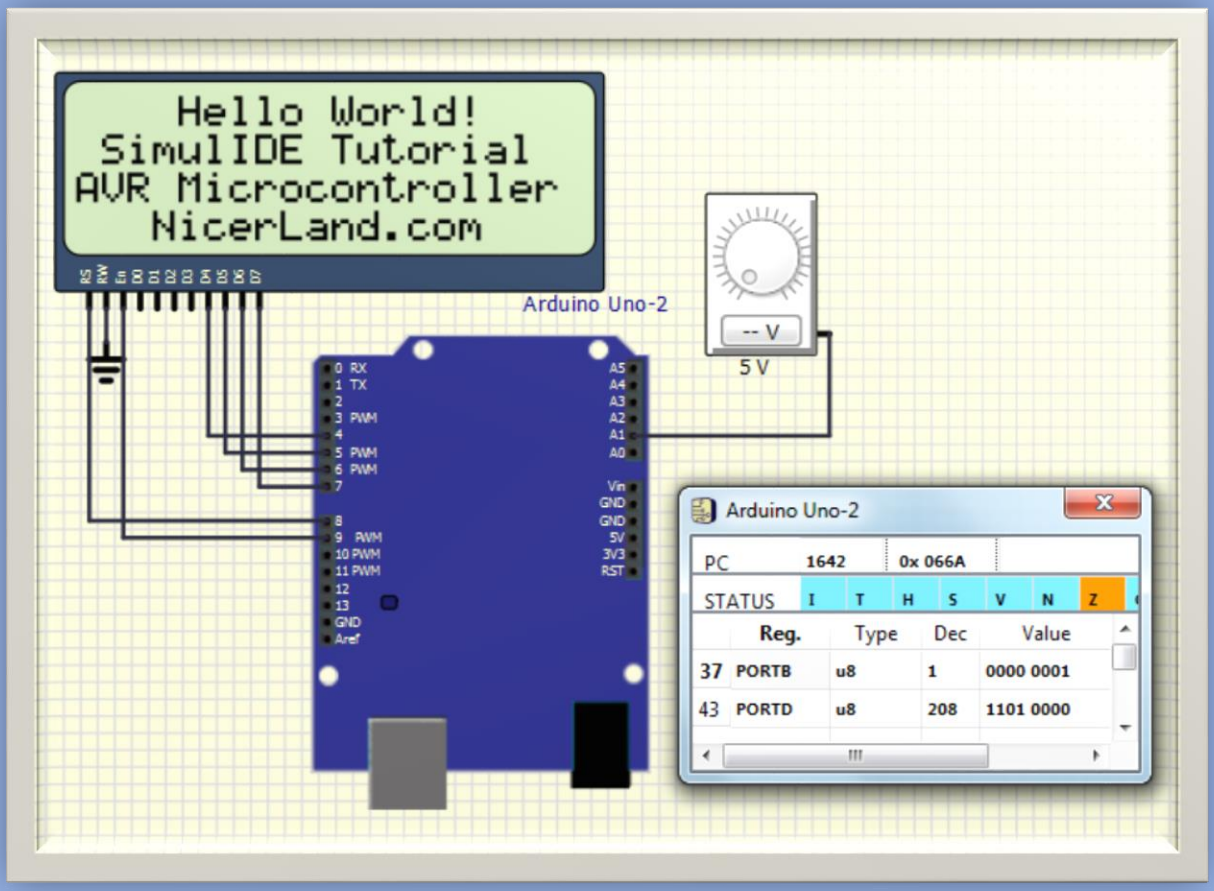
AVR Simulation Using SimulIDE



Sepehr Naimi

NicerLand.com

8/8/2021

Contents

Preface

SimulIDE is a free and open source simulator which can be used to simulate AVR and PIC microcontrollers and Arduino boards.

