Electrical Training

Week 1: Prototyping and Breadboarding

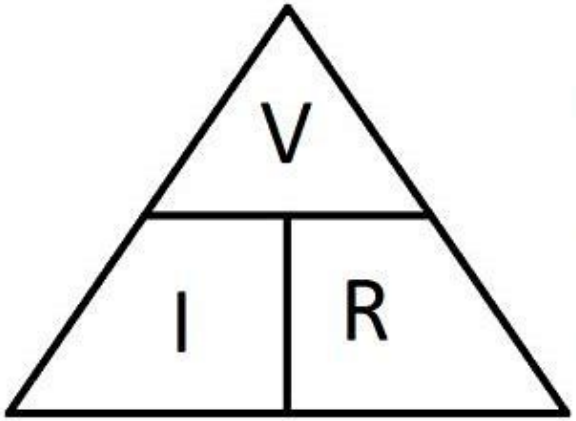
Agenda

- Electronics and Breadboarding
- Arduino Introduction
- Lab A--guided
- Lab B--challenges

Electronics and Breadboarding

Electricity Basics

- Voltage-V
  - Measured in Volts
- Resistance-R
  - Measured in Ohms
- Current-I
  - Measured in Amperes



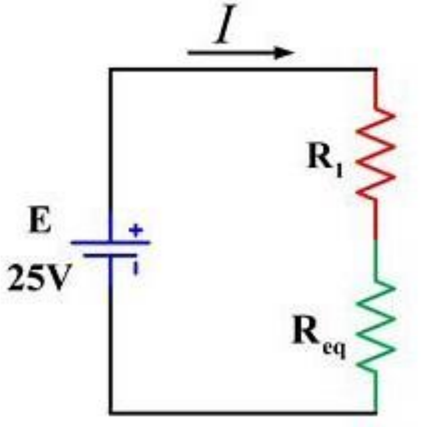
I = V/R

V = IR

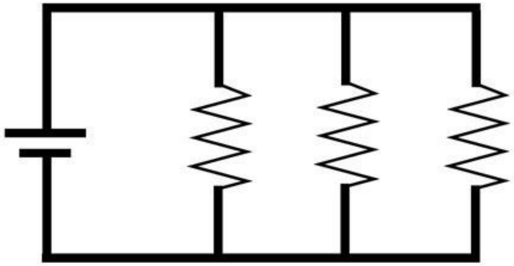
R = V/I

Electricity Basics

- Series
  - Current is constant across components
  - Voltage might not be the same across components
- Parallel
  - Voltage is constant across all branches
  - Current might not be the same on each branch

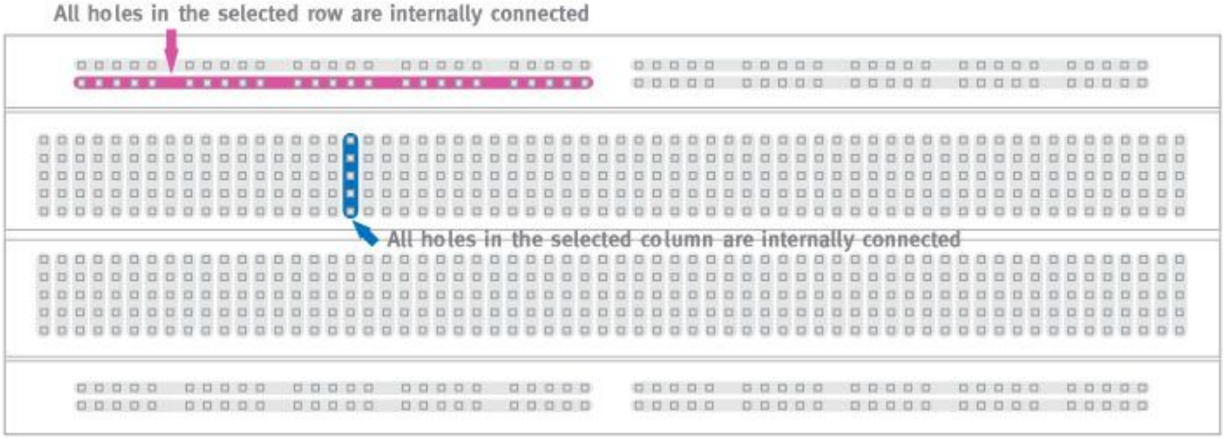


(b) Equivalent Series Circuit



Breadboard Basics

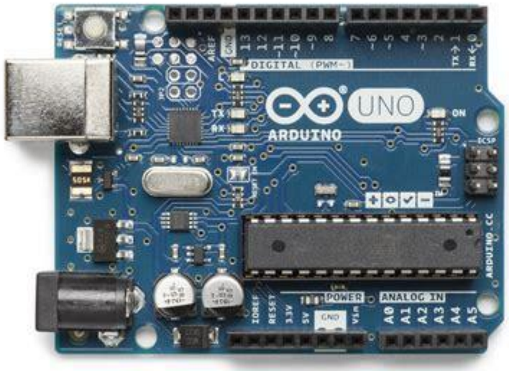
- Power rails-2 rows on each long side
- Internal connections arranged in columns, not rows
- DIP support - splits the columns down the middle


