

Manejo vía WEB de 5 LED y 1 MicroServo

(Versión 26-9-19)

(Basado en proyecto similar de la pagina <http://elprofegarcia.com>)

Introducción:

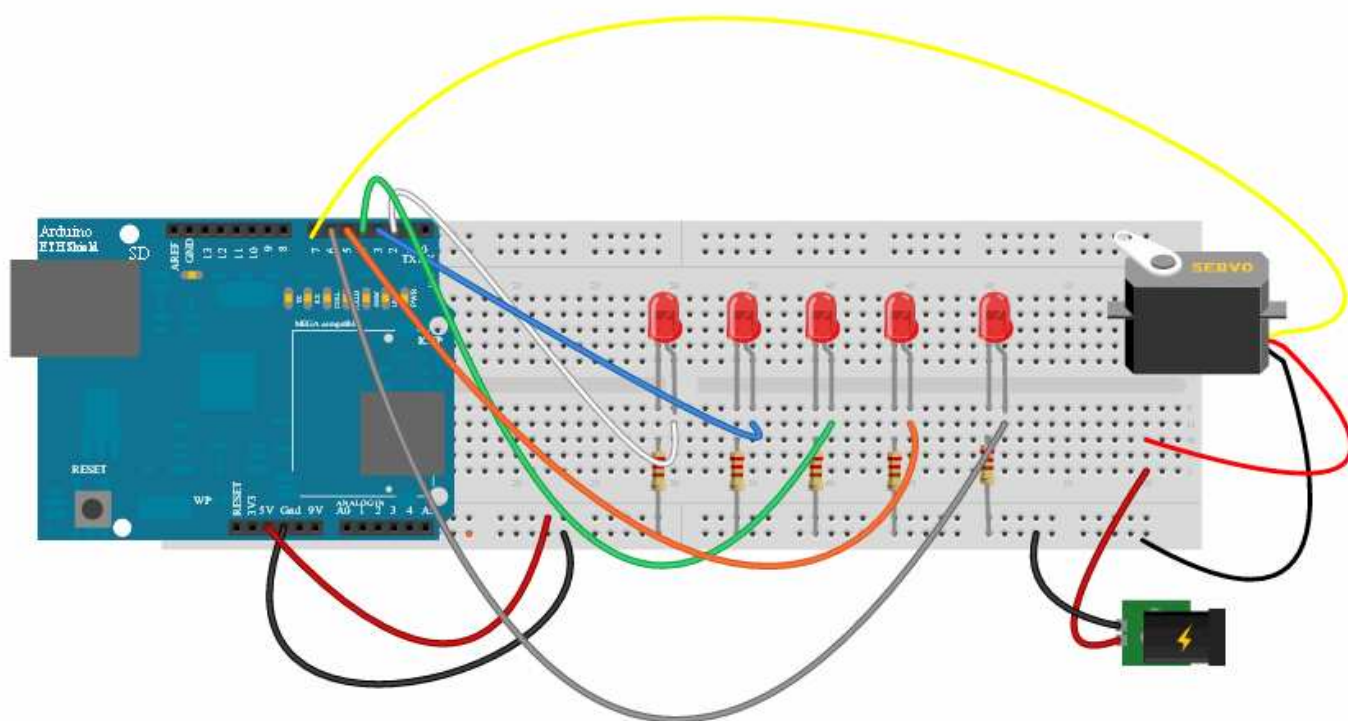
Este proyecto tiene la función de manejar mediante una pagina Web el encendido y apagado de 5 LEDs y manejo de un motor microservo.

El **Arduino Uno con su Shield Ethernet** es empleado como servidor Web, siendo la pagina generada de menor peso, permite aumentar la cantidad de objetos a controlar en relación al proyecto similar con botones.

El programa se altero del original, utilizando los conceptos mencionados cuando se trato el tema de servomotores.

