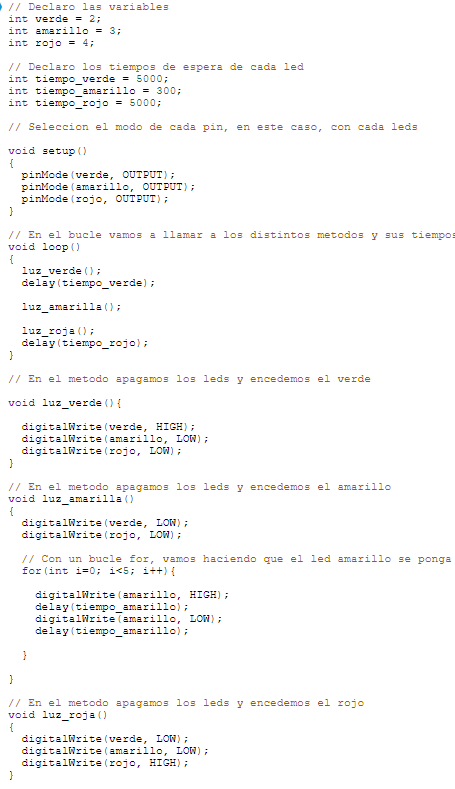
***Practica 5: Semáforo y Proyecto Libre***

*David Martínez Díaz GII-ADE*

* **Proyecto: Semáforo.**

***Código final:***



***Explicación del código:***

Primero declaras las variables “amarillo, verde, rojo”, que son los dispositivos leds igualándolos a sus respectivos pines a los cuales van enchufados.

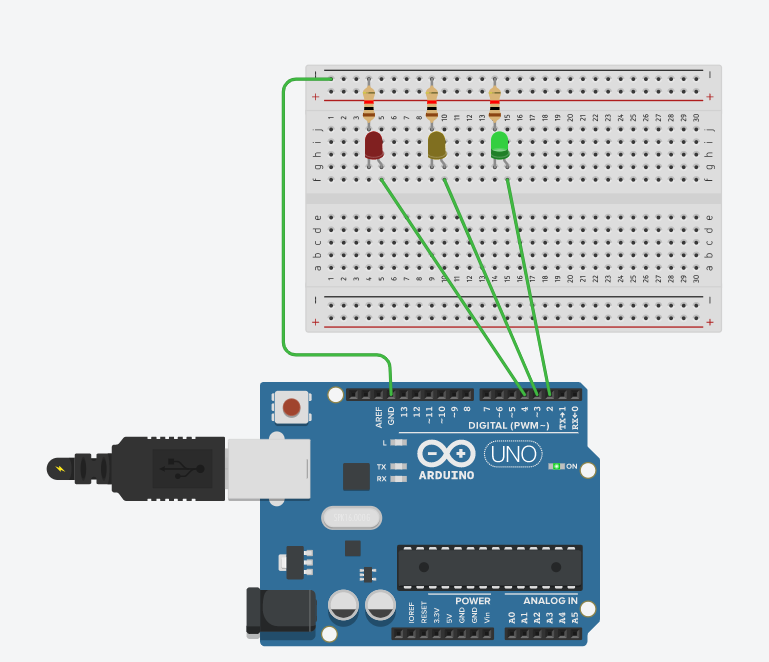
Por otro lado, declaramos las variables de tiempos de espera para cada uno de los leds, por ellos son llamados “tiempo\_verde, tiempo\_rojo, tiempo\_amarillo”.

Ahora empezamos con la función setup(), este es la función que se llama nada más comenzar la simulación, ahí definimos que, en los pines de los leds, se forman en modo OUTPUT, para que cuando reciban cierta señal actúen y se iluminen entre otras más.

Después en la función loop(), la cual es la que se ejecuta en un bucle infinito, llamamos a distintas funciones externas, como son luz\_verde, luz\_roja, luz\_amarillo, con sus respectivos tiempos de espera a través de la función delay().

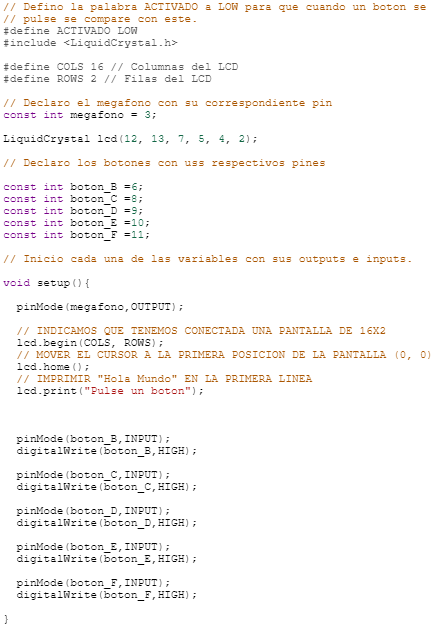
Una vez dentro de los métodos comentemos la función “luz\_verde” y la función “luz\_roja”, donde simplemente escriben el modo en el que se encuentran el pin, haciendo que se enciendan su respectivo led y se apaguen los otros dos.