All holes in the selected row are internally connected
All holes in the selected column are internally connected

Arduino Introduction

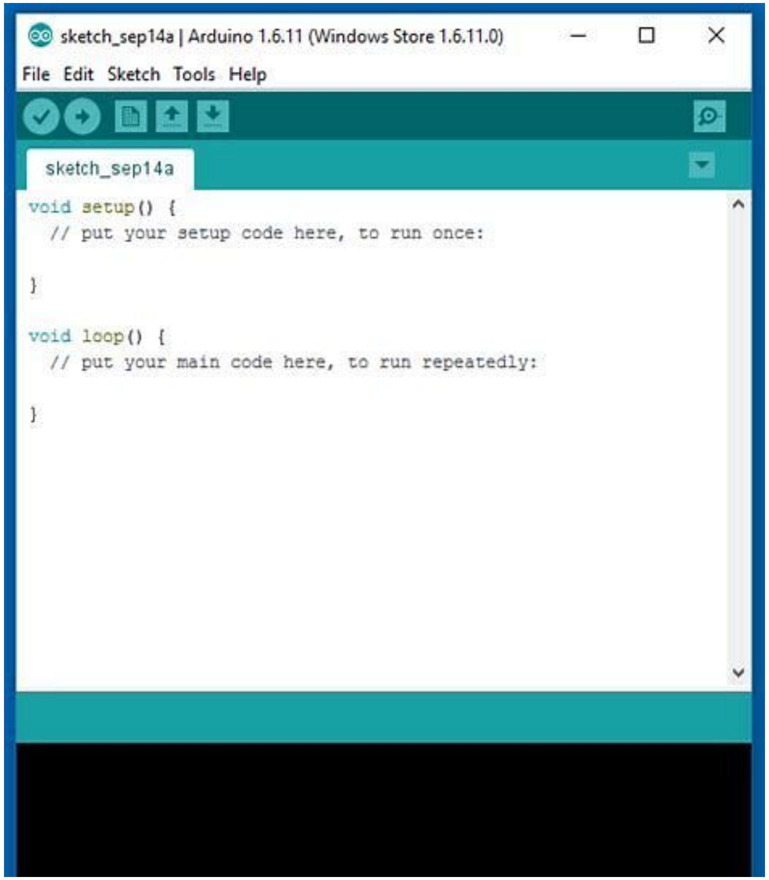
What is a Microcontroller?

- A microcontroller (small computer) with input and output ports for controlling electronics
- We are using an Arduino Uno, a specific model of microcontroller



How to control it?

- C++ code defines Arduino’s behaviour
- Can be written in the Arduino IDE
    - Compiled and pushed onto the device through USB