Servo: <http://www.bolanosdj.com.ar/MOVIL/ARDUINO2/arduino.htm>

Esquema Fritzing del circuito empleado



Es importante recalcar que no es conveniente alimentar el motor servo con la misma fuente de Arduino, aun cuando utilicemos una fuente externa por el jack de la placa Arduino, dado que se exige el regulador de tensión interno de la placa.

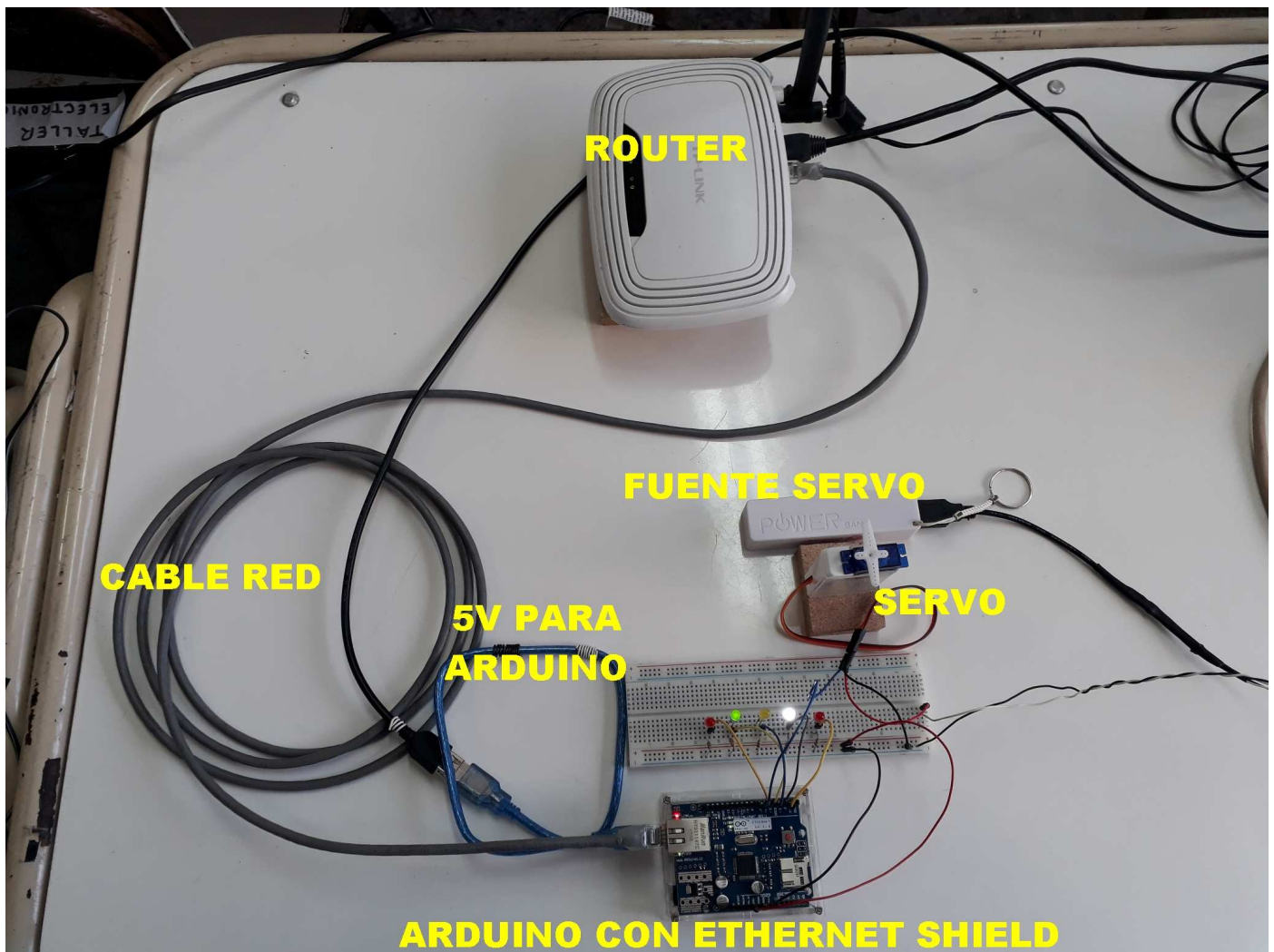
Lo conveniente es emplear una fuente propia para el motor y otra para la placa Arduino.