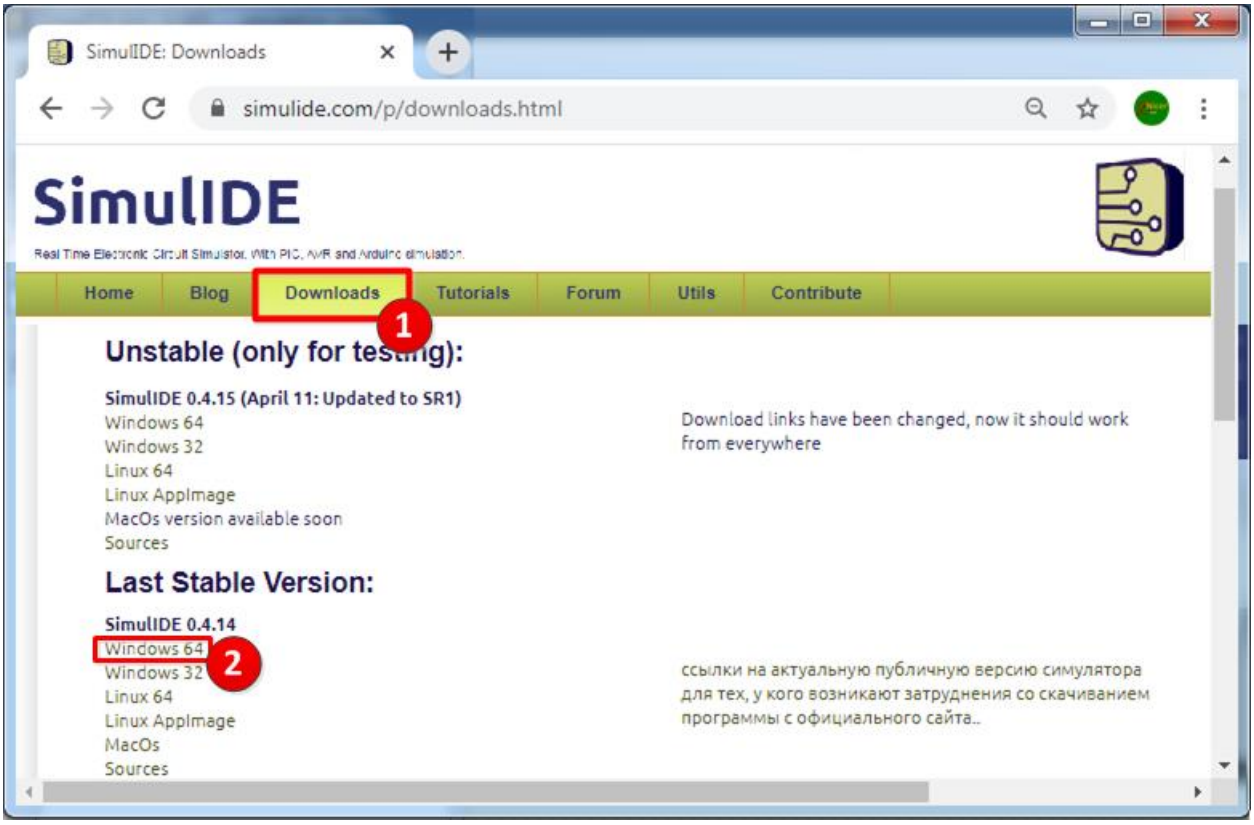
In the step-by-step tutorial, you will learn to download *Simulide* and simulate a simple circuit in the Simulide. Then, you write a program in Atmel Studio (or Microchip Studio) and simulate it in *Simulide*.

Dedication

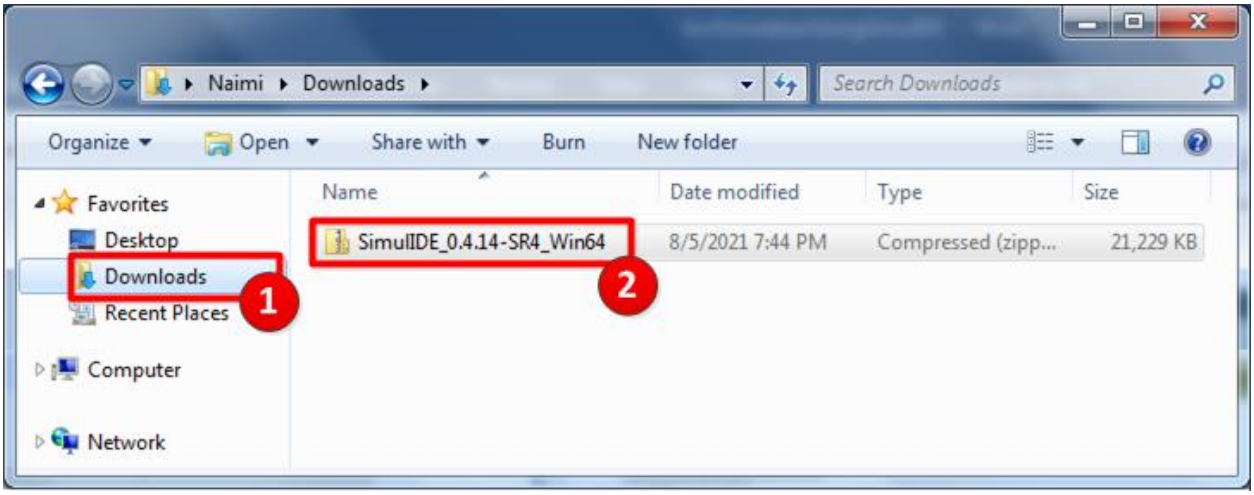
The step-by-step tutorial is dedicated to the memory of Vahid Mokhtari who sacrificed himself to serve humanity and make a better world and passed away a few days ago in Tehran.

Downloading SimulIDE

1. To download the latest version of SimulIDE, go to Simulide.com and click on Downloads.
2. Click on ***Windows 64*** (or any other choice depending on your operating system). A zip file will be downloaded to your computer.