sketch_sep14a | Arduino 1.6.11 (Windows Store 1.6.11.0)
File Edit Sketch Tools Help
sketch_sep14a

```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

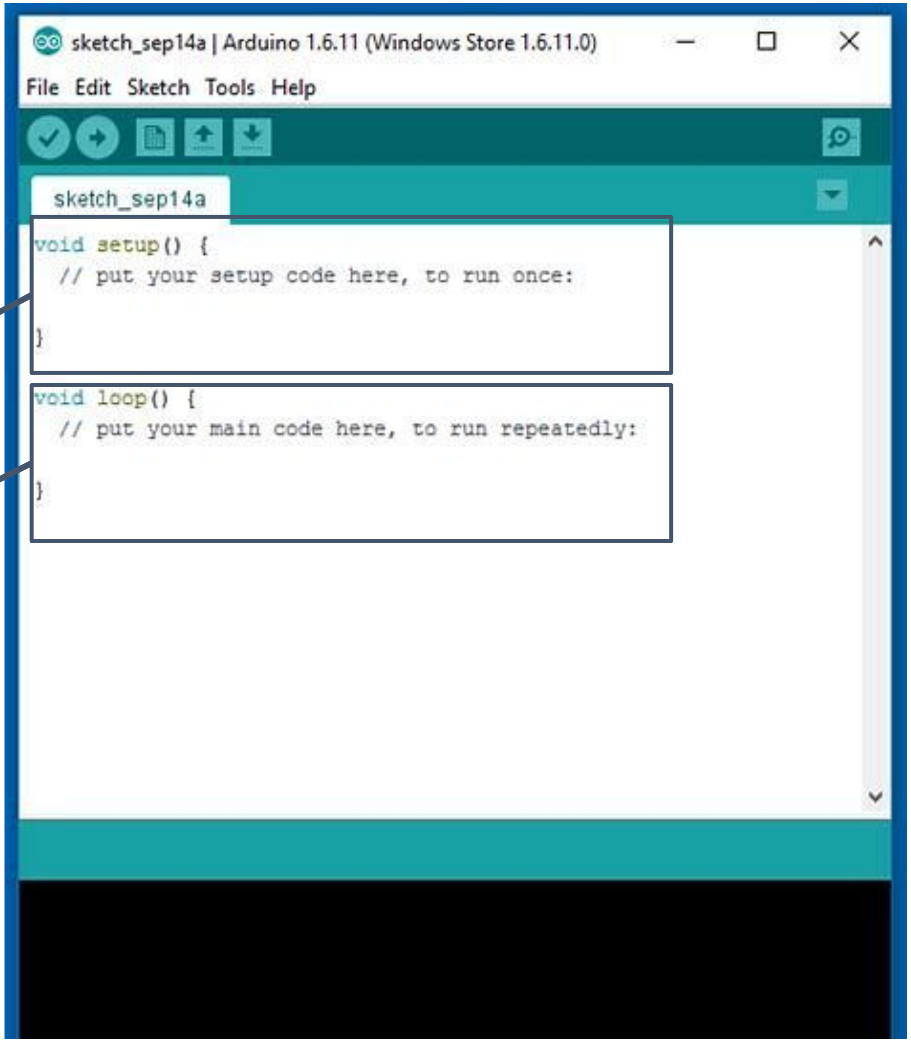
I/O Capabilities

- Digital (1 or 0) inputs and outputs
- Pulse Width Modulation (PWM) output
- Analog input (continuous range of voltage values)
- Support for various communication protocols
    - UART/Asynchronous (Serial)
    - I2C
    - SPI

Arduino Code

Arduino code can be written in three areas:

- inside the setup() function
- inside the loop() function
- outside either - usually variables or other functions


sketch_sep14a | Arduino 1.6.11 (Windows Store 1.6.11.0)
File Edit Sketch Tools Help
sketch_sep14a

```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

Arduino Code Cont.

Defining variables:

```
int a = 5;        |    bool b = false;
int a;            |    bool b;
a = 5;            |    b = true;
```

As seen above, there might be a need to separate the declaration and the initialization.

Arduino Code Cont.

If blocks:

```
if ( condition ) {

      //your code here to execute once
      //if condition is true

   }
```

While blocks:

```
while ( condition ) {

      //loop will repeat if condition is true

   }
```

Arduino Code Cont.

Basic Arduino-exclusive functions:

- pinMode(*pinNo* , *[INPUT/OUTPUT]*)
  - Initializes a digital pin as an input or output. Done in the setup() function

- digitalRead(*pinNo*)
  - Reads digital input pin and **returns a bool** to reflect state (true if high, false if low)

Arduino Code Cont.

More basic Arduino-exclusive functions:

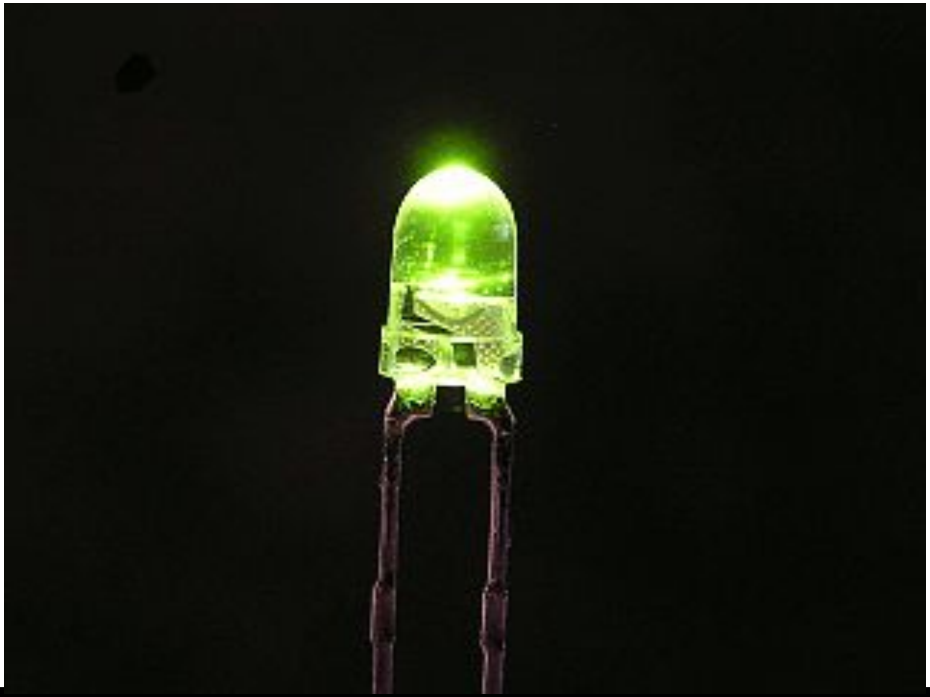
- digitalWrite(*pinNo* , *value*)
  - Writes *value* to specified pin number

