

CoderDojo Bray : LED Sushi

The basic Arduino program to start you off with the LED Dojo Sushi challenges is provided: blink.

“Blink” is a super simple program which flashes the small LED on your Arduino UNO board on and off once per second.

Using this “Blink” program as a base, we have a series of challenges to go deeper into controlling the LEDs and learning some programming techniques.

There is also a variation on the simple “Blink” program, which does the same thing, but in a different way. “Blinknodelay” blinks your LED but does not use the `delay()` function. This shows you that there is nearly always more than one way to do something: which way you pick depends on your preference, or what else you need your software to do, or which seems better to you.

That’s the best thing about software you write yourself: you can do it **your way**.

If you can complete these Challenges, then you have mastered controlling LEDs with your Arduino. Congratulations!

Challenge #0 – Getting Started

Level

Beginner : You need to complete this challenge to get started with working with your Arduino.

Challenge

- Get the Arduino IDE installed on your computer
- Build the “Blink” sketch
- Download it to the Arduino and check that the LED beside pin 13 blinks on and off once per second

Extra Credit

- Explain to a mentor how to build the sketch without downloading it
- Look up the special Arduino function calls made in the Arduino Reference Guide

Aims

Get the Arduino IDE installed on your computer and correctly communicating with an Arduino board connected to it over USB.

When you have this working, your environment is up and running and you are ready for the rest of the LED Sushi challenges.

Resources

An Arduino, USB cable, and a laptop.

Notes for Mentors

See the coderoj.com site for troubleshooting information, if the Arduino IDE will not communicate with the Arduino board.

Challenge #1 - Blinking slower

Level

Beginner

Challenge

- Make a copy of the “Blink” sketch
- Change it so that it blinks the LED off and on once per 3 seconds, and not once per second

Extra Credit

- Make the LED stay on for 4 seconds and off for 2 seconds

Aims

Show you can edit and modify the code, build it, and see your changes working.

Resources

An Arduino, USB cable, and a laptop.

Notes for Mentors

If basic communication is working, this one should be easy.

Challenge #2 - Blinking faster

Level

Beginner

Challenge

- Make a copy of the “Blink” sketch
- Change the sketch so that the LED is on for 100 milliseconds and off for 50 milliseconds

Extra Credit

- Change the sketch so that the LED flashes 5 times per second
- Find out how fast you have to switch the LED off and on per second before you can no longer tell that the LED is blinking at all, and it just looks like it's on all the time.

Aims

Learn about how fast your vision can see the light flashing. Be able to make changes to the software and see those changes in action.

Notes for mentors

Explain that the speed of flashing is related to the speed of a film projector, the speed that TV and LCD monitors refresh at.

Challenge #3 – Move your LED to a breadboard

Level

Intermediate: includes hardware

Challenge

- Make a copy of the “Blink” sketch
- Get a breadboard, a LED and some jumper wires from the Arduino supplies
- Build a LED circuit on the breadboard and make it flash on and off with the Blink sketch

Extra Credit

- Change the circuit so that the LED is connected to PIN 12 instead of PIN 13
- Now change the code for your sketch so that the LED flashes on and off when connected to PIN 12.

Aims

Build a simple circuit on an external breadboard, and control it from the Arduino. Then modify that circuit so that the LED is controlled by a different PIN, and modify the sketch to match that PIN.

Challenge #4 – Flashing 3 LEDs at the same time

Level

Intermediate – easy if you have completed Challenge #3

Challenge

- On your breadboard, build a circuit with 3 LEDs and connect this up to three different outputs on the Arduino
- Make a copy of the Blink sketch
- Change the sketch so that all three LEDs flash on and off at the same time

Extra Credit

- Change the sketch so that the LEDs flash on and off at different times

Aims

Build a circuit to drive more than one LED, and control them all.

Challenge #5 – Flashing 3 LEDs in sequence

Level

Advanced

Challenge

- On your breadboard, build a circuit with 3 LEDs and connect this up to three different outputs on the Arduino (If you have done Challenge #4 already, this is the same circuit)
- Make a copy of the Blink sketch
- Let's call our three LEDs: LED1, LED2, and LED3
- Change your sketch so that the LEDs flash like this:
LED1 LED2 LED3 LED1 LED2 LED2 LED3 ...

Extra Credit

Use a function to control the LED: pass in the LED to control as a parameter to the function

Aims

Control LEDs in a pattern.