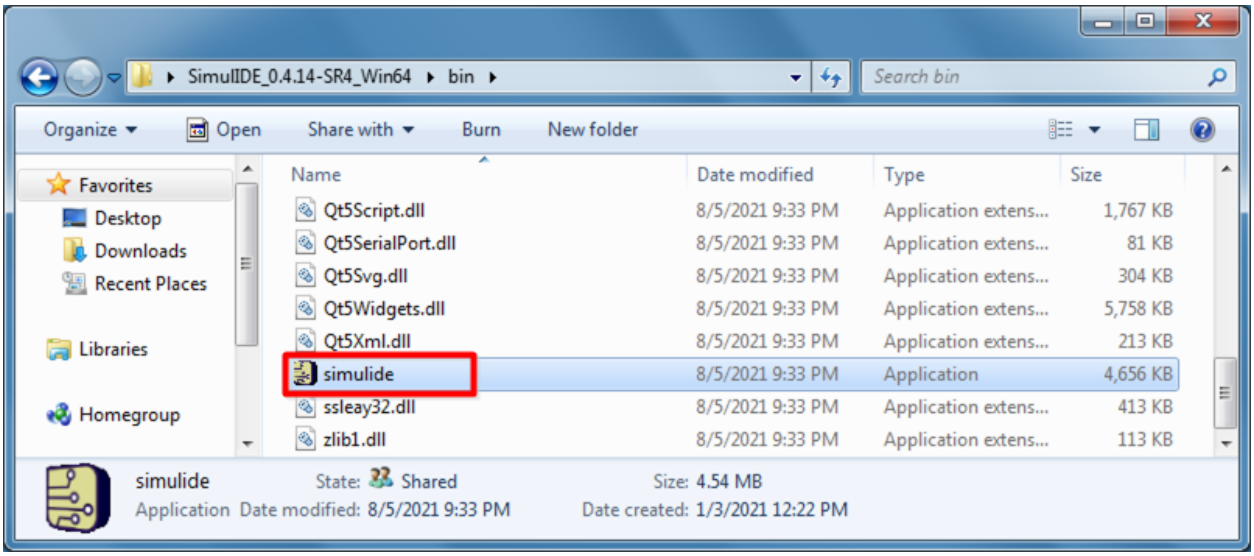


3. Unzip the downloaded file. (If you don't know where the file is downloaded, go to ***My Computer*** and click on ***Downloads***. The downloaded file is most probably here. Double click on the ***SimulIDE_0.4.14-SR4_Win64*** file. The compressed file contains a directory. To unzip, drag the directory to your desktop.)

