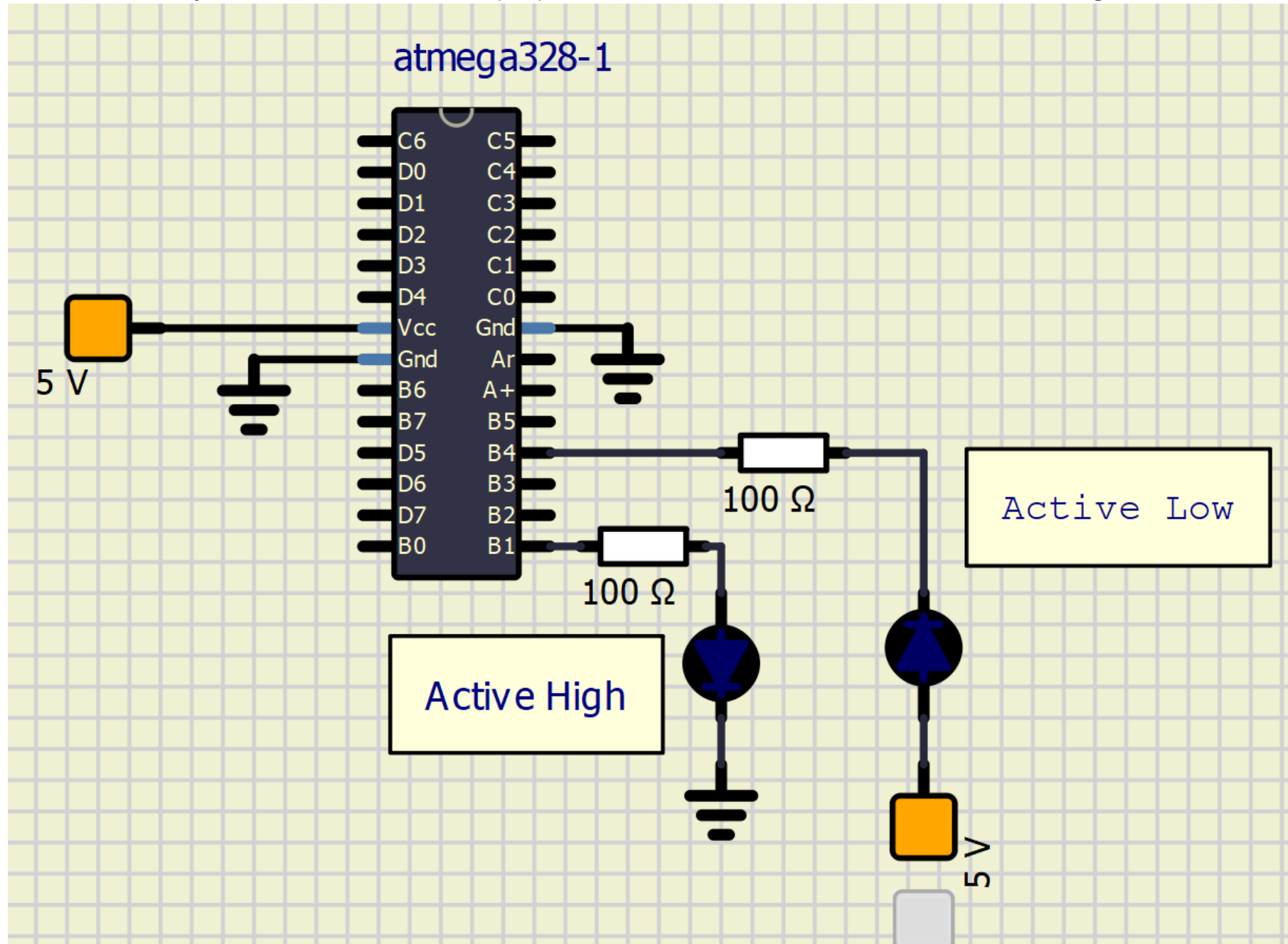☰ 157 lines (125 sloc) │ 5.28 KB   <> 📄   Raw   Blame   🖥 📋 ✏ 🗑

Lab 2: FILIP KOCUM

github link of my repository

Preparation tasks (done before the lab at home)