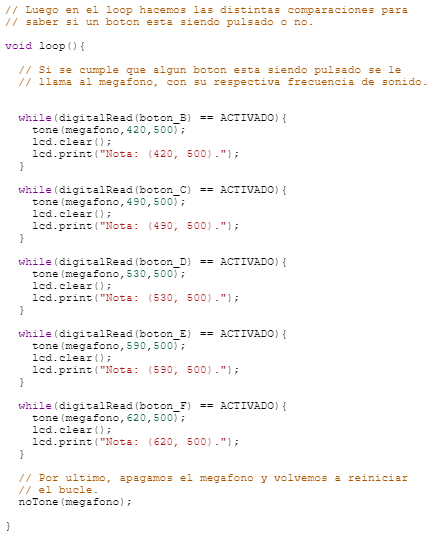
En la función luz\_amarilla se presenta una variante, en esta se realiza un bucle for donde se van encendiendo y apagando la luz, para que de ese efecto intermitente característicos de los semáforos.



* **Proyecto: Piano.**

***Código final:***

******

******

***Explicación del código:***

En primer lugar, para este proyecto nos va a hacer falta la libre LiquidCrystal.h, la cual nos permitirá utilizar la pantalla LCD a nuestro gusto.

Además, he definido una constante “ACTIVADO = LOW”, porque más adelante lo utilizare para comprobar ciertas condiciones, “COLS = 16”, una constante que nos diga el número de columnas que tendrá la pantalla LCD y “ROWS = 2” otra constante para las filas de la pantalla LCD.

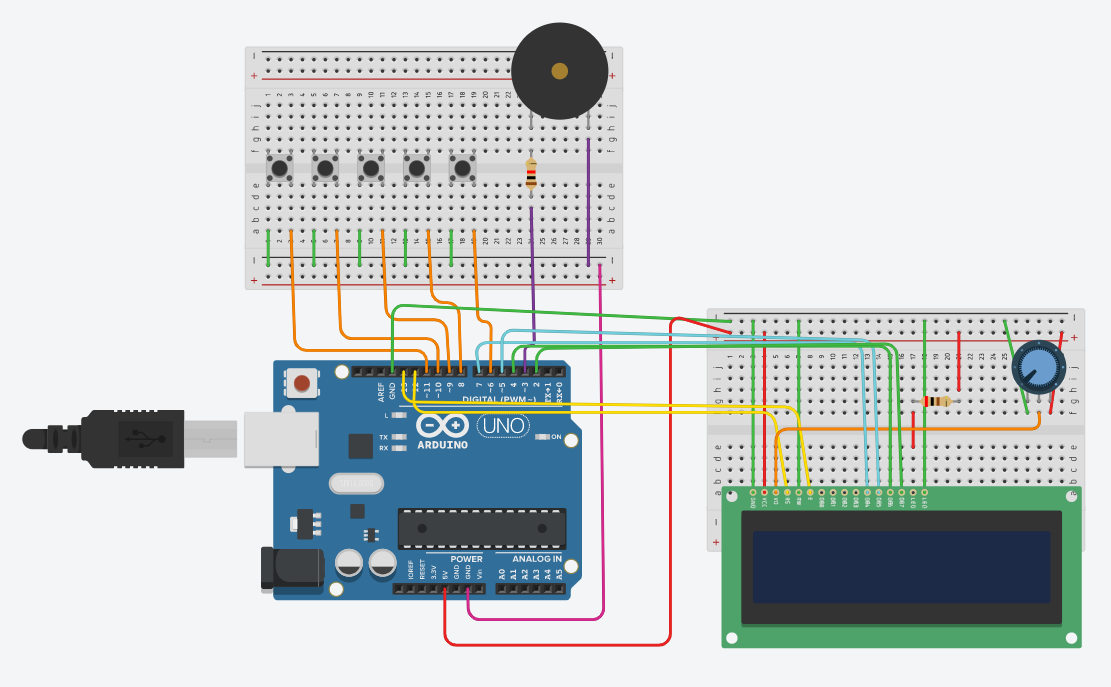
Luego definimos, los dispositivos que utilizaremos, en primer lugar, el megáfono, asignándole el pin 3, luego la pantalla LCD con respectivos pines (12, 13, 7, 5, 4, 2) y por último los botones que vamos a utilizar (B = 6, C = 8, D = 9, E = 10, F = 11).

En la función de inicio setup(), vamos a poner en modo OUTPUT el megáfono o timbre, iniciamos el LCD con sus respectivas filas y columnas, ponemos el cursor en el (0,0) y escribimos “Pulsa un botón” para explicar que se debe hacer en el circuito.

Entonces declaramos los botones a modo INPUT, y los definimos a HIGH, para observar que los botones no están pulsados al principio.

Luego en el bucle loop(), realizamos una serie de bucles while(), donde cada vez que este presionando un botón ya sea (B,C,D,E,F) con la condición [ digitalRead(boton\_B) == ACTIVADO ], utilizando la función tone(), con distintos valores y poniendo por pantalla la nota que se está tocando, por ejemplo, ("Nota: (420, 500).").

Por último, reseteamos el megáfono a través de la función noTone() y volvemos a empezar el bucle.



Enlaces:

<https://www.tinkercad.com/things/hrmChBVkJTL>

<https://www.tinkercad.com/things/k007i1vDsAB>