Arduino Workshop Handouts

Basics of Programming and Electronics reference sheets

Anything you put in void setup() will be run once at the start at the program, so this is where you should set up your variables and set your pins to inputs/outputs. Anything in void loop () is run continuously as long as the arduino is connected to the power. This is where your main program should be written.

You set up a variable by first declaring what kind of variable it is, then the name of the variable and finally what the value of the variable is.

For example:

int LEDPin = 13;

means that you have a variable which is an integer, named LEDPin and has a value of 13.

The semicolon at the end is the equivalent of a full stop at the end of a sentence. Each statement you make must end in a semicolon. As a rule of thumb, if a line of code doesn't end in a bracket it should probably end in a semicolon. You also have to remember that **programming code is case sensitive**! If you use a capital letter for a statement then it might not be understood by the computer.

Some important statements you'll need are:

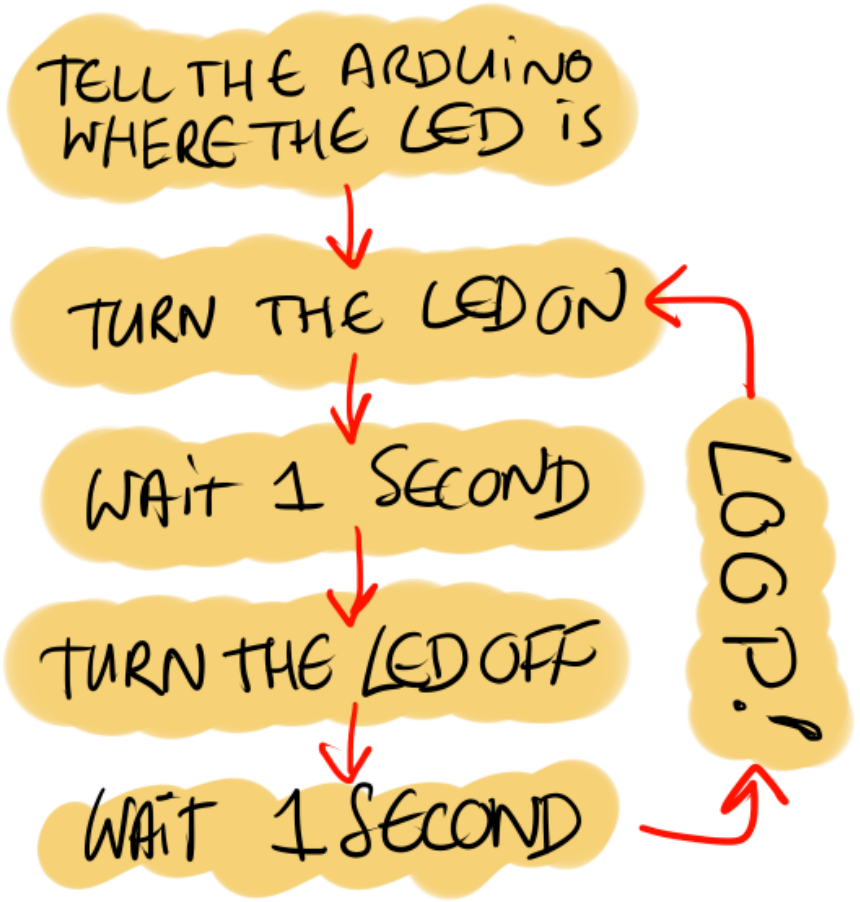
digitalWrite, the statement that tells the computer what to output and to where. If you have an LED set to a specific pin, you have to use digitalWrite to tell the arduino to send power to that pin in order to light it.

pinMode is how you tell the arduino that a pin is either going to be outputting information or inputting information. Either the arduino then will be sending information to the pin or it will be looking for information being sent from the pin, but never both.

delay does exactly what it looks like it should do – the computer will pause for a while before it starts the next step of the program. delay is always set in milliseconds, so delay(1000) will cause the program to pause for a second before continuing onto the next step.

