

ACTIVIDAD N°8:

APLICACION PRACTICA



1. DESCRIPCION

- **OBJETIVOS**

- 1.Realizar el conexionado de los distintos periféricos EZweb
- 2.Programar el microservidor EZweb.

- **METODOLOGIA**

En esta actividad realizaremos una aplicación practica en la que podremos ver la cantidad de luz que tenemos en una habitación, visualizaremos el estado de dos puertas (si están abiertas o cerradas) y podremos activar un motor (en los dos sentidos) y pararlo, indicándonos el sentido de giro del motor.

Esta aplicación consta de 4 páginas. La primera página es una página que contiene un menú que nos llevara a las páginas que sirven para visualizar la cantidad de luz que hay en una habitación, la página en la que visualizamos el estado de dos puertas y la página en la que controlaremos el motor de una persiana.

- **CONOCIMIENTOS PREVIOS**

Para poder seguir con garantías esta unidad didáctica:

Se requiere:

1. Conocimientos básicos de programación de páginas web.

Se recomienda:

2. Conocimientos básicos sobre electrónica.

- **MATERIAL Y EQUIPAMIENTO NECESARIO**

- ♦ Microservidor EZweb distribuido por MSE (Microsystems Engineering de Bilbao)
- ♦ Convendría tener el manual del microservidor Ezweb.
- ♦ Para poder grabar los programas y realizar distintas pruebas utilizaremos el entrenador PIC-SCHOOL distribuido por MSE (Microsystems Engineering de Bilbao).

• MONTAJE DEL PROTOTIPO

En este apartado realizaremos una descripción del patillaje del Microservidor Ezweb.

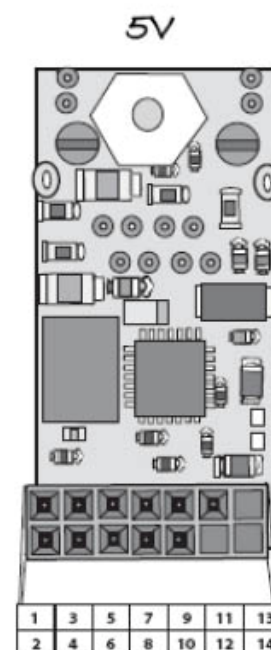
• DESCRIPCION DEL MICROSERVIDOR EZWEB

• PATILLAJE DEL EZWEB

Para realizar las siguientes actividades utilizaremos el microservidor Ezweb de 5V cuyo patillaje es el que a continuación se muestra.

Características técnicas

Pin Number	Description
1	Analog Input 1/Digital I/O
2	Analog Input 2/Digital I/O
3	Analog Input 3/Digital I/O
4	Analog Input 4/Digital I/O
5	Analog Input 5/Digital I/O
6	Digital I/O
7	Digital I/O
8	TX - Serial Data Transmit (from EZ Web Lynx to PC or other serial device)/Digital I/O
9	RCV - Serial Data Receive (from PC to EZ Web Lynx)/Digital I/O
10	Dallas DS1621 or DS1631 I ² C Temp Sensor SCL (clock) line/Digital I/O
11	Dallas DS1621 or DS1631 I ² C Temp Sensor SDA (data) line/Digital I/O
12	RST (Active-low Reset)
13	Vdd (5V)
14	GND



Tal como podemos ver, disponemos de 14 patillas que pueden trabajar como entradas/salidas digitales.

Los pines 1-5 pueden ser usados como entradas analógicas. Estas entradas tienen el rango GND-Vdd con una resolución de 0,02V en el módulo de 5V. Además pueden ser utilizados como entradas/salidas digitales. Al igual que los pines 1-5 los pines 6 y 7 son entradas/salidas digitales.

Los pines 8 y 9 se utilizan para la transmisión/recepción de datos. En cambio los pines 10 y 11, se utilizan como clock y data del bus I²C en el que solo podremos conectar los sensores de temperatura DS1621 y DS1631 de la casa Dallas.

• APLICACIÓN.

