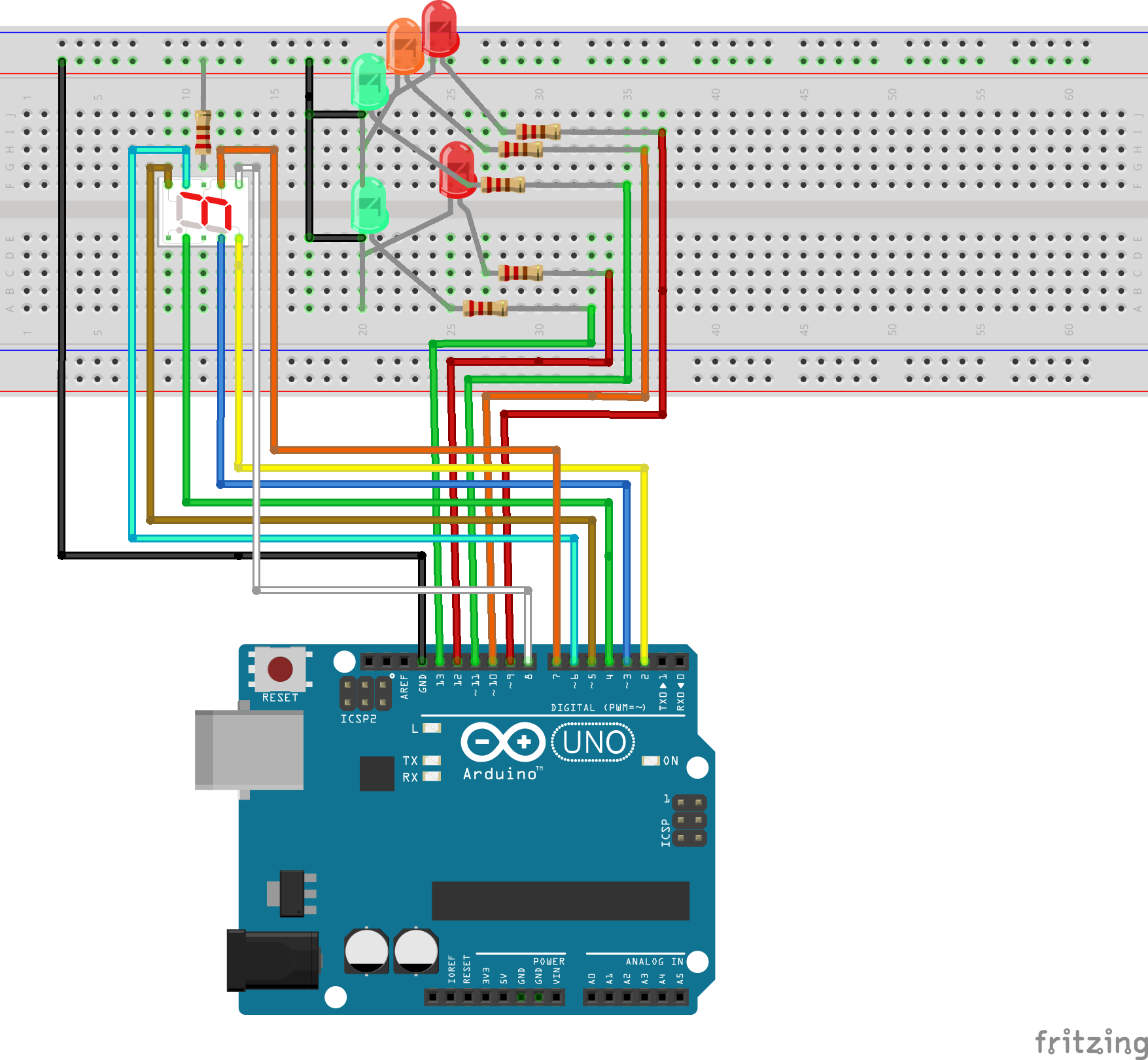
**Funcionamiento de un semáforo de coches y peatones con tiempo de espera mostrado en un display**

* **Descripción:** Se trata de realizar un circuito para simular el funcionamiento de un semáforo “real”, donde se pueda mostrar en un display el tiempo que han de esperar los peatones para que se ponga en verde y a su vez, una vez se ha puesto en verde para los peatones, se mostrará el tiempo que tienen para poder cruzar antes de que se ponga en rojo.
* **Objetivos:** Este ejercicio pretende que el alumno asocie ideas previas y reutilice código utilizado en proyectos anteriores. El objetivo es el de consolidar conocimientos en materia de electrónica digital y el uso de sentencias de control condicional para el control de los distintos estados del semáforo.
* **Ámbito de aplicación:** Ejercicio más avanzado enfocado al ámbito de la informática y la tecnología, aunque en este caso se necesitan ciertos conocimientos de programación (estructuras de control, funciones y matrices (arrays)).
* **Nivel:** Ejercicio de nivel intermedio-avanzado, tanto para el montaje del circuito eléctrico como en la parte de programación, sobre todo si se quiere entender completamente toda su lógica.
* **Conocimientos previos:** Antes de realizar este ejercicio los alumnos/as han debido de realizar algunos ejercicios básicos con Arduino, donde hayan activado salidas digitales, y hayan hecho uso de funciones propias a la hora de programar, con uso de variables y sentencias de control condicional. Además, es conveniente estar también familiarizado con el manejo de un display de 7 segmentos, por todo ello, se recomienda haber realizado los dos ejercicios planteados anteriormente, llamados *Semaforo\_tres\_leds* y *Funcionamiento\_led\_7\_segmentos*.
* **Diagrama o Esquema:**