- delay(*noOfMilliseconds*)
  - Arduino waits for the specified time

Lab A

Lab A - Premise

- Explore several ways to blink an LED with an Arduino microcontroller



Lab A - Step 1

- Download Arduino IDE
  - https://www.arduino.cc/en/Main/Software
- Configure IDE
  - Choose Board
  - Select on the top left: Tools -> Board -> Arduino/Genuino Uno

Lab A - Step 2

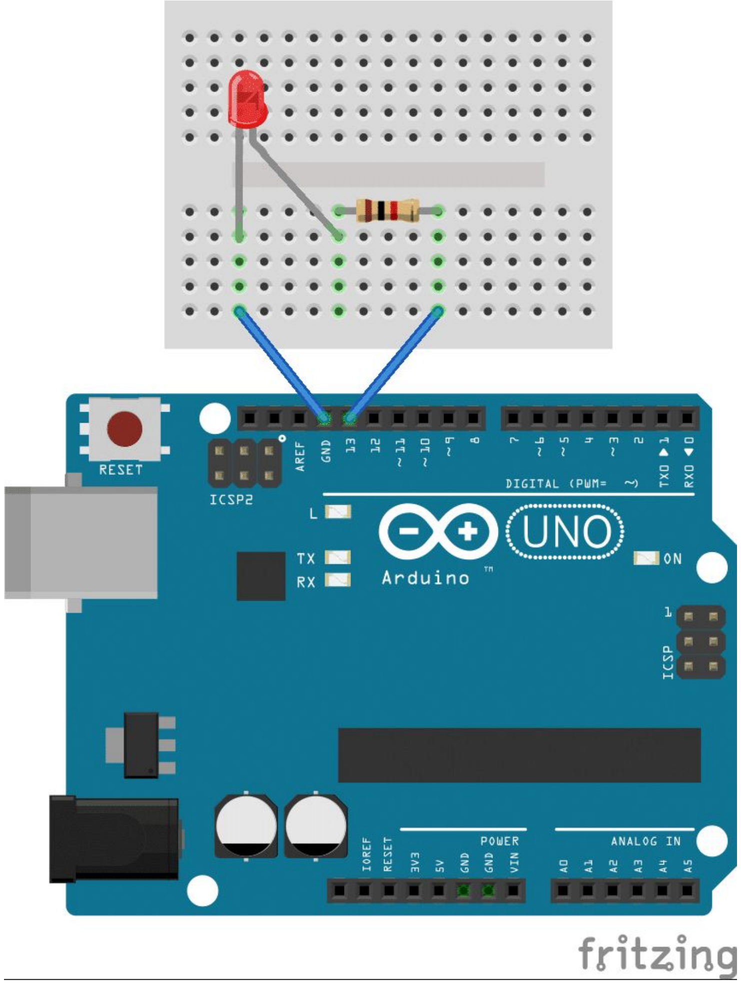
- Plug in Arduino with provided cable
- Choose Port
  - Select available port (if detected, name of board is indicated)
  - This is also under Tools

Lab A - Step 3

- Compile and upload the blinking LED example program onto the Uno
  - Files->Examples->Basics->Blink
  - Uses the onboard LED in the Arduino
  - You should see this LED blink on and off

Lab A - Step 4

- Modify the code of the example program to use an external LED on a breadboard
- Don’t forget to set which pin is controlling your LED



From circuitbasics.com

Lab B

Lab B

Choose a task to complete:
(in order of difficulty)

- Use the provided switch as a digital input to the Arduino to control the LED
- Create code to control 2 LEDs independently with the one corresponding switch
- Use the switch to select between two blink frequencies of an LED
- Create code to implement a binary counter with three LEDs, using a switch to iterate through each number (turning the switch on, then off, represents one count)