Lecture 2 CS3514 14/9/17

Blink | Arduino 1.6.7

File Edit Sketch Tools Help

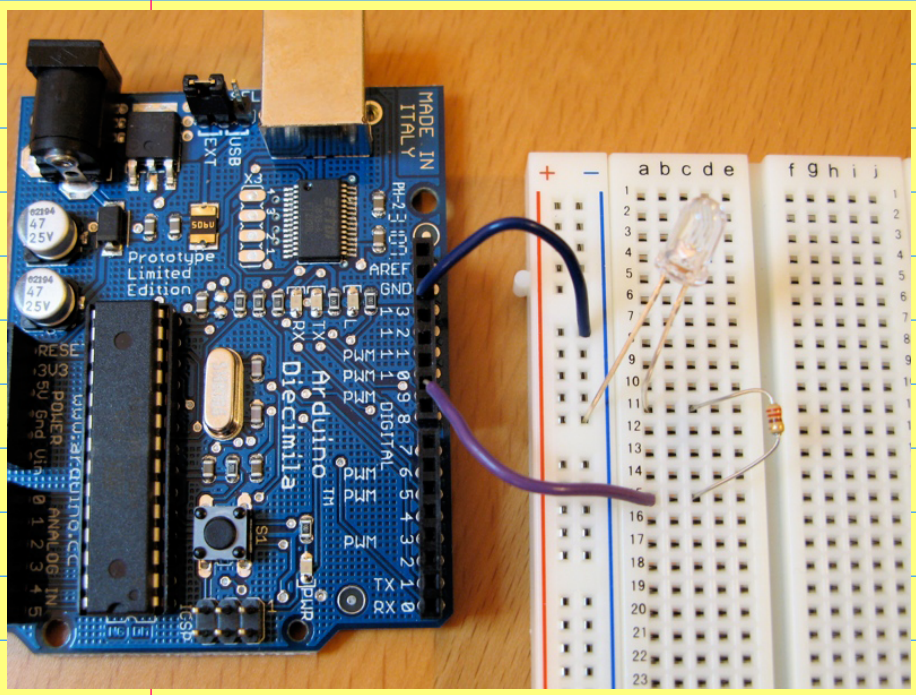
Blink §

```
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin 13 as an output.
  pinMode(13, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(13, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);              // wait for a second
  digitalWrite(13, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);              // wait for a second
}
```

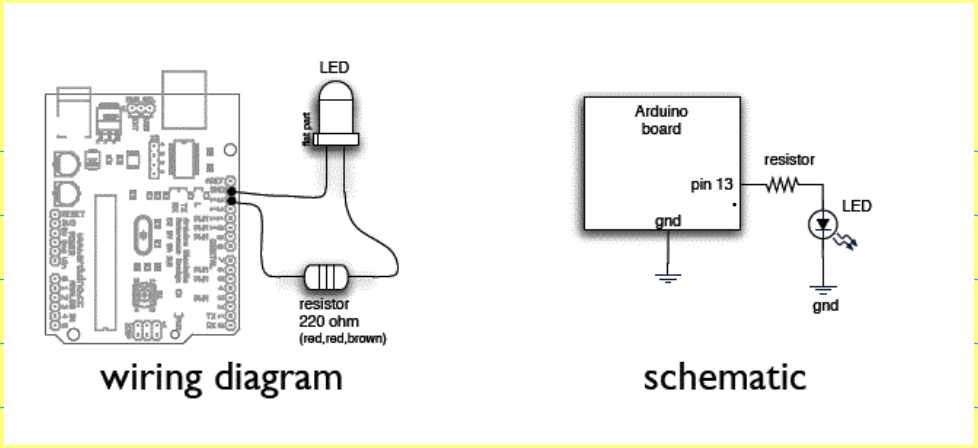
Compiling sketch...