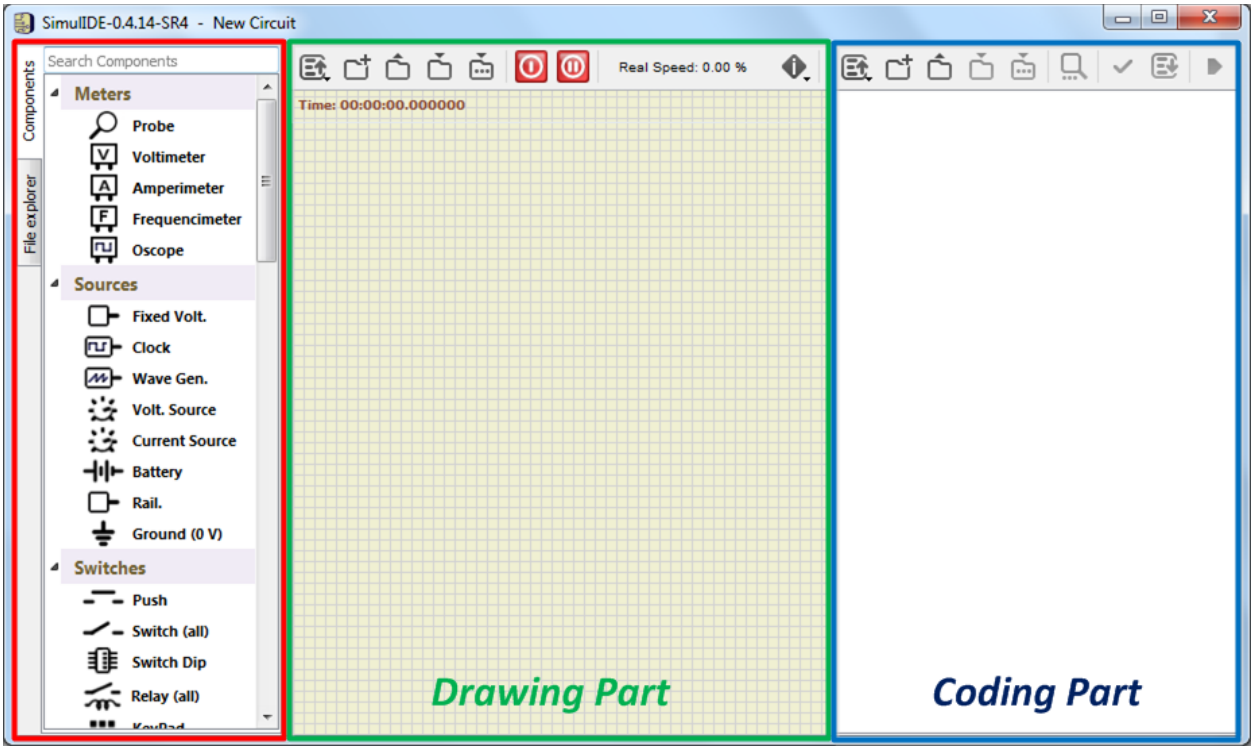


Opening SimulIDE

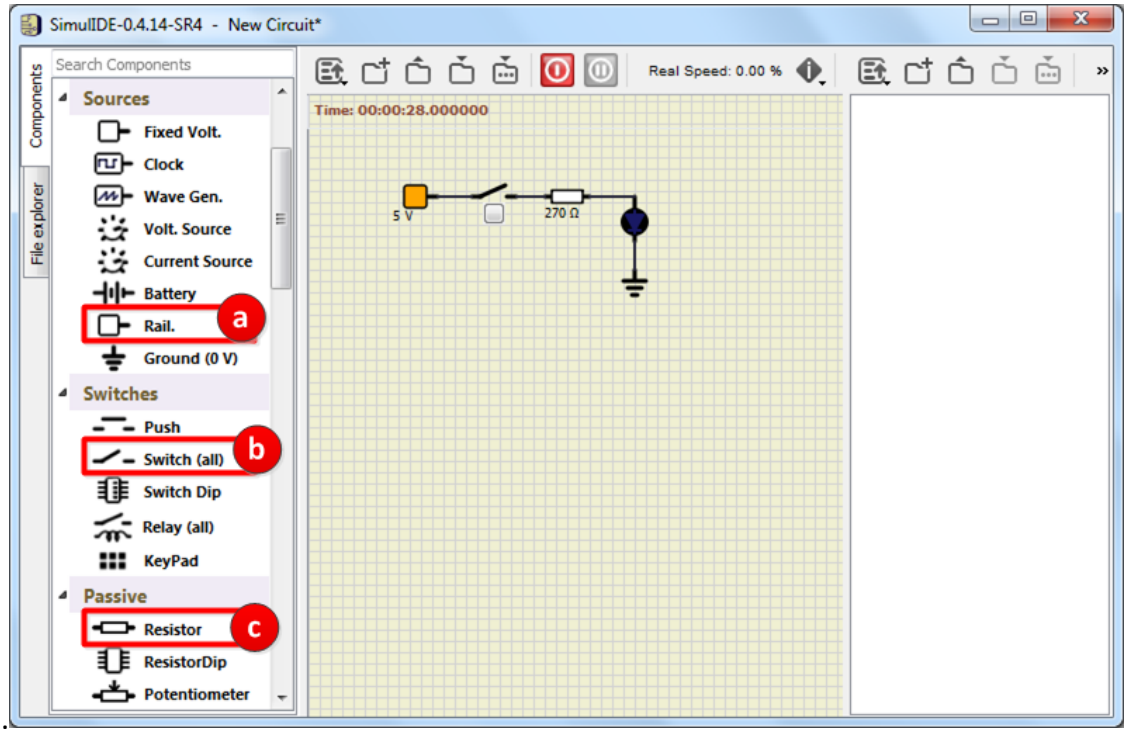
4. Open the decompressed directory. Click on the ***bin*** directory. Then, click on the ***simulide*** file to open it.



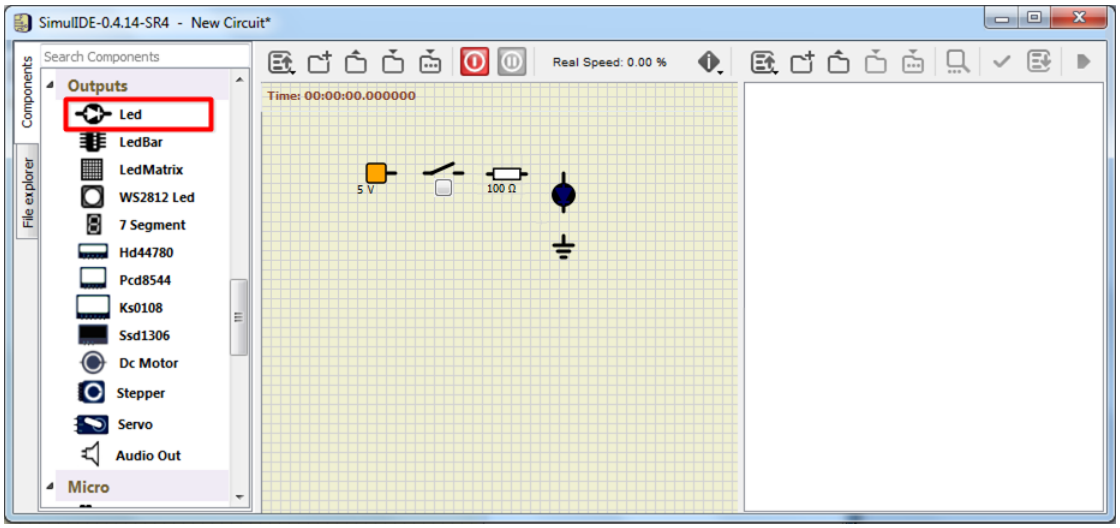
5. SimulIDE has 3 parts:
    a. the left part contains some components which can be used in circuits.
    b. In the center part, you can draw circuits.
    c. The right part can be used to edit source codes.



6. Draw the following circuit. To do so:
    a. Drag the ***Rail*** component and drop onto the Drawing Part. (Put the mouse pointer on the ***Rail***. Press the left button of mouse and while the button is down, move the mouse to the Drawing Part. Then, release the mouse button.)
    b. Drag a ***Switch*** component.
    c. Drag a ***Resistor***.
    d. Drag and drop a ***Ground (0V)***.

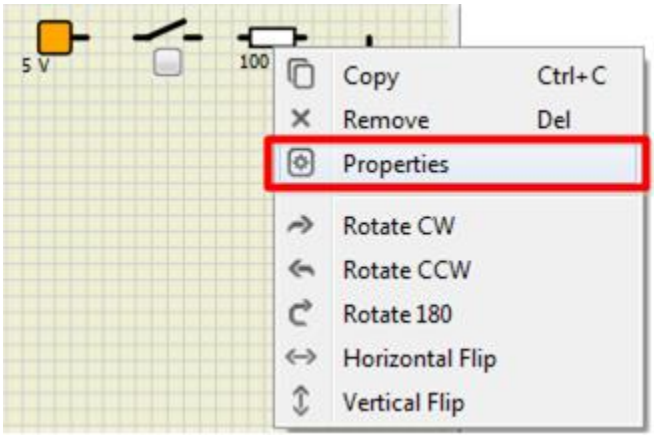


    e. Drag an ***LED*** and drop to the proper location. Then, to rotate the LED, right click on the LED and choose rotate CW (clockwise).

