introduction to Digital Electronics

Install the Arduino IDE 1.8.5 on your laptop if you haven’t already!
Electronics can add interactivity!

Any sufficiently advanced technology is indistinguishable from magic
- Arthur Clark.
Electronics can add interactivity!

Any sufficiently **well executed** technology is indistinguishable from magic - in the eyes of kids!
Electronics can be complex
Electronics can be complex
Avoid unnecessary complexity!
Let's look at a circuit!

Battery + Light Bulb
Let’s look at a circuit!

Battery

Power Source

+ 

Light Bulb

Load
Let's look at a circuit!
Let's look at a circuit!

\[ V = IR \]

Voltage = Current × Resistance

\[ 5 = I \times 330 \]

Current \( \approx 0.015 \text{A} \)
Let's look at a circuit!
Let's look at a circuit!
Let's look at a circuit!
Let's look at a circuit!

\[ V_I - V_{\text{ref}} = IR \]
Let's look at a circuit!
Let's look at a circuit!
Let's look at a circuit!

5v
330Ω
0v
Time for some LIGHT!

5v

330Ω

0v

LED
Light Emitting Diode
Time for some LIGHT!

Diodes conduct current primarily in one direction. Needs resistor!
Time for some LIGHT!

LED
Light Emitting Diode

5V

330Ω

0V

Longer is positive!
Now in real life!

5v

330Ω

0v

Power Rails
Now in real life!

5v

330Ω

0v

Power Rails
Now in real life!

5v

330Ω

0v
Now in real life!
Now in real life!
Now in real life!

5V

330Ω

0V

Momentary Switch
Modify your circuit

5v

330Ω

0v
A closer look...

VCC (also Vdd, V_high)
Voltage Common Collector
(usually your power supply!)

V_2 - V_{GND} = 1.4V

LED Forward Voltage Drop
≈ 1.4V  ≈ 0\Omega

Components have rules!
A closer look...

\[ V_{CC} \approx 5V \] (also \( V_{dd}, V_{high} \))

Voltage Common Collector
(usually your power supply!)

\[ V_2 \approx 1.4V \]

\[ V_{CC} - V_2 = I \cdot R \]

\[ 5 - 1.4 = I \cdot 330 \]

\[ I \approx 10 \text{ mA} \]

LED Forward Voltage Drop
\[ \approx 1.4V \approx 0\Omega \]

**Components have rules!**
A closer look...

\[ V_{CC} \sim 5 \text{V} \]

\[ R = 330 \Omega \]

\[ V_2 \]

\[ V_{GND} \sim 0 \text{V} \]

\[ I \approx 10 \text{ mA} \]
A closer look...

\[ V = IR \]

\[ I \approx 10 \text{ mA} \]

\[ I \approx \text{BIG NUMBER} \]
Digital vs Analog Circuits

Analog Circuits
Range of voltages
Usually requires math!

Digital Circuits
Usually 2 distinct voltages (high & low)
5v and 0v (roughly)
0110100001101001...
Using your Arduino!

A Microcontroller
.. or a small computer!
Has inputs and outputs you can control

Arduino Uno
Arduino Nano
Teensy
Using your Arduino!
Blink!
Blink!

Try making the colors alternate!
Pulse Width Modulation

PWM!

Digital is only 0v or 5v, so how do we get values in between?

LEDs can only be in 2 states- on or off!

PWM to make the LED seem ‘dimmer’

50% Duty Cycle

- 50% of the time on
- 50% of the time off

90% Duty Cycle

- 90% of the time on
- 10% of the time off
PWM

The Fading example demonstrates the use of analog output (PWM) to fade an LED. It is available in the File->Sketchbook->Examples->Analog menu of the Arduino software.

Pulse Width Modulation, or PWM, is a technique for getting analog results with digital means. Digital control is used to create a square wave, a signal switched between on and off. This on-off pattern can simulate voltages in between full on (5 Volts) and off (0 Volts) by changing the portion of the time the signal spends on versus the time that the signal spends off. The duration of "on time" is called the pulse width. To get varying analog values, you change, or modulate, that pulse width. If you repeat this on-off pattern fast enough with an LED for example, the result is as if the signal is a steady voltage between 0 and 5V controlling the brightness of the LED.

In the graphic below, the green lines represent a regular time period. This duration or period is the inverse of the PWM frequency. In other words, with Arduino's PWM frequency at about 500Hz, the green lines would measure 2 milliseconds each. A call to `analogWrite()` is on a scale of 0 - 255, such that `analogWrite(255)` requests a 100% duty cycle (always on), and `analogWrite(127)` is a 50% duty cycle (on half the time) for example.
Servo and Moving Parts

Rotary actuator that allows for precise control of linear position

Arduino friendly!

Built-in Library

- SIGNAL (Pin 6)
- POWER (5V)
- GND
What does the Arduino sense when it’s not connected to GND?

Value is *floating*!

Pull up resistor!
Button Inputs

int inPin = 8;  // pushbutton connected to digital pin 8
int val = 0;    // variable to store the read value

void setup()
{
    pinMode(inPin, INPUT);  // sets the digital pin 8 as input
}

void loop()
{
    val = digitalRead(inPin);  // read the input pin
}
Button Inputs

```cpp
int inPin = 8; // pushbutton connected to digital pin 8
int val = 0; // variable to store the read value

void setup()
{
    pinMode(inPin, INPUT_PULLUP); // sets the digital pin 8 as input
}

void loop()
{
    val = digitalRead(inPin); // read the input pin
}
```
Button Inputs

Digital Pin 8
LED strips

A strip of LEDs
Individually Addressable LED strips!

Objectively pretty neat!

Integrated Circuit & I²C communication

ws2812 RGB LED

Run on 5v and can be controlled with an Arduino!

Watch out for current!
Each color ~ 15mA, total 50mA on ‘white’.

Arduino max current ~1A
Projects for you to try

Easy:

**Servo Steering**
One button turns the servo 90 clockwise, the other turns the servo 90 counterclockwise!

**Light Switch**
Pressing a button turns the light on. Pressing the button again turns the light off!

Medium:

**Random Traffic Lights**
Have a RGB LED change to a random color every couple of seconds!

**Color Mixer**
Have 3 buttons that control the colors of a single RGB LED. Depends on which button is pressed, the light shows a different color!

Challenging:

**Night Light**
Using a photodiode, make a light turn on only when it’s dark!

**Reflex Game**
Make a simple reflex game out of 1 LED and 1 Button. When the light turns on, see how fast you can press the button. Make a simple scoreboard using the WS2812 Lights!
To be Continued…

Electronics **Part II!**

Communication

Wireless

Perf boards

Soldering

And more!

Take anything you’d like to keep with you (you can use these on your toys/sketch models)
introduction to Digital Electronics