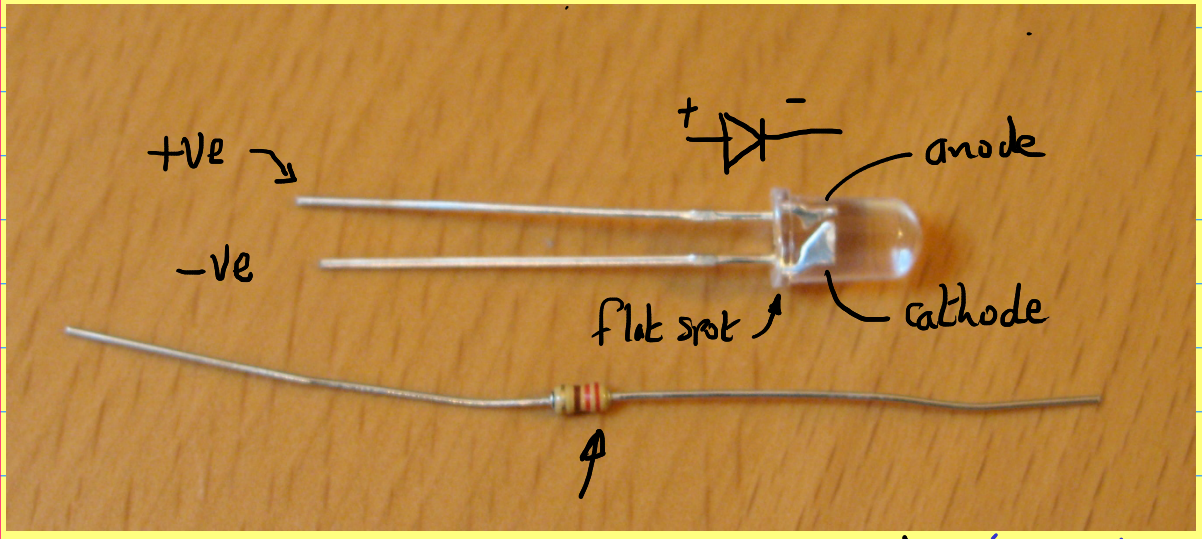
1 Arduino Duemilanove or Diecimila, ATmega328 on COM23



Hardware Setup for the Blink example

Note: in this photo, pin 10 is being used instead of pin 13.

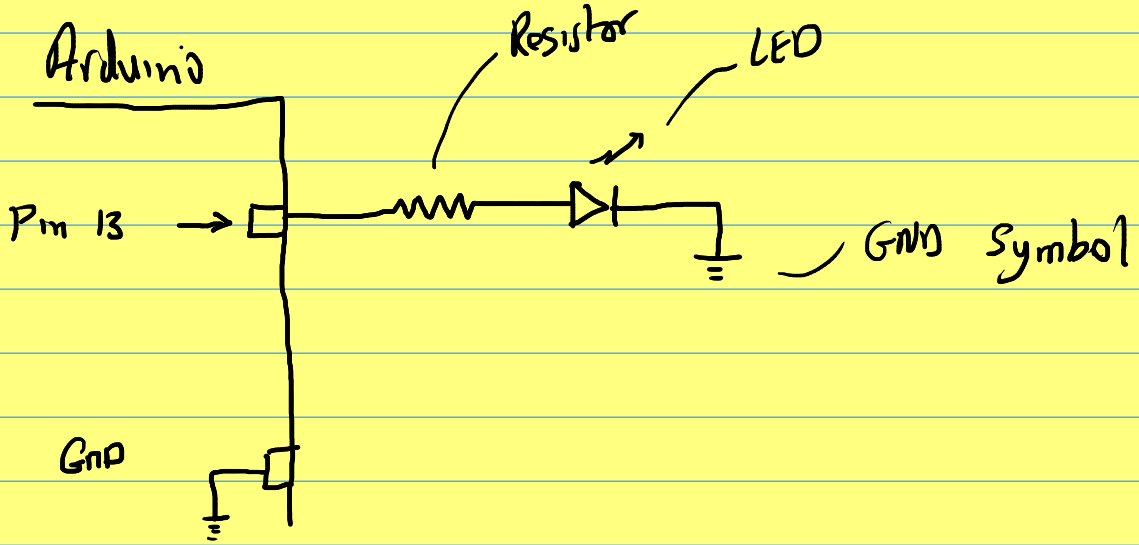




Resistance is measured in 'Ohms'