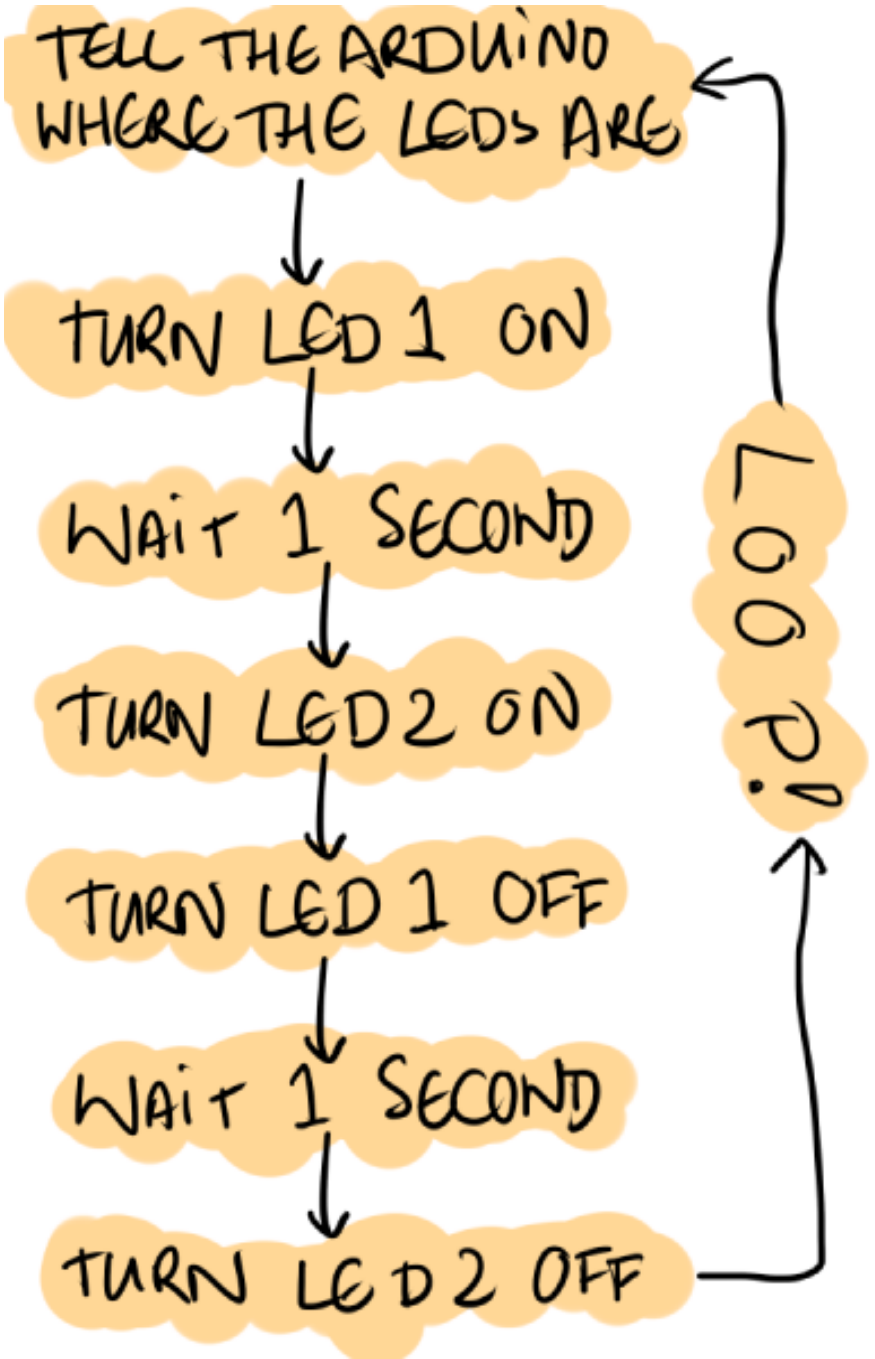
Components in a circuit are either digital or analog – if it works like a lightswitch with on and off and only set steps like that, it's a digital component. If it's more like a dimmer switch, it's analog. This will affect how it's programmed, but it's easy enough to work out. You only need to know how to do digital signals for this project. Any programs with multiple components connected to the same pin must use either a digital or analog signal for them, but if you have components connected to different pins, then they can be different types.

Usually when you're starting to build a program, it helps to start with a flowchart. This can then be turned into the code easily without losing track of how the program was meant to work. For example, you can see how this flowchart turned into this program where the LED was connected to pin 13.




```
void setup() {
        int LEDPin = 13;
        pinMode(LEDPin, OUTPUT);
        }


void loop() {
        digitalWrite(LEDPin, HIGH);
        delay (1000);
        digitalWrite(LEDPin, LOW);
        delay(1000);
        }
```



