|  |  |
| --- | --- |
| |  | | --- | | **INGENIERÍA EN SISTEMAS COMPUTACIONALES**  Ejercicio Encender - Apagar |   **TECNOLÓGICO DE ESTUDIOS SUPERIORES DE CHALCO** |



|  |  |
| --- | --- |
| |  | | --- | | **INTEGRANTES**  Arfaxad Zadot Aguilar Jiménez |   **SISTEMAS PROGRAMABLES** |

**DESCRIPCIÓN**

**INTRODUCIÓN**

Hace unos años apareció un proyecto libre, llamado **Arduino, el** cual facilitaba el acceso a esta clase de dispositivos a los estudiantes, dado que es una placa basada en open hardware (su diseño es libre y es posible reproducirlo por cualquier persona).

Inicialmente, la placa se conectaba a través de un puerto USB para programarlo (normalmente se hace en base a su IDE) La programación de Arduino no se realizaba a bajo nivel con ensamblador como muchos microcontroladores (a partir de ahora los llamaremos PICs), sino que se realiza con un lenguaje más comprensible por la mayoría de nosotros, C/C++, cestos elementos, un programador que no sabe de PICs, sería capaz de programar el Arduino en poco tiempo.

**OBJETIVO**

Desarrollar un sistema de tránsito en Arduino utilizando 2 leds rojos, 2 leds verdes y 2 leds amarillos.

**MATERIALES**

* 1 computadora con IDE Arduino y Proteus 8.
* Protoboard.
* 1 resistencias de 330K.
* 1 leds para cada color [Rojo, Verde, Amarillo].
* Modulo Ethernet
* Arduino Uno.

**DESARROLLO DE CODIGO**

#include <SPI.h> **//Aqui incluimos la libreria SPI**

#include <Ethernet.h> **//Aqui incluimos la librería Ethernet**

byte mac[] = {0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED}; **//Declaración de la dirección MAC**

IPAddress ip(192, 168, 1, 45); **//Declaración de la IP**

EthernetServer servidor(80); //**Declaración del puerto 80**

int PIN\_LED = 8; **//Aqui establecemos la variable PIN\_LED como un valor entero**

String readString = String(30); **//lee los caracteres de una secuencia en una cadena.**

//Los strings se representan como arrays de caracteres (tipo char)

String state = String(3);

void setup() {

Ethernet.begin(mac, ip); **//Inicializamos con las direcciones asignadas**

servidor.begin(); **//inicia el servidor**

pinMode(PIN\_LED, OUTPUT);

digitalWrite(PIN\_LED, LOW);

state = "OFF";

}

void loop() {

**//EthernetClient Crea un cliente que se puede conectar a**

**//una dirección específica de Internet IP**

EthernetClient cliente = servidor.available();

if (cliente) {

boolean lineaenblanco = true;

while (cliente.connected()) {

if (cliente.available()) {

char c = cliente.read();

if (readString.length() < 30) {

readString.concat(c);

//Cliente conectado

//Leemos petición HTTP carácter a carácter

//Almacenar los caracteres en la variable readString

} if (c == '\n' && lineaenblanco) { **//Si la petición HTTP ha finalizado**

int LED = readString.indexOf("LED=");

if (readString.substring(LED, LED + 5) == "LED=T") {

digitalWrite(PIN\_LED, HIGH);

state = "<b>Encendido</b>";

}

else if (readString.substring(LED, LED + 5) == "LED=F") {

digitalWrite(PIN\_LED, LOW);

state = "<b>Apagado</b>";

}

else if (readString.substring(LED, LED + 5) == "LED=V") {

digitalWrite(PIN\_LED, LOW);

delay(100);

digitalWrite(PIN\_LED, HIGH);

delay(100);

digitalWrite(PIN\_LED, LOW);

delay(100);

digitalWrite(PIN\_LED, HIGH);

delay(100);

digitalWrite(PIN\_LED, LOW);

delay(100);

digitalWrite(PIN\_LED, HIGH);

delay(100);

state = "<b>Parpadeo</b>";