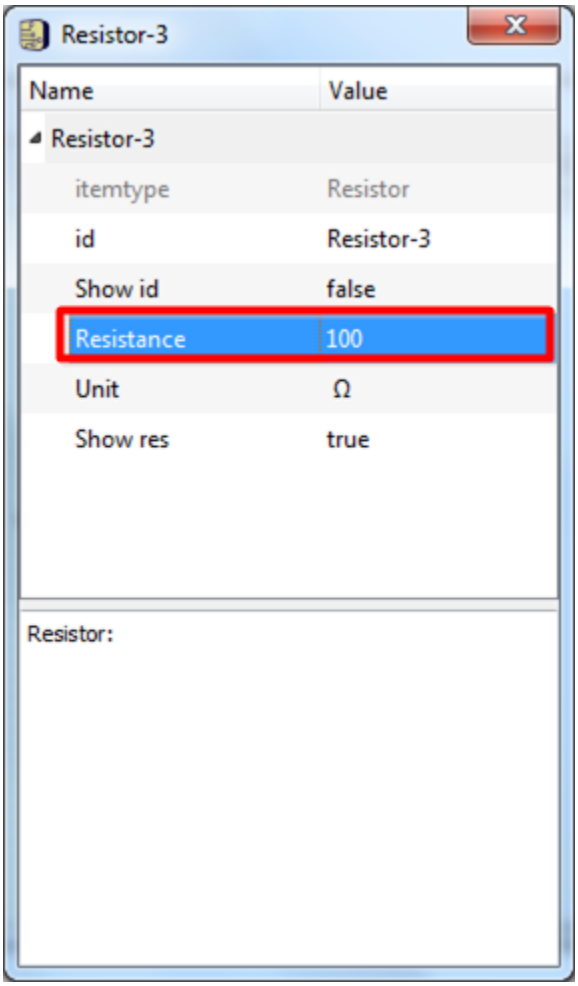


Changing the Properties

The resistor is 100Ω. But we need a 270Ω resistor. To access the properties of the resistor, right click on the resistor and choose Properties.

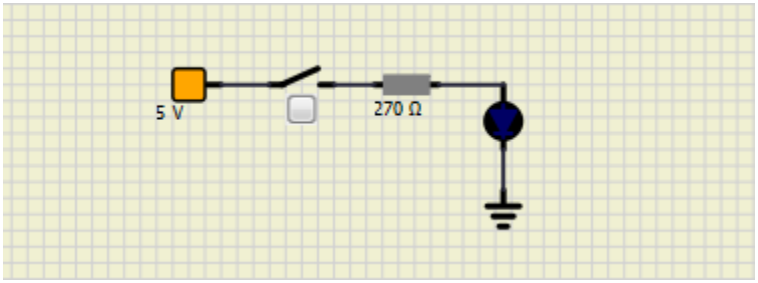


7. Double click on resistance. Then, type 270 and press Enter. Close the Properties.



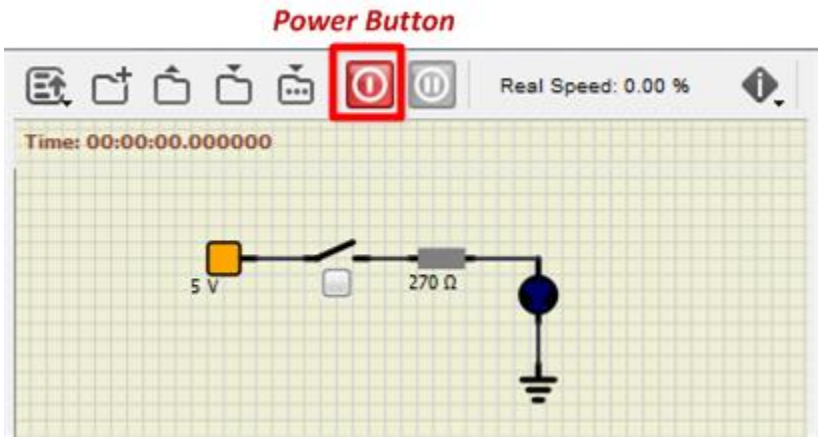
Wiring

8. To connect Ground to the LED, move the mouse to the top of Ground. The shape of the mouse pointer changes to cross. Now, click with the left button of mouse on top of the Ground. Then, move the mouse cursor to the cathode pin of LED and click on it. A wire should be added between the Ground and the LED.
9. Similarly, connect the other components together, as shown below.