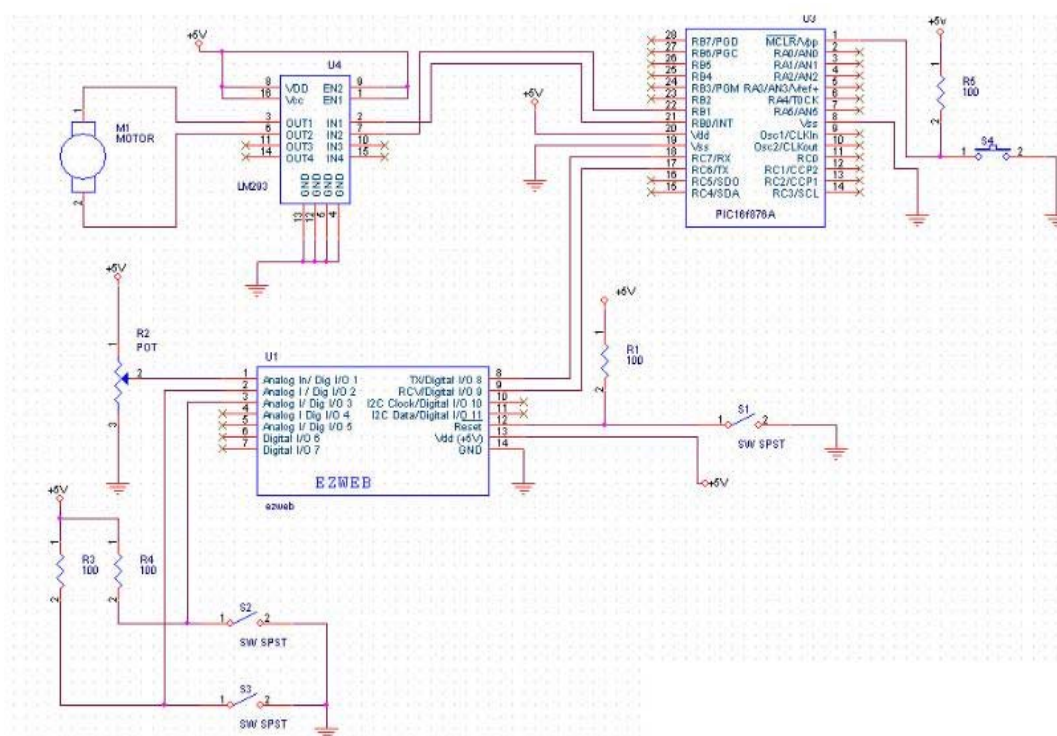
En este apartado realizaremos la aplicación práctica, utilizando los distintos elementos con los que cuenta el PicSchool. La programación de las páginas web las realizaremos utilizando el software Dreamweaver, donde insertaremos los distintos comandos que se utilizan con el microservidor EZweb.

Este ejemplo de aplicación consta de un esquema de conexiones, de los programas de las páginas web y del programa del microcontrolador.

Los programas de las páginas web son ficheros con extensión htm que tendremos que empaquetar con el software csfs y lo cargaremos en el microservidor mediante ftp.

• ESQUEMA DE CONEXIONES

En esta aplicación utilizaremos el siguiente esquema :



- PROGRAMAS

En este apartado veremos los distintos programas de los que consta la aplicación

- PROGRAMAS DE LAS PAGINAS WEB

En este apartado veremos los programas de las 4 páginas web de las que consta nuestra aplicación.