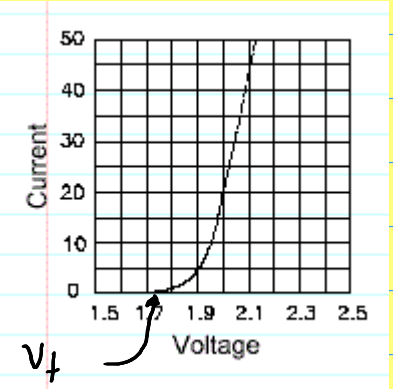
* LEDs are polarised, the way they are connected is important.
* Resistors are not polarised

Again the Circuit related to the blink example is:



What is the Current through the LED

Note: LED is 'non linear' — Does not obey ohms Law



— We don't vary current to change brightness (generally)

— When threshold voltage, V_f, is surpassed there is nothing to limit the current & the LED may be destroyed.

We should use a resistor in series with a LED to limit the current that can flow through the LED.

This is called a Current Limiting Resistor.

To calculate the current flowing in our circuit, we can use ohms Law

$V = IR$ (Voltage = Current x Resistance)

(Volts) = (Amps) x (Ohms)

$I = V/R$

$= 5V / 220\Omega = 0.027A = 27mA$

This is in safe range according to datasheet

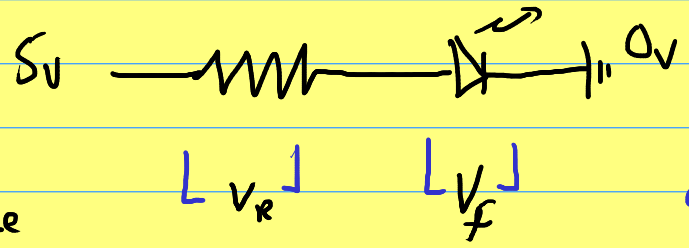
This calculation assumes no voltage drop across the LED
- not strictly true

In fact, there is a voltage drop across each component in an electronic circuit (even the wires). Each of these drops add up to the voltage of the source.

* The current flowing through components connected in series is the same in each component

So, even though our LED doesn't obey Ohm's law, we can still calculate the current flowing through it.

A more accurate picture is:



V_R = voltage drop across resistor.
V_f = voltage drop across the LED.

$$V_{source} = V_R + V_f$$

Series circuit => Current flowing through resistor = " " " LED

V_f is typically 2V for red LEDs (4V for blue)

$$V_R = V_{source} - V_f = 3V$$

Current through R:

$$I_R = V_R / R = 3V / 220\Omega = 13.6mA$$

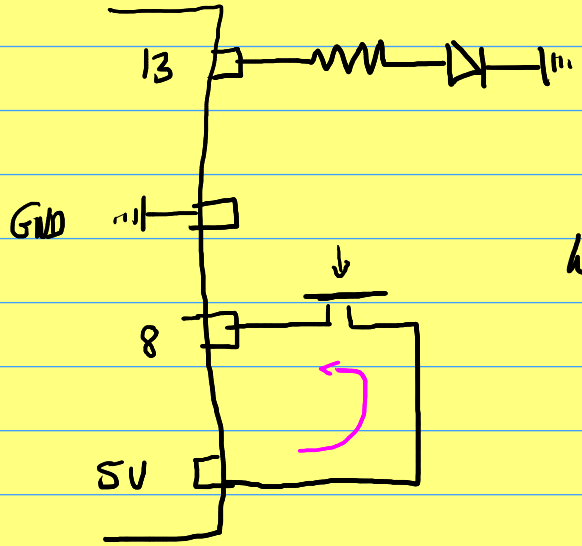
= Current through LED

We can also use this formula to calculate the resistance needed to protect a LED.

$$R = \frac{V_s - V_f}{I}$$, where I represents a safe LED current (from datasheet)

Digital Input

Connecting a Switch to the Arduino.



When the switch is pressed, the circuit is closed and 5V is 'seen' @ Input Pin 8.

sketch_sep22a

```
#define IN 8
#define OUT 13

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

  pinMode(IN, INPUT);
  pinMode(OUT, OUTPUT);
}

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

  if (digitalRead(IN)){
     digitalWrite(OUT, HIGH);
  }else{
     digitalWrite(OUT, LOW);
  }
}
```

When the switch is open, the voltage on pin 8 is questionable. Static electricity from the previous closing of the switch, or from the surrounding environment, may present a voltage to the pin and this could be read as a non-zero logic level!