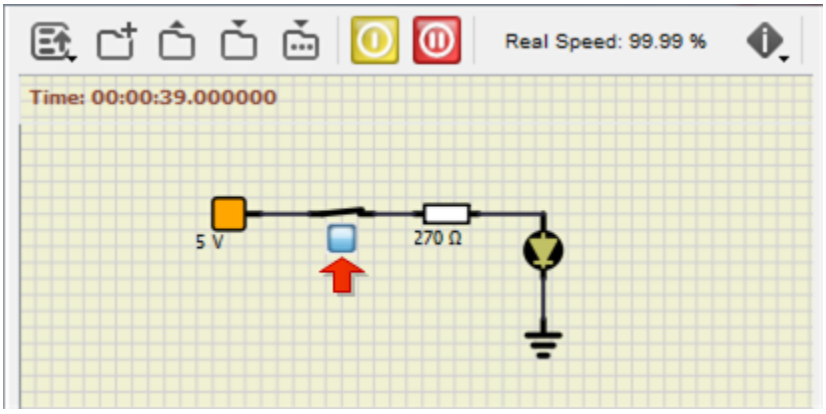


Simulating

10. To run the simulator, press the Power button.

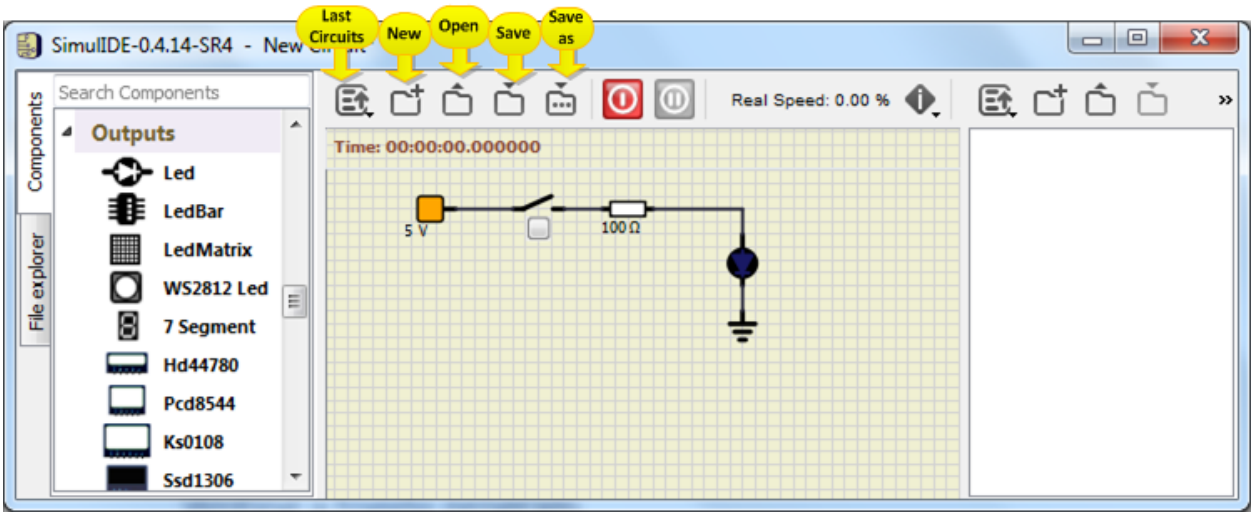


11. Now, press the gray button below the switch. When the switch is closed, the LED is ON. Otherwise, it is OFF.



Saving the simulation

12. To save the circuit, press Ctrl + S or click on the save icon.



Writing a toggle program

13. In Atmel Studio (or Microchip Studio) make a new project and write the following Assembly or C program. Choose ATmega328 as the chip.

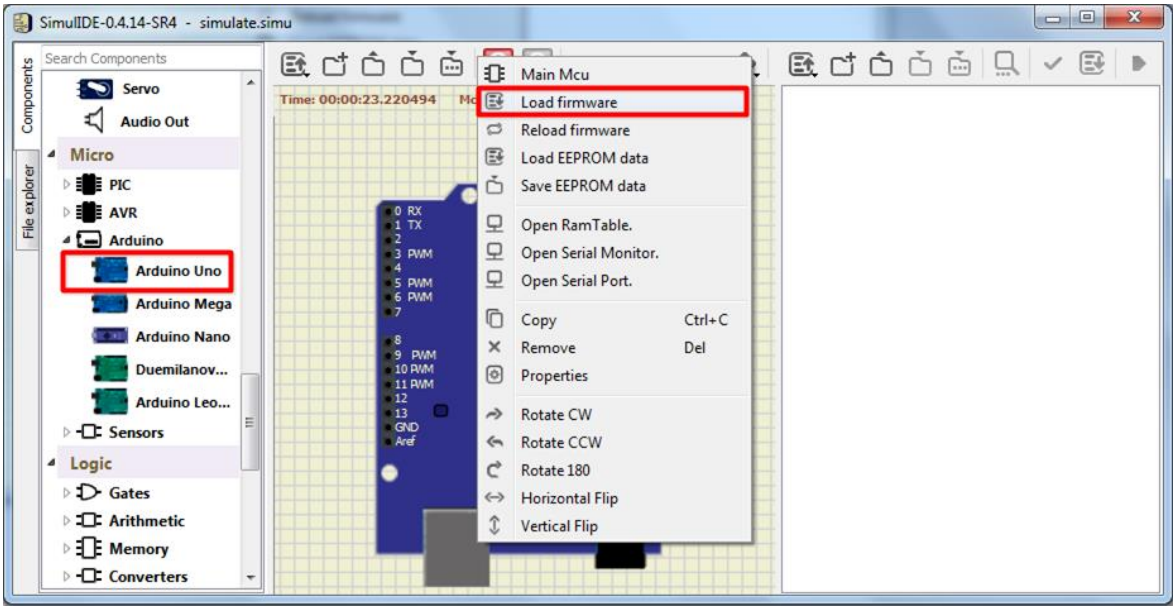
C

```
#include <avr/io.h>
#define F_CPU 16000000UL
#include <util/delay.h>

int main(void)
{
    DDRB |= (1<<5);
    while (1)
    {
        PORTB ^= (1<<5);
        _delay_ms(500);
    }
}
```