- PROGRAMA DE LA PAGINA PRINCIPAL

Esta página consta de un menú que nos posibilita ir a una de las tres páginas para poder controlar nuestra aplicación.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<!--(EXEC SERIAL_START=0X4E,0X3A)-->
//inicializo el buffer con los caracteres N:
<!--(EXEC SERIAL_END=0X3B)-->
// indico como fina de buffer el carácter “;”
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Documento sin título</title>
<style type="text/css">
<!--
.Estilo1 {
    font-size: 24px;
    font-weight: bold;
    color: #0000CC;
}
.Estilo3 {
    color: #0000CC;
    font-size: 24px;
}
-->
</style>
</head>
```

<body>

<p align="center" class="Estilo3">Monitorizar Luz</p>

//La palabra Monitorizar Luz tiene un hipervínculo a la página luz.htm, página web que nos indica la cantidad de luz que hay en una habitación.

<p align="center" class="Estilo3">Monitorizar Puertas</p>

//La palabra Monitorizar Puertas tiene un hipervínculo a la página puertas.htm, página web que nos indica el estado de las puertas de la casa.

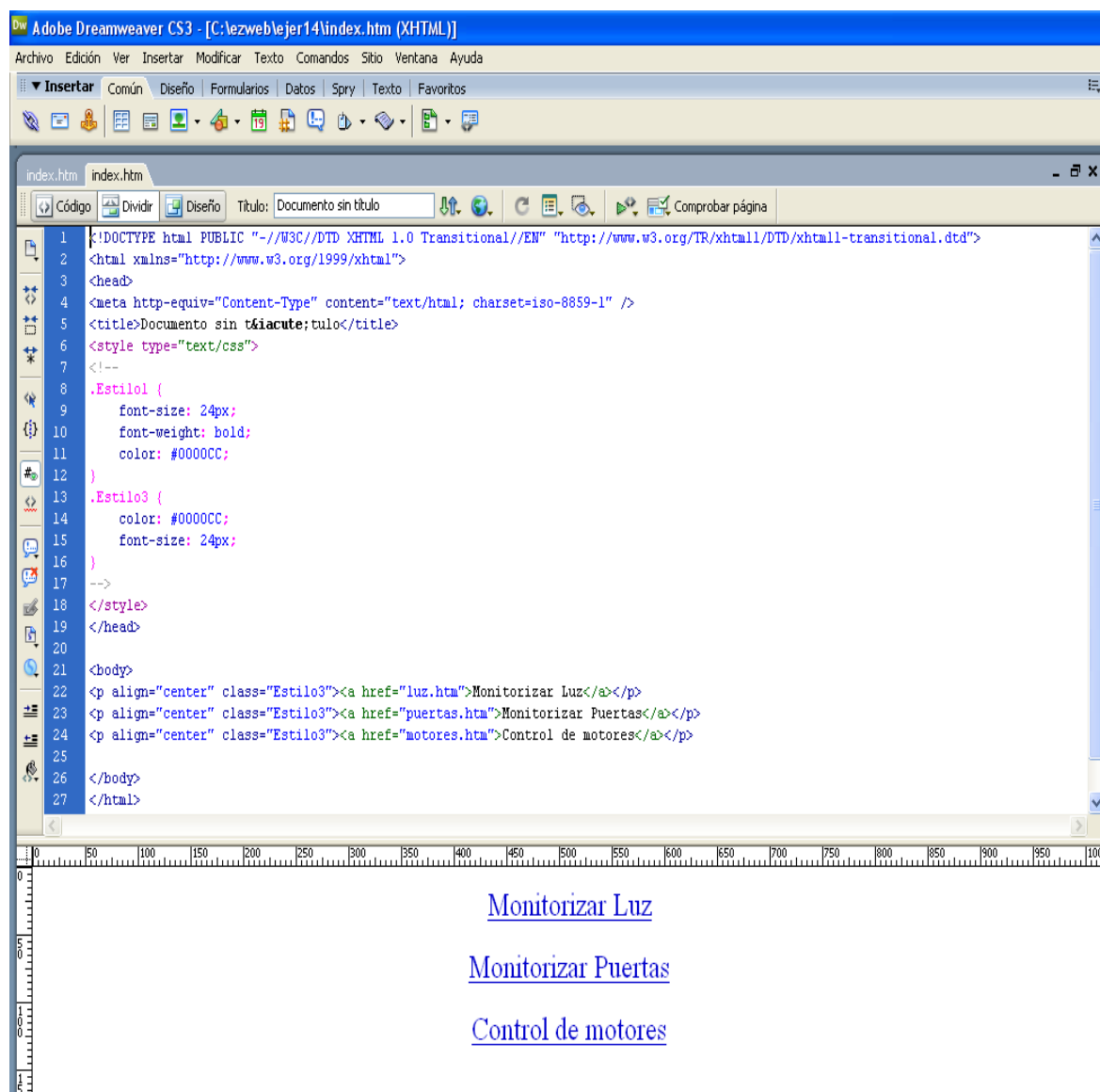
<p align="center" class="Estilo3">Control de motores</p>

//La palabra Control de motores tiene un hipervínculo a la página motores.htm, página web que nos permite controlar el motor de una persiana.

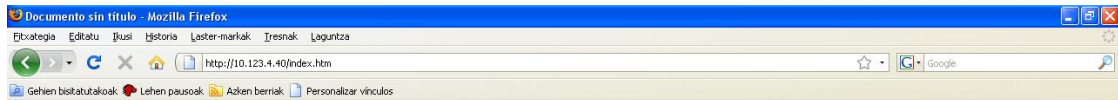
</body>

</html>

En la siguiente imagen podemos ver las diferentes opciones que hay dentro de nuestra pagina Web y lo que veríamos en el dreamweaver: Haciendo click-en los links o hipervinculos accederíamos a dichas paginas.



En la siguiente imagen podemos ver la página web que nos permitirá acceder a las demás páginas.



[Monitorizar Luz](#)

[Monitorizar Puertas](#)

[Control de motores](#)



• PROGRAMA DE MONITORIZAR LUZ

En este caso, monitorizaremos la luz. Para ello visualizaremos el valor de la entrada analógica y mostraremos la imagen de una lámpara encendida o apagada dependiendo si la entrada recibida es un "1" o un "0".

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
<head>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
```

```
<meta http-equiv="Refresh" content="2" />
```

```
//Refresco cada 2 segundos
```

```
<title>Monitorizar Luz</title>
```

```
<style type="text/css">
```

```
<!--
```

```
.Estilo1 { font-size: 24}
```

```
-->
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div align="center">
```

```
<table width="50%" border="0">
```

```
<tr>
```

```
<th scope="col"><!--(IF PIN_ANALOG1<=2,5)-->
```

//Si el valor de la entrada analógica es menor o igual a 2,5 voltios mostrar la imagen de la lámpara apagada.

```
<!--(ELSE)-->
```

//En los demás casos mostrar la imagen de la lámpara encendida.

```
<!--(ENDIF)--></th>
```

```
</tr>
```

```
<tr>
```

```
<th scope="col">El Valor de luz es: <!--(PIN 1 ANALOG*20,0)-->%</th>
```

//mostrar el valor de la señal analógica y mostrarla en tanto por ciento. Para ello el valor lo multiplicamos por 20 ya que el 100% de la señal lo tenemos cuando la señal es de 5V.

```
</tr>
```

```
</table>
```