Mentor notes

Use this challenge to introduce

Challenge #6 – Sweeping the LEDs back and forth

Level

Advanced

Challenge

- On your breadboard, build a circuit with 4 LEDs and connect this up to four different outputs on the Arduino.
- Make a copy of the Blink sketch
- Let's call our four LEDs: LED1 LED2 LED3 LED4
- Change your sketch so that the LEDs flash like this:
LED1 LED2 LED3 LED4 LED3 LED2 LED1 LED2 LED3 LED4

Extra Credit

Use a for() loop together with a function() to control the LEDs

Aims

Introduce for() loops to make the code neater and clearer to understand

Notes for Mentors

Explain how for() loops and functions() can simplify the code and reduce the amount of code you have to write

Challenge #7 – S.O.S

Level

Advanced

Challenge

- On your breadboard, build a circuit with a single LED (or you can use a circuit with >1 LED and just control one of them)
- Make a copy of the Blink sketch
- The international distress call in Morse code is SOS. In Morse code, this is three short flashes, three long flashes, and three short flashes, repeated forever
- Change your sketch so that it flashes the Morse code SOS signal

Extra Credit

Find the description of the Morse code symbols on the internet and make a different pattern: explain that pattern to a mentor

Aims

More advanced LED control: introducing Morse code.

Notes for Mentors

Kids will need to look up Morse code on Wikipedia or similar

Challenge #8 – Your Name in Morse Code

Level

Advanced

Challenge

- On your breadboard, build a circuit with a single LED (or you can use a circuit with >1 LED and just control one of them)
- Make a copy of the Blink sketch
- Change your sketch so that it flashes out your first name in Morse code. You can look up the Morse code for your name on the Internet.

Extra Credit

- How would you make sure that upper and lower case letters generate the same Morse code symbol (since Morse code has no upper or lower case)

Aims

Introduce more complex Morse code, and the notion of converting between upper and lower case characters.

Notes for Mentors

Encourage the use of a function to output a Morse code letter, and use a data structure (an array, maybe) to hold the letters to output.

Challenge #9 – Morse Code

Level

Very Advanced

Challenge

- On your breadboard, build a circuit with a single LED (or you can use a circuit with >1 LED and just control one of them)
- Make a copy of the Blink sketch
- Make a sketch which can send a message in Morse code. The message can be a fixed string which is built into your sketch. The sketch should know the complete Morse code “alphabet”, which you can look up on the Internet.

Extra Credit

Extra credit here given for program structure. The program should have a definition of the Morse code alphabet in a data structure, and be able to look up the correct output symbol for any letter in the fixed string, so that you can just change the string and get a different output pattern

Notes for Mentors

This is quite a complex program, and a Ninja who can successfully complete this has a very good grasp of basic programming.

Challenge #10 – RGB

Level

Intermediate

Challenge

- On your breadboard, build a circuit with one Red, one Green and one Blue LED, close together
- Make a copy of the Blink sketch
- Change the sketch so that it uses PWM to control the level of the Red, Green and Blue LED.

Extra Credit

- See if you put a piece of paper over the LEDs can you set the colour you see to Yellow.
- Can you set the colour to Purple?

Aims

Introduce PWM: the kids can look this up on Wikipedia, or have a mentor or more advanced Ninja explain this to them.

Mentor Notes

LED colour mixing will probably not work too well with individual LEDs. To have the best chance of it working, make sure the LEDs are as close together as possible, and use a “diffuser” (piece of paper, handkerchief ...) to try to diffuse the light from the LEDs.

Challenge #11 – RGB LED Strip

Level

Intermediate

Challenge

- On your breadboard, build a circuit which uses transistors to control an RGB LED strip. You will need to get a mentor to get the right parts for you, and show you the idea of the circuit.
- Make a copy of the Blink sketch
- Change the sketch so that it continually changes the colour of the RGB LED strip, gradually changing through all the colours on the colour wheel. You will need to use PWM (Pulse Width Modulation) for this challenge: you can look up PWM for Arduino. Ask the other Arduino ninjas for help with PWM if you can't find it – and if they don't know, then come and find a mentor.

Extra Credit

- Cycle the colours of the LED strip to match your favourite team, or to match the colours of the Irish flag.

Aims

Introduce control of the LED strips, which can be used for some very cool visual effects.

Notes for Mentors

The LED strip has to be powered using an external 12V power supply: help the kids set this up. The Arduino can be powered by the same power supply : you need to be careful wiring up the power supply to that you don't damage something connected to the Arduino (or the Arduino itself)