En la foto del circuito ensayado, se utilizo el puerto USB para la placa Arduino y el Shield Ethernet y los LEDs, y un cargador portátil de celular para el Servo.

Se conecta a un Router mediante cable de red, al mismo Router esta conectado el resto de los dispositivos con los cuales veremos la pagina servida por el Arduino, PC, movil etc.

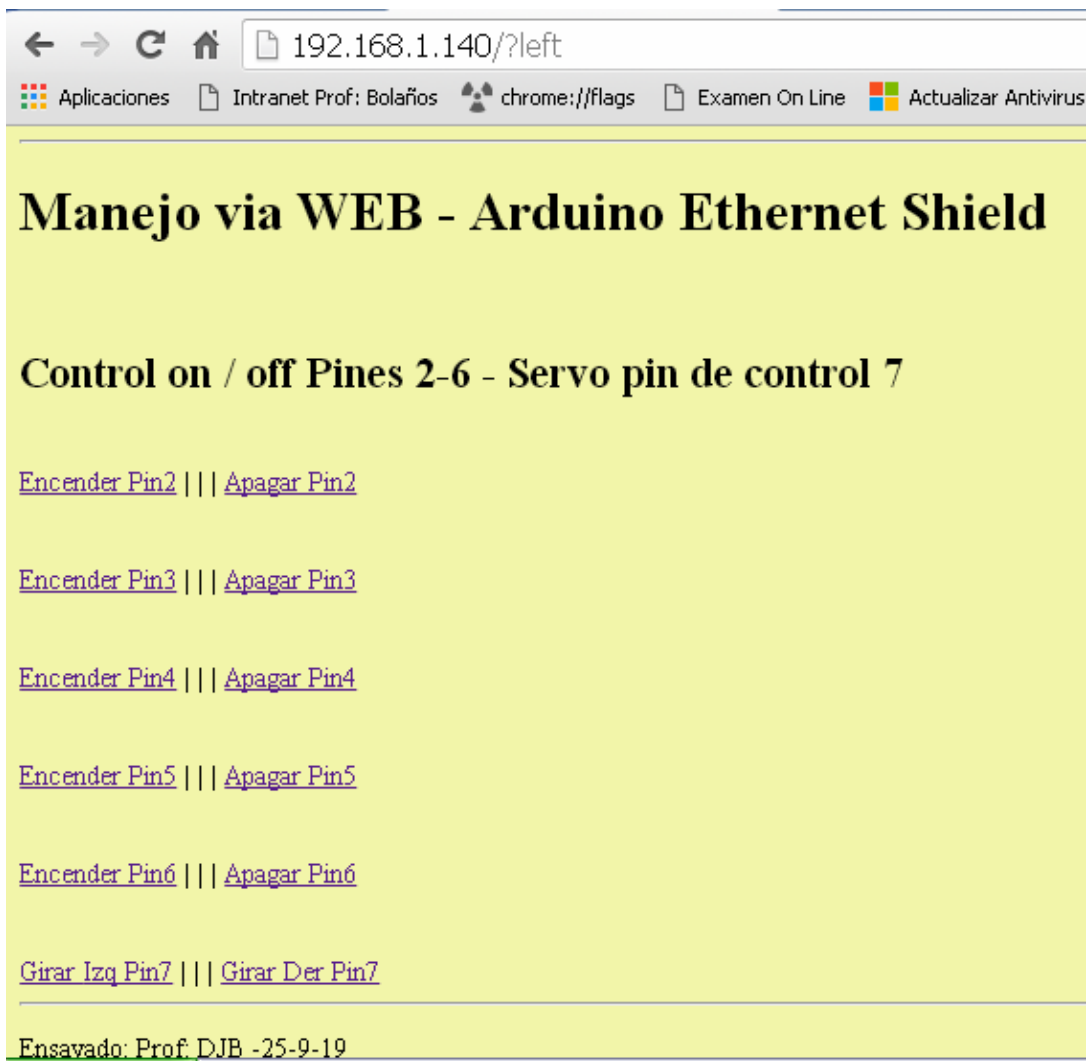
Visualizamos la página con la dirección IP.