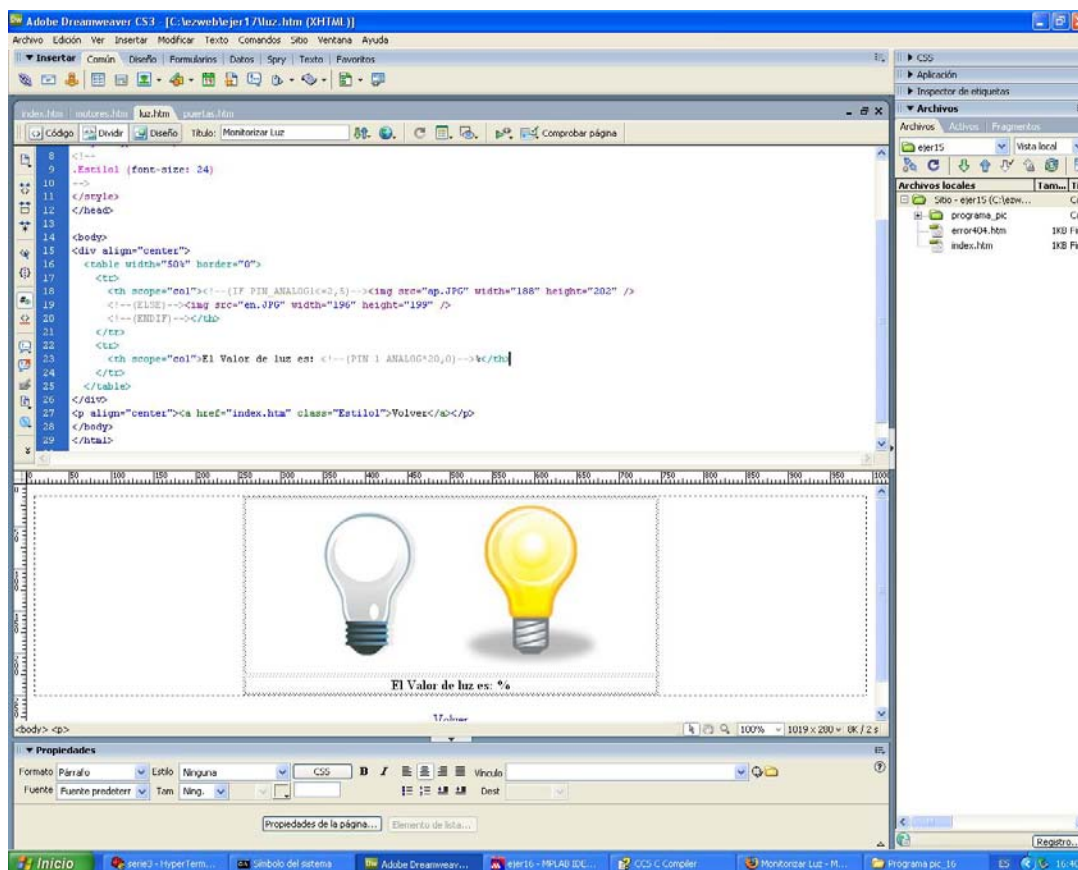
```
</div>
```

```
<p align="center"><a href="index.htm" class="Estilo1">Volver</a></p>
```

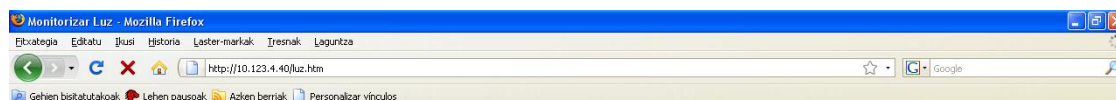
</body>

</html>

En la siguiente imagen podremos ver el programa escrito con dreamweaver:

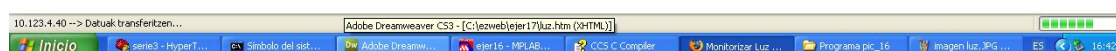


En la siguiente imagen podemos ver el ejemplo de la aplicación cuando en la entrada introducimos una señal analógica de 0V:

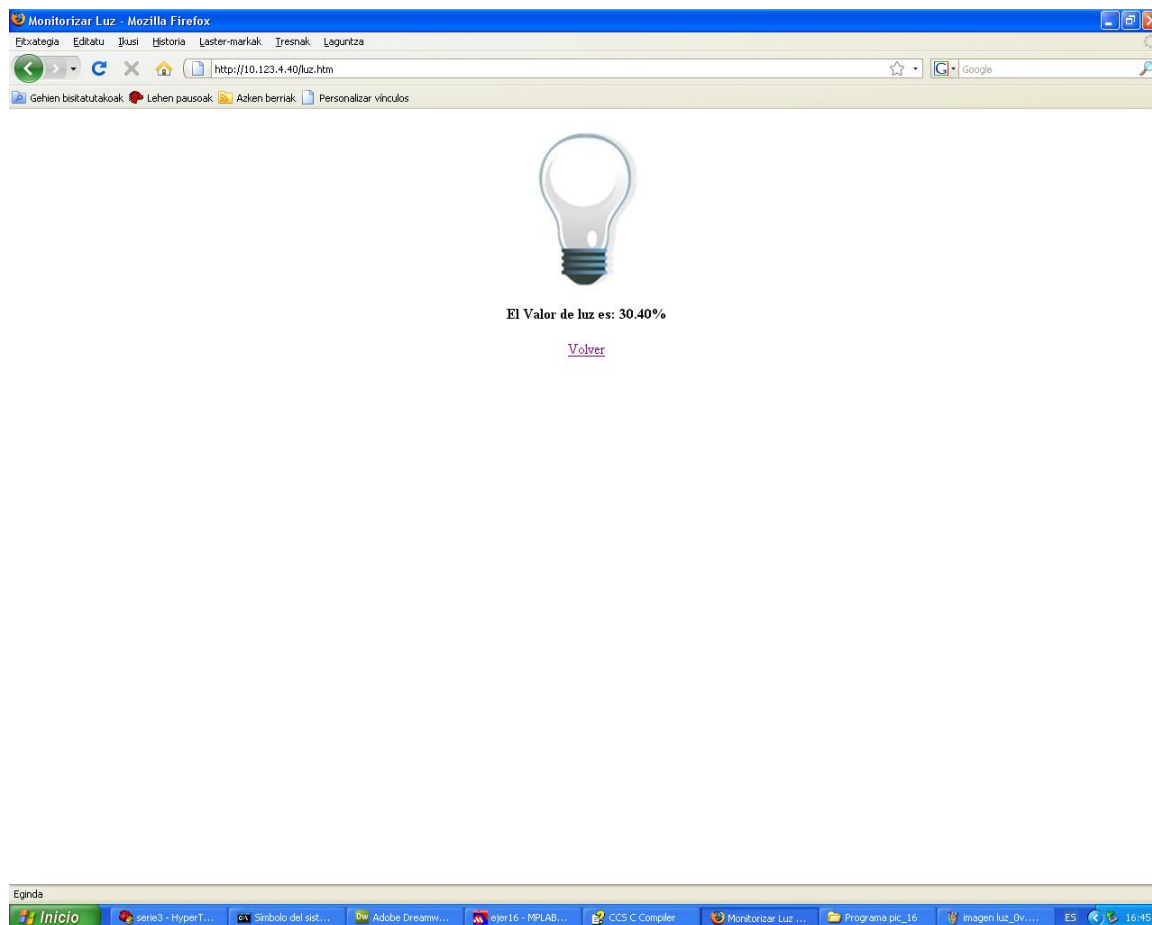


El Valor de luz es: 0.00%

[Volver](#)



En la siguiente imagen podemos ver el ejemplo de la aplicación cuando en la entrada introducimos una señal analógica de 1,53V:



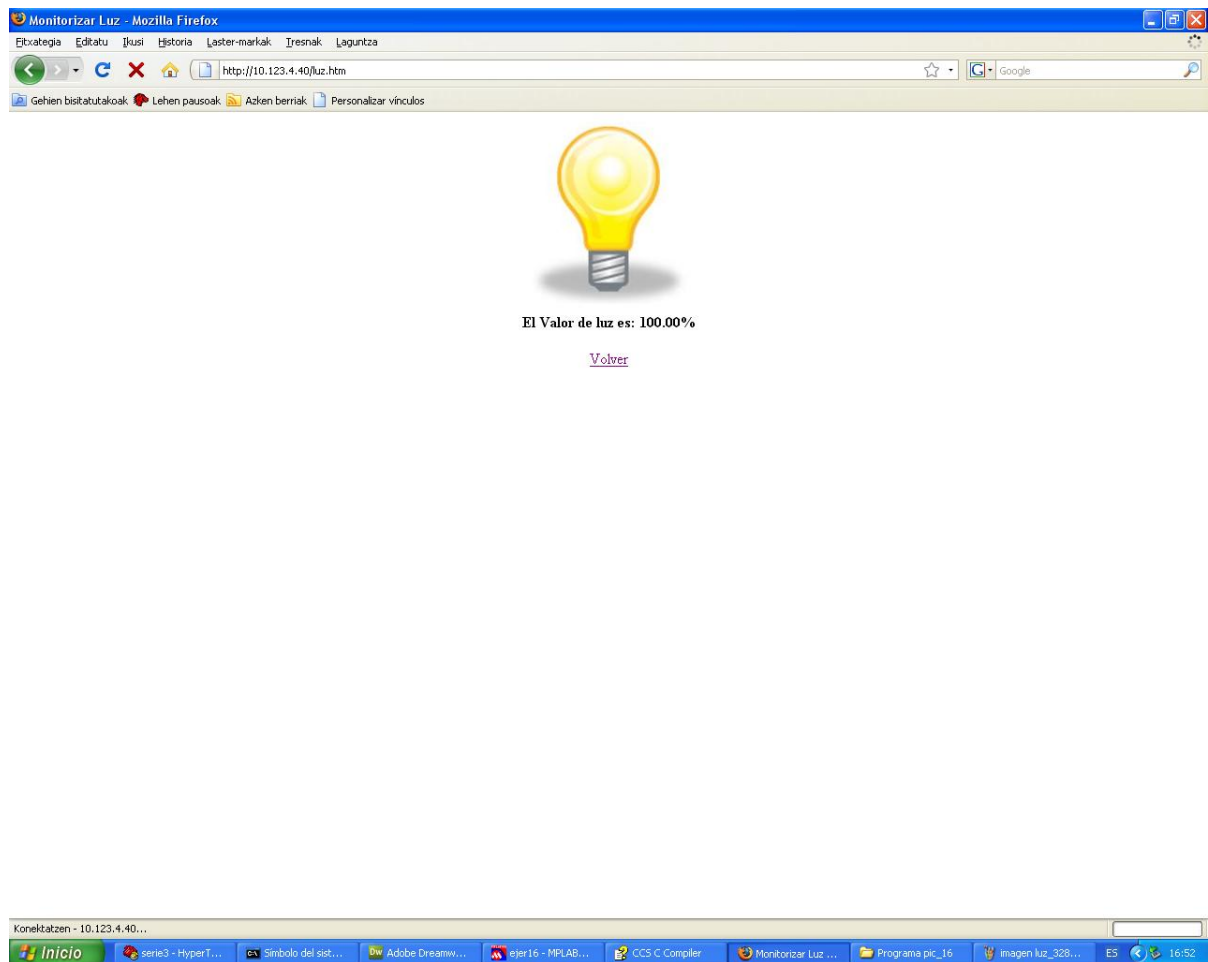
En la imagen aparece la imagen de la lámpara apagada por que el valor de la señal analógica es menor que 2,5V y el valor de la señal analógica puesto en tanto por ciento y cuyo valor es de 30,40%.