1. Draw two basic ways to connect a LED to the output pin of the microcontroller: LED active-low, LED active-high.

atmega328-1

C6	C5
DO	C4
D1	C3
D2	C2
D3	C1
D4	C0
Vcc	Gnd
Gnd	Ar
B6	A+
B7	B5
D5	B4
D6	B3
D7	B2
B0	B1

5 V

100 Ω

Active Low

100 Ω

Active High

5 V

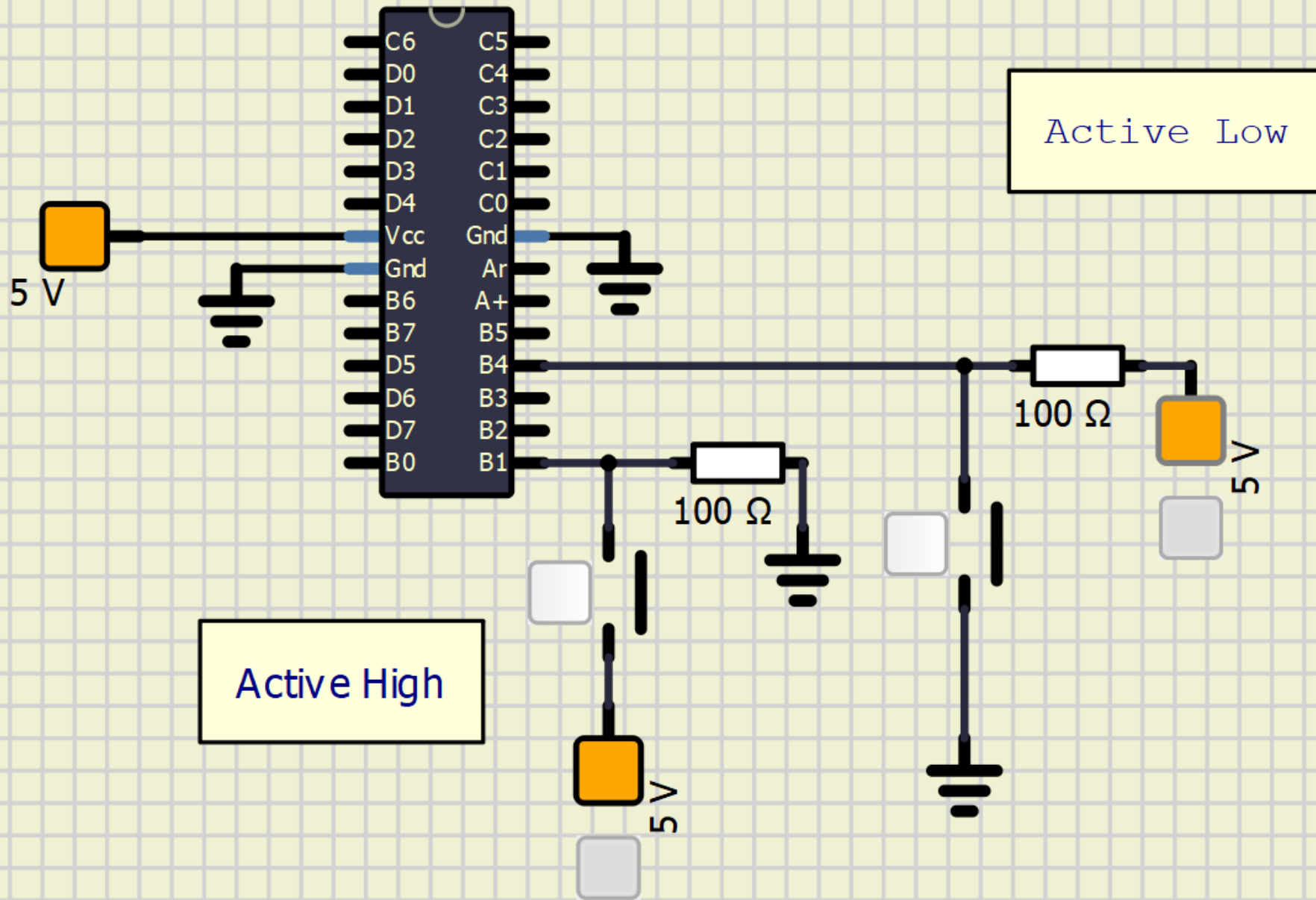2. Calculate LED resistor value for typical red and blue LEDs.

$$R = \frac{V_{SUPPLY} - V_{LED}}{I} = \text{(5-2)/0,02 = 150 } \Omega \qquad R = \frac{V_{SUPPLY} - V_{LED}}{I} = \text{(5-3)/0,02 = 100 } \Omega$$

LED color	Supply voltage	LED current	LED voltage	Resistor value
red	5 V	20 mA	2V	150 ohm
blue	5 V	20 mA	3V	100 ohm

Note that, equation was generated by Online LaTeX Equation Editor using the following code.