Clickeando el la acción deseada, veremos inmediatamente el resultado.



En la PC, abrimos el navegador y cargamos la dirección IP de la pagina, en este caso:

192.168.1.140



Se incluye en un video con el funcionamiento.

El programa cargado en el Arduino se puede ver a continuación:

```

/* Control Arduino Ethernet
FUENTE: http://www.elprofegarcia.com/
*****
REVISADO PROF: BOLAÑOS DJ - 25-9-19

IMPORTANTE: RETIRE LA TARJETA SD SI SE ENCUENTRA COLOCADA

*****
Controla el encendido o apagado de los Pin 2,3,4 ,5 y 6 del Arduino.
Controla el giro de un servomotor conectado al pin 7

PARAR EL SERVO se cambio el codigo original del ejemplo fuente
por otro empleado anteriormente en el manejo de servo
*/

#include <SPI.h>
#include <Ethernet.h>
#include <Servo.h>

Servo servo1;

int PINSERVO = 7; // pin 7 conectado a señal del servo
int PULSOMIN = 750; // pulso minimo en microsegundos
int PULSOMAX = 2750; // pulso maximo en microsegundos
int num; // variable para almacenar valor leido
int ANGULO; // valor de angulo a cargar en el servo

int trespservo =300;// tiempo de detencion en posicion del servidor

byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED }; //Direccion Fisica MAC
byte ip[] = { 192, 168, 1, 140 }; // IP Local que usted debe configurar
//Router profesor Bolaños esta reservada esta direccion IP
byte gateway[] = { 192, 168, 1, 1 }; // Puerta de enlace
byte subnet[] = { 255, 255, 255, 0 }; //Mascara de Sub Red
EthernetServer server(80); //Se usa el puerto 80 del servidor
String readString;

void setup() {

  Serial.begin(9600); // Inicializa el puerto serial

  servo1.attach(PINSERVO, PULSOMIN, PULSOMAX); // inicializacion de servo

  while (!Serial) { // Espera a que el puerto serial sea conectado, Solo necesario para el Leonardo
    ;
  }
  pinMode(2,OUTPUT); // Se configura como salidas los puertos del 2 al 6
  pinMode(3,OUTPUT);
  pinMode(4, OUTPUT);
  pinMode(5,OUTPUT);
  pinMode(6,OUTPUT);

  Ethernet.begin(mac, ip, gateway, subnet); // Inicializa la conexion Ethernet y el servidor
  server.begin();
  Serial.print("El Servidor es: ");
  Serial.println(Ethernet.localIP()); // Imprime la direccion IP Local
}

void loop() {
  // Crea una conexion Cliente
  EthernetClient client = server.available();
  if (client) {
    while (client.connected()) {
      if (client.available()) {
        char c = client.read();