En la siguiente imagen podemos ver el ejemplo de la aplicación cuando en la entrada introducimos una señal analógica de 3,28V:



En la imagen aparece la imagen de la lámpara encendida por que el valor de la señal analógica es mayor que 2,5V y el valor de la señal analógica puesto en tanto por ciento y cuyo valor es de 65,20%.

En la siguiente imagen podemos ver el ejemplo de la aplicación cuando en la entrada introducimos una señal analógica de 5V:



En este caso podemos ver que el valor de la señal analógica en tanto por ciento es del 100%.

- PROGRAMA MONITORIZAR PUERTAS

En esta página podremos visualizar el estado de dos puertas. En este caso estamos visualizando el estado de dos señales digitales.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<meta http-equiv="Refresh" content="2" />
//Refresco de 2 segundos
<title>control puertas</title>
</head>

<body>
<p>&nbsp;</p>
<div align="center">
<table width="50%" border="1">
<tr>
<th scope="col">Puerta1</th>
<th scope="col"><!--(IF PIN_INPUT2=1)-->
//Si entrada digital2 (donde esta conectada la puerta1) es igual a 1 mostrar
imagen de candado abierto.
<!--(ELSE)-->
//En caso contrario mostrar candado cerrado.
<!--(ENDIF)--></th>
</tr>
<tr>
<th scope="col">Puerta2</th>
<th scope="col"><!--(IF PIN_INPUT3=1)-->
//Si entrada digital3 (donde esta conectada la puerta2) es igual a 1 mostrar
imagen de candado abierto.
<!--(ELSE)-->
//En caso contrario mostrar cerrado.
<!--(ENDIF)--></th>
</tr>
```

```

</table>

</div>

<p>&nbsp;</p>

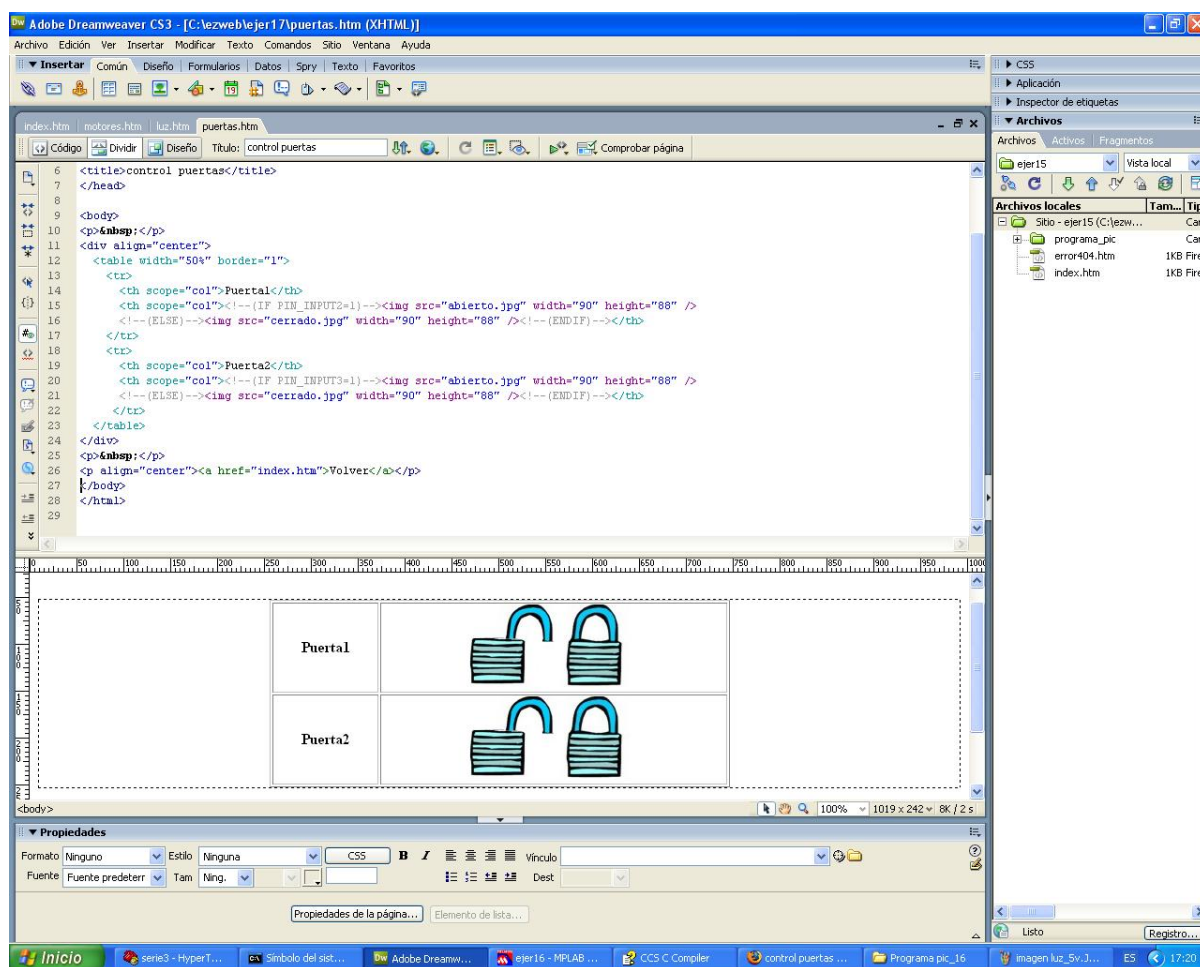
<p align="center"><a href="index.htm">Volver</a></p>