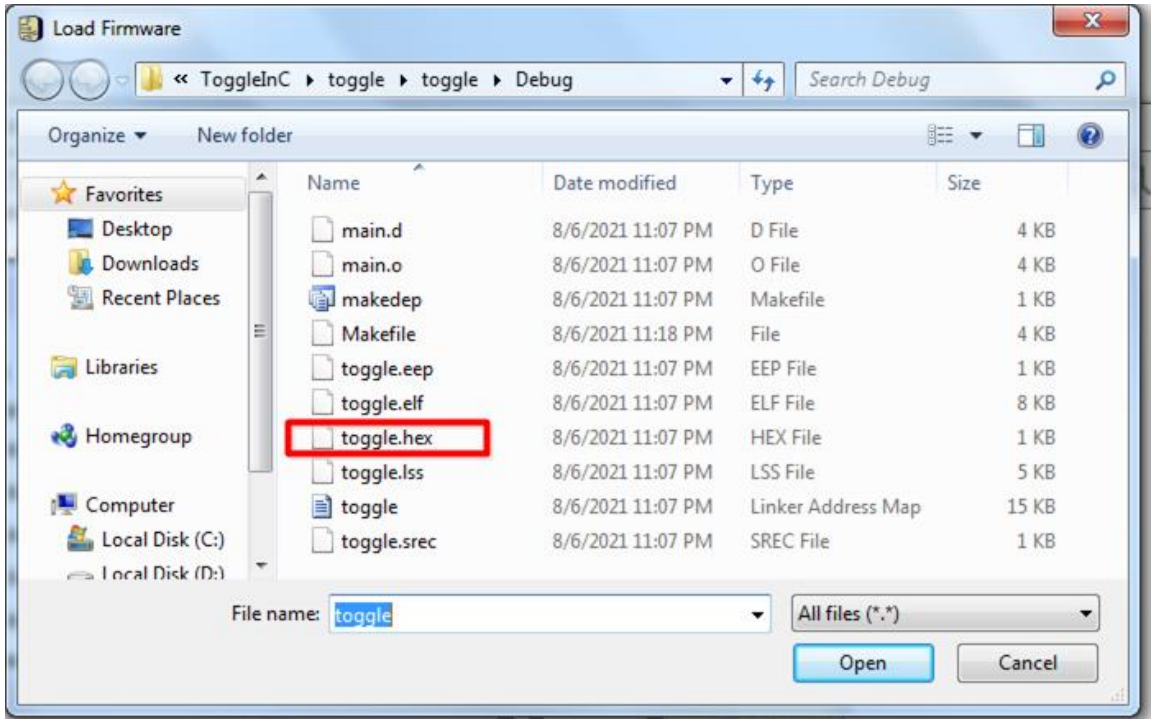
Assembly

```
        LDI     R16, HIGH(RAMEND)
        OUT     SPH, R16
        LDI     R16, LOW(RAMEND)
        OUT     SPL, R16

        SBI     DDRB, 5

L3:     SBI     PORTB,5
        CALL    DELAY
        CBI     PORTB,5
        CALL    DELAY
        RJMP    L3

DELAY:
        LDI     R20, 64
D0:     LDI     R21, 200
D1:     LDI     R22, 250
D2:     NOP
        NOP
        DEC     R22
        BRNE    D2
        DEC     R21
        BRNE    D1
        DEC     R20
        BRNE    D0
        RET
```

14. Press F7 to build the hex file.
15. Open Simulide.
16. Drag and drop an Arduino UNO board from the Component list.

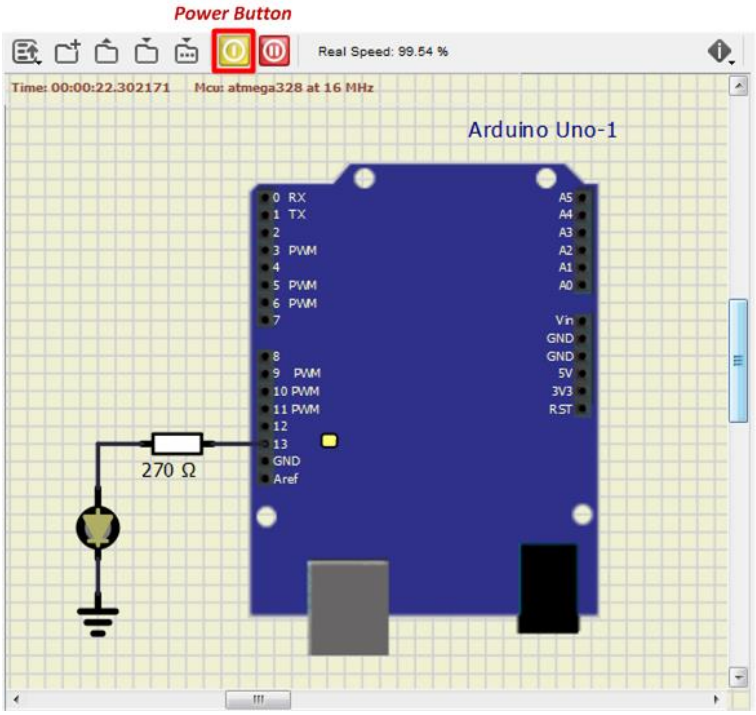


17. Right click on the Arduino board and choose ***Load Firmware***. Then, choose the hex file which you made in Atmel Studio.