```

```

//Lee caracter por caracter HTTP
if (readString.length() < 100) {
  //Almacena los caracteres a un String
  readString += c;

}

// si el requerimiento HTTP fue finalizado
if (c == '\n') {
  Serial.println(readString); //Imprime en el monitor serial

  client.println("HTTP/1.1 200 OK"); //envia una nueva pagina en codigo HTML
  client.println("Content-Type: text/html");
  client.println();
  client.println("<HTML>");
  client.println("<HEAD>");
  client.println("<TITLE>Ethernet Arduino</TITLE>");
  client.println("</HEAD>");
  client.println("<BODY>");
  client.println("<hr />");
  client.println("<H1>Manejo via WEB - Arduino Ethernet Shield</H1>");

  client.println("<body style='background-color: #F2F5A9;'>"); //Color de fondo

  client.println("<br />");
  client.println("<H2>Control on / off Pines 2-6 - Servo pin de control 7</H2>");
  client.println("<br />");

  client.println("<a href='\"/?button2on\"'> Encender Pin2</a> "); // construye en la pagina cada uno de
los botones
  client.println(" | | | ");
  client.println("<a href='\"/?button2off\"'> Apagar Pin2</a><br /> ");
  client.println("<br />");

  client.println("<br />");
  client.println("<a href='\"/?button3on\"'> Encender Pin3</a> ");
  client.println(" | | | ");
  client.println("<a href='\"/?button3off\"'> Apagar Pin3</a><br /> ");
  client.println("<br />");

  client.println("<br />");
  client.println("<a href='\"/?button4on\"'> Encender Pin4</a> ");
  client.println(" | | | ");
  client.println("<a href='\"/?button4off\"'> Apagar Pin4</a><br /> ");
  client.println("<br />");

  client.println("<br />");
  client.println("<a href='\"/?button5on\"'> Encender Pin5</a>");
  client.println(" | | | ");
  client.println("<a href='\"/?button5off\"'> Apagar Pin5</a><br />");
  client.println("<br />");

  client.println("<br />");
  client.println("<a href='\"/?button6on\"'> Encender Pin6</a> ");
  client.println(" | | | ");
  client.println("<a href='\"/?button6off\"'> Apagar Pin6</a><br /> ");
  client.println("<br />");

  client.println("<br />");
  client.println("<a href='\"/?left\"'>Girar Izq Pin7</a>");
  client.println(" | | | ");
  client.println("<a href='\"/?right\"'>Girar Der Pin7</a><br />");
  client.println("<hr />");
  client.println("<p>Ensayado: Prof: DJB -25-9-19</p>");
  client.println("<p>Fuente: pagina Profe Garcia</p>");
  client.println("<br />");
  client.println("</BODY>");
  client.println("</HTML>");

  delay(1);
  //detiene el cliente servidor
  client.stop();

  //control del arduino si un boton es presionado

```

```

if (readString.indexOf("?button2on") >0){
  digitalWrite(2, HIGH);
}
if (readString.indexOf("?button2off") >0){
  digitalWrite(2, LOW);
}

if (readString.indexOf("?button3on") >0){
  digitalWrite(3, HIGH);
}
if (readString.indexOf("?button3off") >0){
  digitalWrite(3, LOW);
}

if (readString.indexOf("?button4on") >0){
  digitalWrite(4, HIGH);
}
if (readString.indexOf("?button4off") >0){
  digitalWrite(4, LOW);
}

if (readString.indexOf("?button5on") >0){
  digitalWrite(5, HIGH);
}
if (readString.indexOf("?button5off") >0){
  digitalWrite(5, LOW);
}

if (readString.indexOf("?button6on") >0){
  digitalWrite(6, HIGH);
}
if (readString.indexOf("?button6off") >0){
  digitalWrite(6, LOW);
}

if (readString.indexOf("?left") >0)

{
  ANGULO =180;
  girarservo();
  delay(trespservo);

}
if (readString.indexOf("?right") >0)

{
  ANGULO =0;
  girarservo();
  delay(trespservo);

}
// Limpia el String(Cadena de Caracteres para una nueva lectura
readString="";

}
}
}
}
//-----

void girarservo()
{
  servo1.write(ANGULO); // envia al servo el valor del angulo
  delay(20); // demora para que el servo llegue a posicion
}

```

Si desea cambiar características visuales, puede basarse en los programas antes vistos.