}

//Cabecera HTTP estándar

cliente.println("HTTP/1.1 200 OK");

cliente.println("Content-Type: text/html");

cliente.println(); //Página Web en HTML

cliente.println("<html>");

cliente.println("<head>");

cliente.println("<title>Encender y apagar</title>");

cliente.println("</head>");

cliente.println("<body width=100% height=100%>");

cliente.println("<center>");

cliente.println("<h1>LED ON/OFF</h1>");

cliente.print("<br><br>");

cliente.print("Estado del LED: ");

cliente.print(state);

cliente.print("<br><br><br><br>");

cliente.println("<input type=submit value=ON style='background:#ccc;color:#000;width:200px;height:' onClick=location.href='./?LED=T'>");

cliente.println("<input type=submit value=OFF style='background:#ccc;color:#000;width:200px;height:75px' onClick=location.href='./?LED=F'>");

cliente.println("<input type=submit value=Parpadeo style='background:#ccc;color:#000;width:200px;height:75px' onClick=location.href='./?LED=V'>");

cliente.println("</center>");

cliente.println("</body>");

cliente.println("</html>");

cliente.stop();

//Cierro conexión con el cliente

readString = "";

}

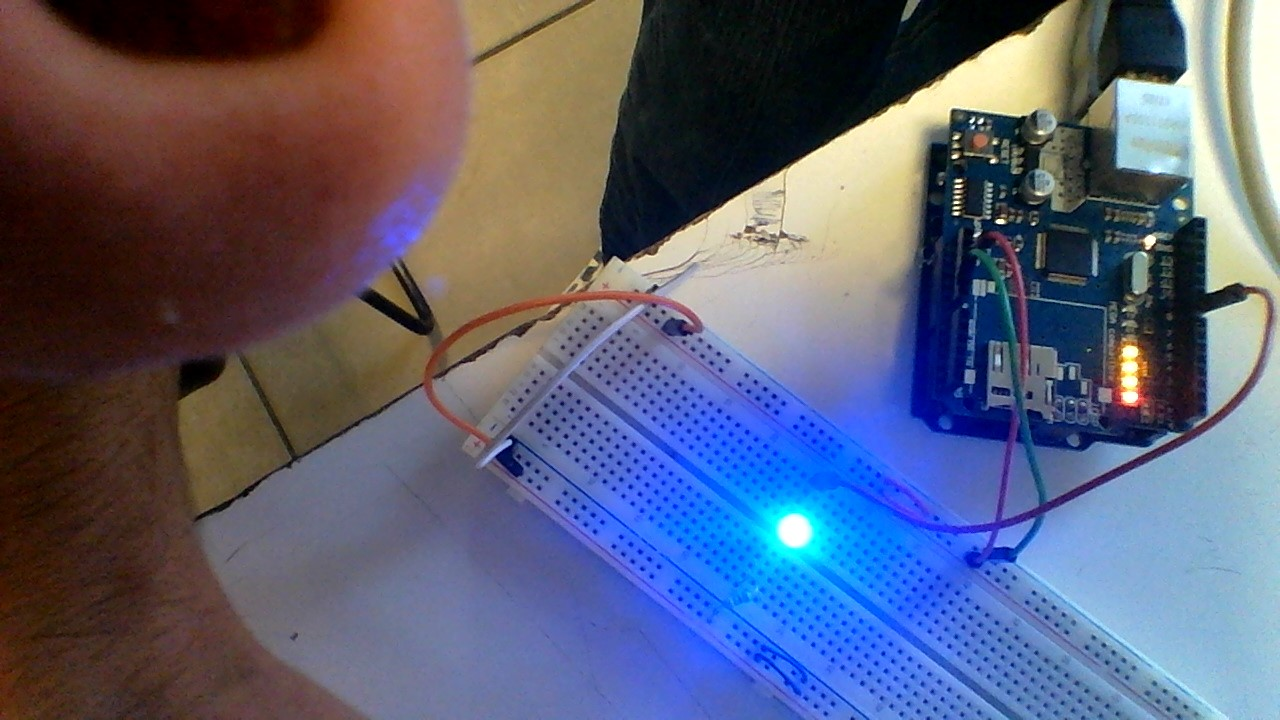
}

}

}

}

Resultado de la simulación.



**CONCLUSIONES**

Cuando se desarrollo en circuito con maquetación los 5 Volts de la batería no iban a ser eficientes para que el Arduino encendiera se optó por una pila de mayor capacidad como la de 9 Volts para que estos encendieran correctamente.