3. Draw the basic ways to connect a push button to the microcontroller input pin: button active-low, button active-high.

atmega328-1

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

5 V

Active Low

100 Ω

5 V

Active High

100 Ω

5 V

Active-low and active-high LEDs

1. Complete tables according to the AVR manual.

DDRB	Description
0	Input pin
1	Output pin

PORTB	Description
0	Output low value
1	Output high value

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

2. See schematic of Arduino Uno board in docs folder of Digital-electronics-2 repository and find out which pins of ATmega328P can be used as input/output pins. To which pin is the LED L connected? Is it connected as active-low or active-high? Note that labels on Arduino ~3 , ~5 , etc. do not mean that the signals are inverted; the ~ symbol indicates that a PWM (Pulse-width modulation) signal can be generated on these pins.

Port	Pin	Input/output usage?

Port	Pin	Input/output usage?
A	x	Microcontroller ATmega328P does not contain port A
B	0	Yes Arduino pin D8
	1	Yes Arduino pin D9
	2	Yes Arduino pin D10
	3	Yes Arduino pin D11
	4	Yes Arduino pin D12
	5	Yes Arduino pin D13
	6	No
	7	No
C	0	Yes Arduino pin A0(D14)
	1	Yes Arduino pin A1(D15)
	2	Yes Arduino pin A2(D16)
	3	Yes Arduino pin A3(D17)
	4	Yes Arduino pin A4(D18)
	5	Yes Arduino pin A5(D19)
	6	Yes/Reset
	7	No
D	0	Yes (Arduino pin RX<-D0)

Port	Pin	Input/output usage?
	1	Yes (Arduino pin TX<-D1)
	2	Yes Arduino pin D2
	3	Yes Arduino pin D3
	4	Yes Arduino pin D4
	5	Yes Arduino pin D5
	6	Yes Arduino pin D6
	7	Yes Arduino pin D7

2. Part of the C code listing with syntax highlighting, which blinks alternately with a pair of LEDs; let one LED is connected to port B and the other to port C:

```c
int main(void)
{
    // Green LED at port B
    // Set pin as output in Data Direction Register...
    DDRB = DDRB | (1<<LED_GREEN);
    // ...and turn LED off in Data Register
    PORTB = PORTB & ~(1<<LED_GREEN);

    // Configure the second LED at port C
    // WRITE YOUR CODE HERE
    DDRC = DDRC | (1<<LED_GREEN1);
    PORTC = PORTC & ~(1<<LED_GREEN1);
    // Infinite loop
    while (1)
    {
        // Pause several milliseconds
        _delay_ms(BLINK_DELAY);
```

```c
    // WRITE YOUR CODE HERE
    PORTB = PORTB ^ (1<<LED_GREEN);      //LED ON
    _delay_ms(BLINK_DELAY);
    PORTB = PORTB & ~(1<<LED_GREEN);     //LED OFF
    PORTC = PORTC ^ (1<<LED_GREEN1);      //LED ON
    _delay_ms(BLINK_DELAY);
    PORTC = PORTC & ~(1<<LED_GREEN1);     //LED OFF
    }


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

Push button

1. Part of the C code listing with syntax highlighting, which toggles LEDs only if push button is pressed. Otherwise, the value of the LEDs does not change. Let the push button is connected to port D:

```c
int main(void)
{
    // Green LED at port B
    // Set pin as output in Data Direction Register...
    DDRB = DDRB | (1<<LED_GREEN);
    DDRC = DDRC | (1<<LED_GREEN1);
    // Configure Push button at port D and enable internal pull-up resistor
    DDRD = DDRD | (1<<BUTTON);
    PORTD = PORTD & ~(1<<BUTTON);
    // Infinite loop
    while (1)
    {
        if(bit_is_clear(PIND, PD0))
            {
                _delay_ms(BLINK_DELAY);
                //led toogling only if button is pressed
```
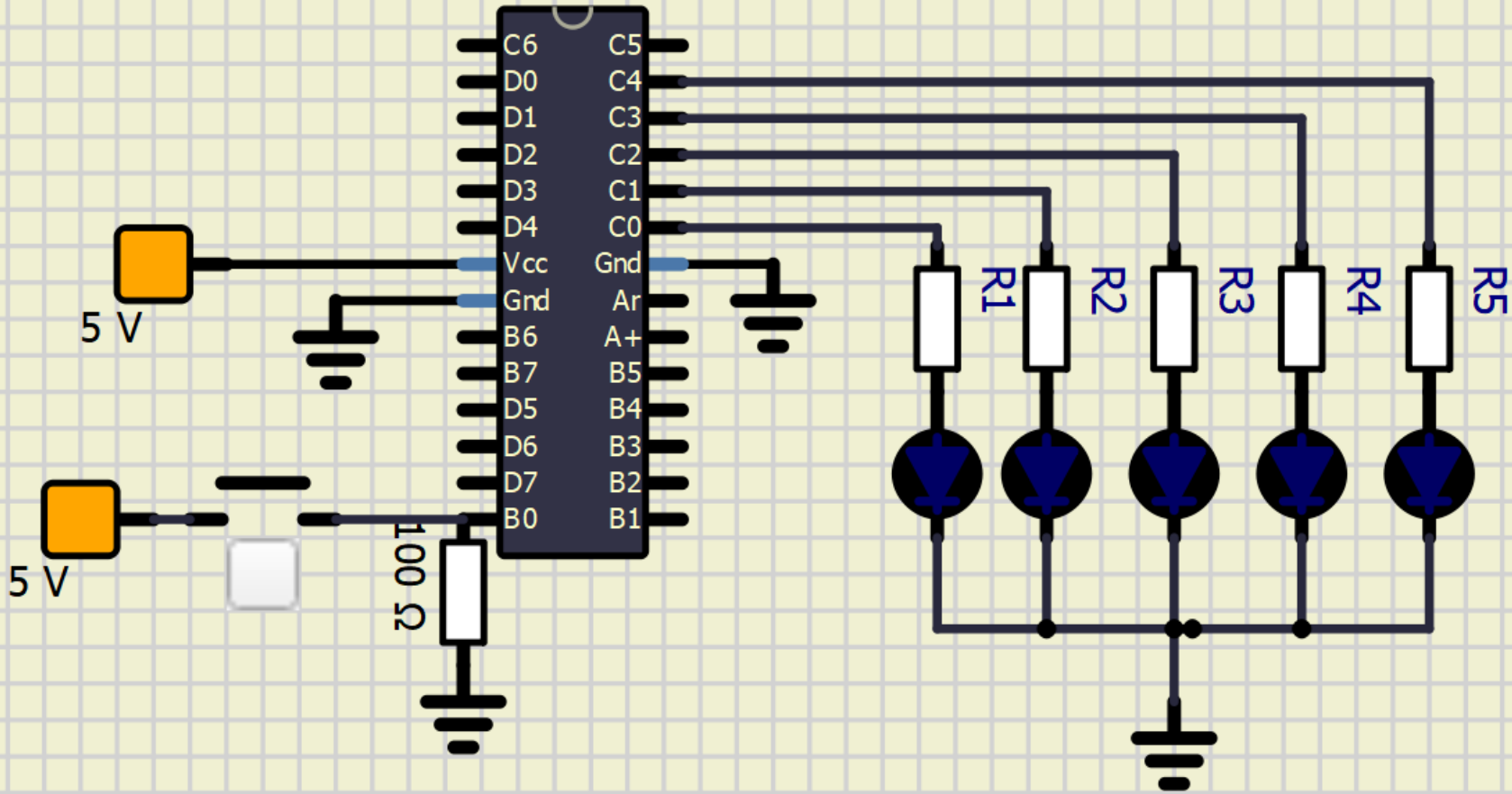
```
                PORTC = PORTC ^ (1<<LED_GREEN);
                PORTB = PORTB ^ (1<<LED_GREEN1);


        }
    }

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

Knight Rider

1. Scheme of Knight Rider application, i.e. connection of AVR device, five LEDs, resistors, one push button, and supply voltage. The image can be drawn on a computer or by hand. Always name all components and their values!

atmega328-1

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

5 V

5 V

100 Ω

R1 R2 R3 R4 R5

R1=R2=R3=R4=R5=150[Ω]