* **Material necesario:**
* 1 x Arduino UNO
* 1 x Protoboard
* 1 x Display de 1 dígito
* 5 x leds (2 rojos, 2 verdes y 1 amarillo)
* 6 x resistencias de 330Ω.
* Cables para conectar todo
* **Desarrollo de la práctica:** Los pasos a seguir serían los siguientes:

1. Montaje eléctrico del sistema de leds que van a componer el semáforo
2. Desarrollo de programa simple que testee que los leds están conectados a las salidas esperadas
3. Montaje eléctrico de del display
4. Desarrollo de un programa simple para testear que los segmentos del display están conectados a las salidas esperadas
5. Desarrollo del programa completo. Una vez testeado que los elementos del circuito funcionan correctamente por separado, se pasa a desarrollar el programa que permite controlar la lógica de un semáforo para coches y peatones mostrando los tiempos de espera/paso en el display.

* **Código:** código fuente empleado en la práctica.

/\*\*\* Semáforo con display \*\*\*/  
  
//\*\* Definiciones \*\*//

byte Digit[10][7] = // Arduino UNO va muy justo de memoria. Por eso lo

{ // definimos como byte y no como int

{ 1,1,1,1,1,1,0 }, // 0

{ 0,1,1,0,0,0,0 }, // 1

{ 1,1,0,1,1,0,1 }, // 2

{ 1,1,1,1,0,0,1 }, // 3

{ 0,1,1,0,0,1,1 }, // 4

{ 1,0,1,1,0,1,1 }, // 5

{ 1,0,1,1,1,1,1 }, // 6

{ 1,1,1,0,0,0,0 }, // 7

{ 1,1,1,1,1,1,1 }, // 8

{ 1,1,1,0,0,1,1 } // 9

};

//Se definen los pines de los leds

//Para coches

byte rojoC=9;

byte amarilloC=10;

byte verdeC=11;

//Para peatones

byte rojoP=12;

byte verdeP=13;

void setup()

{

//inicializa los pines del display (2 - 8)

//inicializa los pines del semáforo coches (9,10,11)

//inicializa los pines del semáforo peatones (12,13)

for (int i= 2; i<14; i++){

pinMode(i, OUTPUT); //de 2 a 13 todo configurado como salidas digitales

}

}

void loop()

{

//Sección para gestionar semáforo de coches

verdeCoches();

for (int seg=9;seg>0;seg--){

escribeDisplay(seg);

if(seg==3){

amarilloCoches();

}else if (seg==1){

rojoCoches();

}

delay(1000);

}

//sección para gestionar semáforo de peatones

verdePeatones();

for (int seg=5;seg>0;seg--){

escribeDisplay(seg);

if(seg==1){

rojoPeatones();

}

delay(1000);

}

}

void verdeCoches(){

digitalWrite(rojoC , LOW) ;

digitalWrite(amarilloC , LOW) ;

digitalWrite(verdeC , HIGH) ;

rojoPeatones();

}

void amarilloCoches(){

digitalWrite(amarilloC , HIGH) ;

digitalWrite(verdeC , LOW) ;

}

void rojoCoches(){

digitalWrite(rojoC , HIGH) ;

digitalWrite(amarilloC , LOW) ;

digitalWrite(verdeC , LOW) ;

}

void verdePeatones(){

digitalWrite(rojoP , LOW) ;

digitalWrite(verdeP , HIGH) ;

}

void rojoPeatones(){

digitalWrite(rojoP , HIGH) ;

digitalWrite(verdeP , LOW) ;

}

// Función que representa en el Display el número que se le pasa por parámetro

void escribeDisplay(int N)

{

for (int i= 0 ; i<7 ; i++)

{ int valor = Digit[N][i] ;

int pin = i+2;

digitalWrite(pin , valor) ;

}

}

* **Enlaces de interés:**
  + Tutorial para el desarrollo de un semáforo de 3 leds

<https://openwebinars.net/blog/tutorial-arduino-ejemplo-semaforo/>

* + Tutorial en el que está basado este ejemplo donde se explica en detalle cómo funciona un display de 7 segmentos. <http://www.prometec.net/display-7segmentos>