</body>

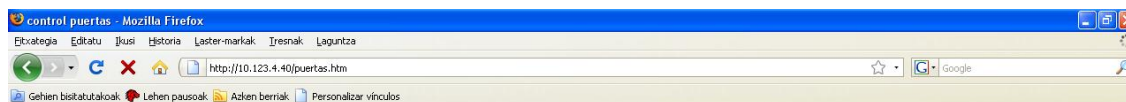
</html>


```

En la siguiente imagen podemos ver la página de monitorizar el estado de las puertas realizado con dreamweaver:

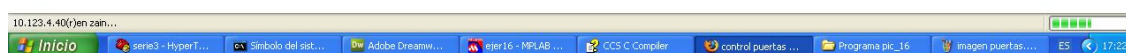


En la siguiente imagen podemos ver la aplicación cuando las puertas están cerradas.

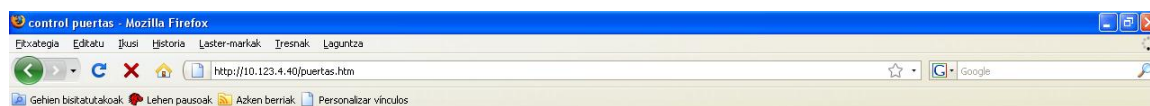




Puerta1	
Puerta2	

[Volver](#)



En la siguiente imagen podemos ver la aplicación cuando las una de las puertas esta abierta.

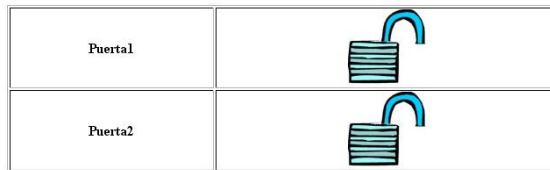
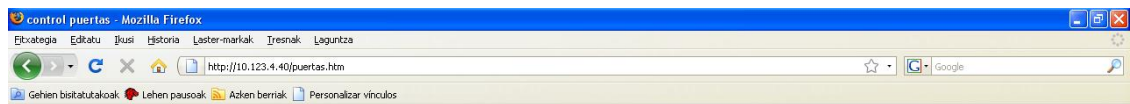


Puerta1	
Puerta2	

[Volver](#)



En la siguiente imagen podemos ver la aplicación cuando las puertas están abiertas.



[Volver](#)



• PROGRAMA DE LA PAGINA DE CONTROL DE MOTOTRES

En esta página tendremos tres botones: Uno para parar el motor, otro para activar el motor a derechas y el tercero para activar el motor a izquierdas. En una casa automatizada, por ejemplo podrían ser las persianas, el boton de activar el motor a derechas podría ser el de subir persianas y el boton de activar el motor a izquierdas sería el de bajar persianas.

A continuación podemos ver el programa correspondiente a la página web que realiza el control de motores.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>activar motores</title>
<style type="text/css">
<!--
.Estilo1 {
    font-size: 24px;
    font-weight: bold;
}
-->
</style>
</head>

<body>
<p class="Estilo1">ACTIVACION DE MOTORES</p>
<table width="91%" border="1">
<tr>
<th width="35%" scope="col">
<form method="get" >
<div align="center">
<input name="SERIAL_STRING" value="a" type="hidden"/>
<input name="PIN_OUTPUT07" value="0" type="hidden"/>
<input name="PIN_OUTPUT05" value="0" type="hidden"/>
<input type="submit" value="PARAR MOTOR" />
//botón que se utiliza para parar el motor y que envía por el puerto serie el
carácter "a" para que cuando el microcontrolador reciba "a" pare el motor.
También coloca el pin 7 y el 5 a "0" para que luego me aparezca el mensaje de
motor parado.
```

```

    </div>
  </form> </th>
<th width="35%" scope="col">
  <form method="get" >
    <div align="center">
      <input name="SERIAL_STRING" value="c" type="hidden"/>
      <input name="PIN_OUTPUT07" value="1" type="hidden"/>
      <input name="PIN_OUTPUT05" value="0" type="hidden"/>
      <input type="submit" value="MOTOR IZQ" />

//botón que se utiliza para mover el motor a izquierdas y que envia por el puerto serie el carácter "c" para que cuando el microcontrolador reciba "c" mueva el motor a izquierdas. También coloca el pin 7 a "1" y el pin 5 a "0" para que luego me aparezca el mensaje de motor izquierda.

    </div>
  </form></th>
<td width="35%"></td>
  <form method="get" >
    <div align="center">
      <input name="SERIAL_STRING" value="b" type="hidden"/>
      <input name="PIN_OUTPUT07" value="0" type="hidden"/>
      <input name="PIN_OUTPUT05" value="1" type="hidden"/>
      <input type="submit" value="MOTOR DCHA" />

//botón que se utiliza para mover el motor a derechas y que envia por el puerto serie el carácter "b" para que cuando el microcontrolador reciba "b" mueva el motor a derechas. También coloca el pin 7 a "0" y el pin 5 a "1" para que luego me aparezca el mensaje de motor derecha.

    </div>
  </form></td>
</tr>
</table>

<table width="311" border="1" align="center">
  <tr>
    <td width="301" height="33">
      <div align="center">
        <!--(IF PIN 7 = 1)-->
          MOTOR IZQUIERDA
// Si el pin 7 es "1" mostrar el mensaje MOTOR IZQUIERDA
        <!--(ELSE)-->
        <!--(IF PIN 5 = 1)-->