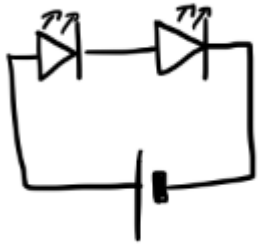
Another example of this process is shown here, where two lights are connected to different pins so that they can be controlled separately to each other. In this example, one LED is connected to pin 9 and the other to pin 10 on the arduino board.

```
void setup() {
        int LEDPin1 = 9;
        int LEDPin2 = 10;
        pinMode(LEDPin1, OUTPUT);
        pinMode(LEDPin2, OUTPUT);
        }
```

```
void loop() {
        digitalWrite(LEDPin1, HIGH);
        delay(1000);
        digitalWrite(LEDPin1, LOW);
        digitalWrite(LEDPin2, HIGH);
        delay(1000);
        digitalWrite(LEDPin2, LOW);
        }
```

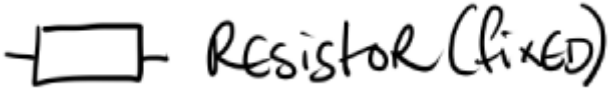
Arduinos can run at either 5V or 3.3V depending on what pin you connect your circuit to. We're going to run the motors at 5V so they're getting enough voltage to run.

You can connect components to a circuit in two ways – parallel or in series. Depending on how you connect the components, you can affect how much power the circuit needs. For example, if you connected two LEDs that needed 3V to run like this circuit on the left,

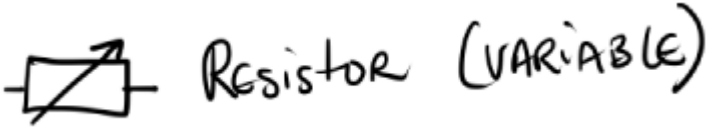


you would need 6V to run the circuit. Since the LEDs are **in series** (one after the other), they need twice as much power to run.

If you connected them **in parallel** like this circuit on the right, You would only need 3V to run the circuit. Connecting components in parallel means they can share the power available.

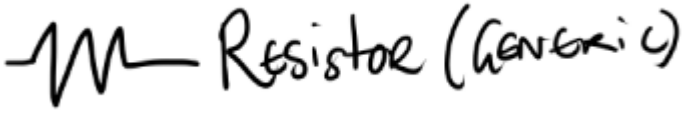


is a resistor with a fixed resistance.

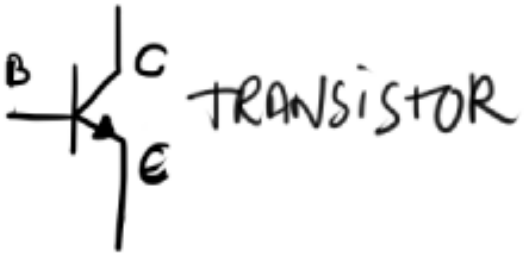


is a resistor with a varying resistance.

These can either be changed as needed if they have a slider or something similar, or sometimes the resistance can be set at the start and is then fixed.

Any kind of resistor can be shown by the symbol 

The advantage of a resistor is that it can help set the appropriate power level to run some components that otherwise would be fried, but also they can control the level of power being drawn by other components. LEDs for example will always draw as much power as they can, so without a resistor then the battery life of an LED circuit is greatly reduced.

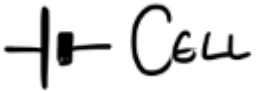


Transistors are tiny chips that can work as **amplifiers or switches**. They can either be used to amplify the strength of a weaker electrical signal and make it stronger. This was first used in hearing aids where a tiny microphone would pick up the sound around it, send that to a transistor as an electrical signal where it would be amplified and then fed back out through a louder speaker.

These can also be used as switches – a smaller current reaching one part of the transistor will open the circuit for a larger current. This can also be used to keep a set resistance level and stop some components from burning out.

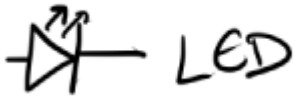
The transistors we're using today are junction transistors, so they have three pins. One pin is the **collector**, where the signals are first sent to, then the next is the **base**, which grounds the circuit and helps to deal with excess current and the last is the **emitter**, where the amplified signal is sent from.

Transistors have a wide range of possible ways to wire them, so the collector, base and emitter pin order may vary depending on what kind of transistor you are using.



Batteries or **cells** are shown with this symbol. The longer line is the positive terminal, the shorter line is the negative terminal.

Bulbs are shown with this symbol.



Other types of lighting use different symbols, for example, the symbol for an **LED** (Light Emitting Diode) is this, with the triangle pointing which way the current flows. An LED usually has a long and short pin to connect it to a circuit– the longer leg is the positive end.