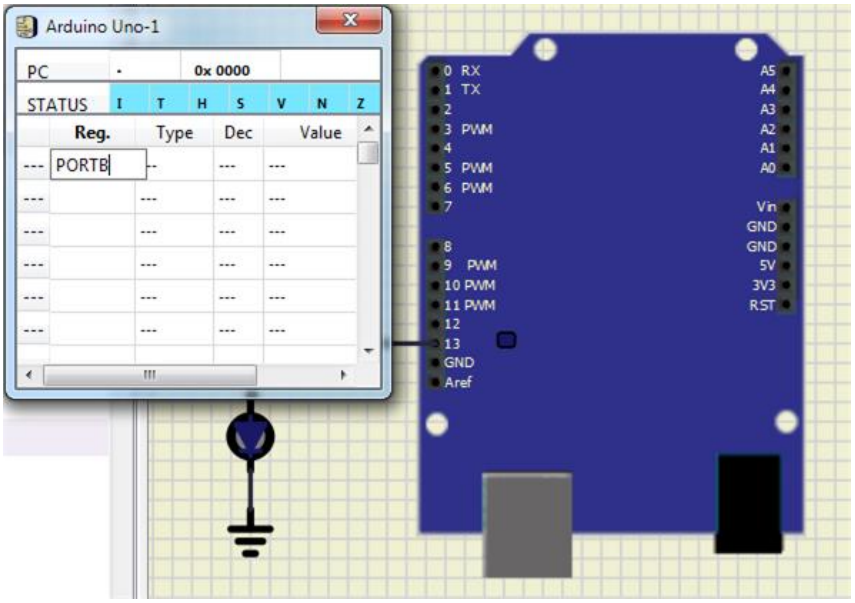
18. Press the Power Button to run the simulator. The LED of the Arduino board begins blinking. The LED is connected to pin 13 of Arduino (PB5 of ATmega328). You can also

connect an LED together with a 270-ohm resistor to pin 13 of Arduino. In the case, the LED also blinks.



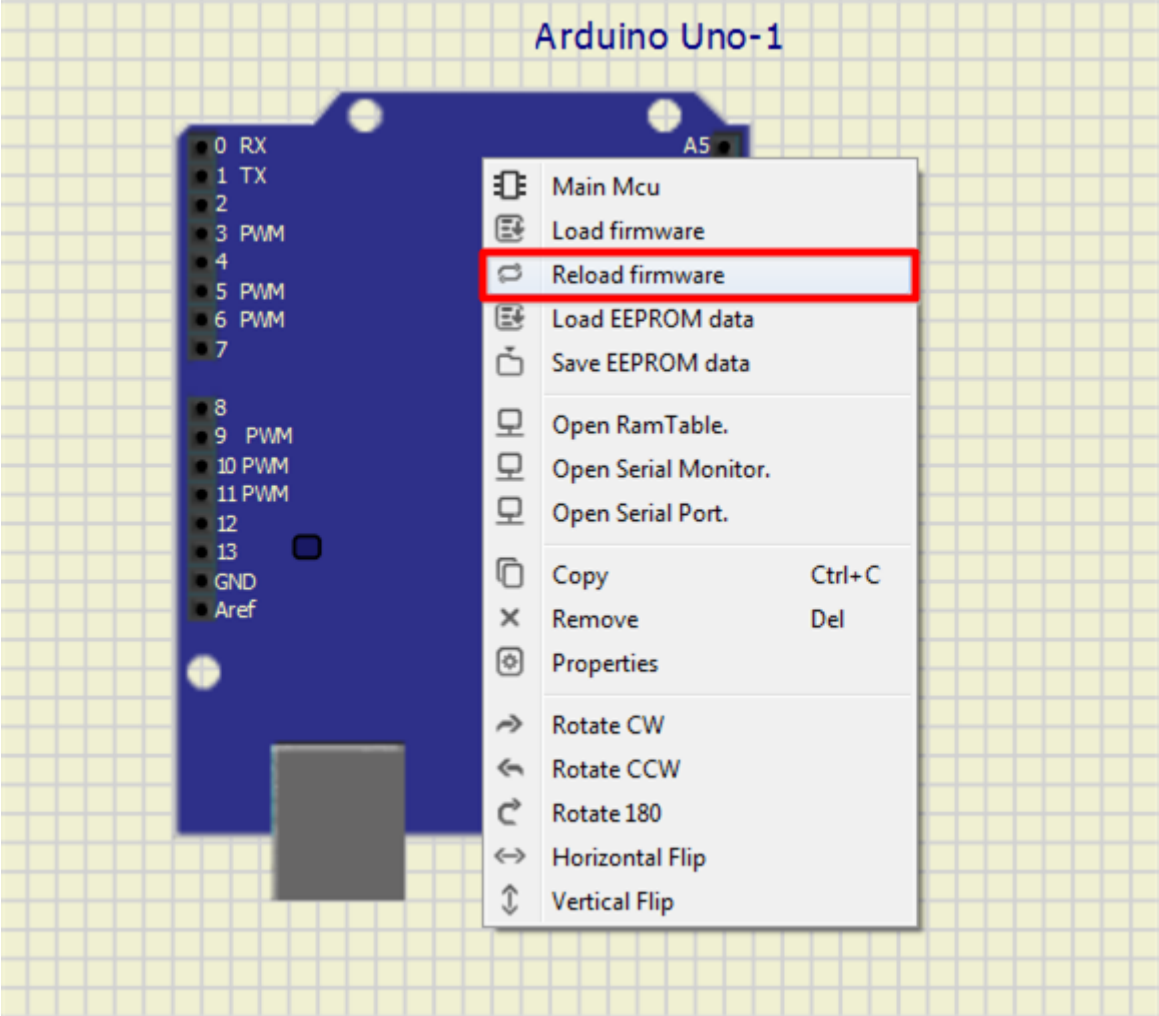
Monitoring Registers

19. You can monitor the values of I/O registers. To do so, right click on the Arduino board and choose Open RamTable.
20. Double click on a cell below the ***Reg.*** column and type PORTB and press Enter. The value of PORTB changes between 0 and 32, while the LED blinks.



Some Notes

1. If you change the source code, you need to reload the firmware to run the new code in SimulIDE. To do so, press the Power button to stop simulation. Then, right click on the Arduino (or the microcontroller) and choose ***Reload firmware***.



2. When you open your simulation file in the simulIDE, don't forget to load the firmware! SimulIDE does not save the firmware.