```

MOTOR DERECHA

// Si el pin 7 es "0" y el pin 5 es "1" mostrar el mensaje MOTOR DERECHA

```
<!--(ELSE)-->
```

MOTOR PARADO

// Si el pin 7 es "0" y el pin 5 es "0" mostrar el mensaje MOTOR PARADO

```
<!--(ENDIF)-->
```

```
</div></td>
```

```
</tr>
```

```
</table>
```

```
<p></p>
```

```
<p>&nbsp;</p>
```

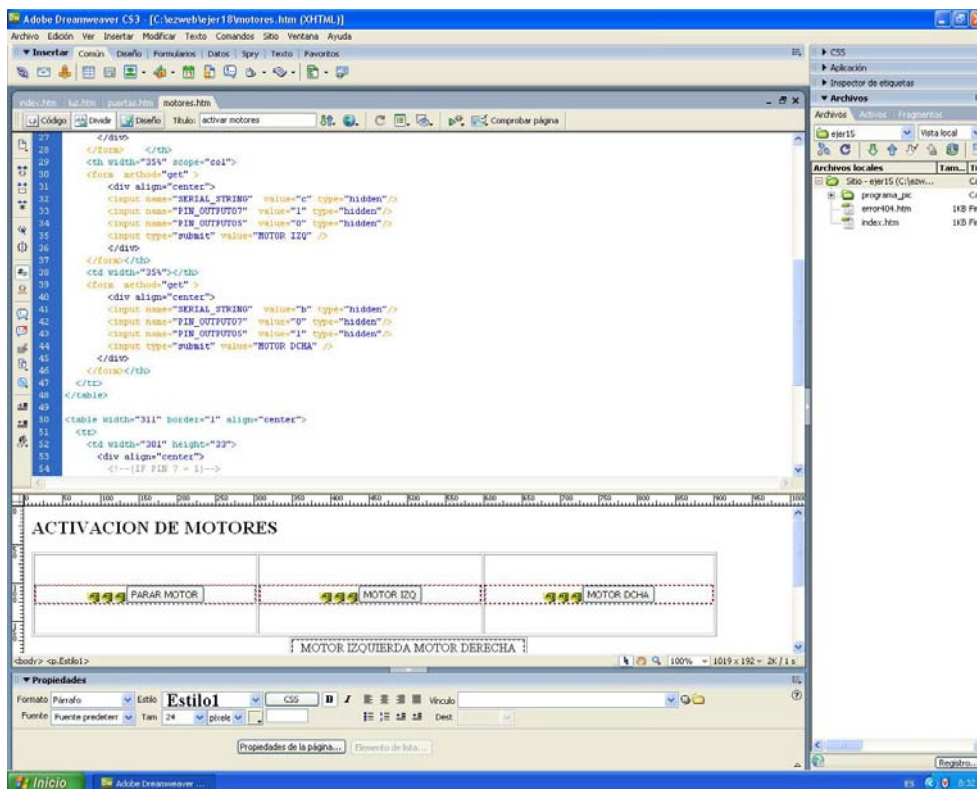
```
<p align="center"><a href="index.htm">Volver</a></p>
```

// Hipervínculo a la página principal para poder seguir el proceso.

```
</body>
```

```
</html>
```

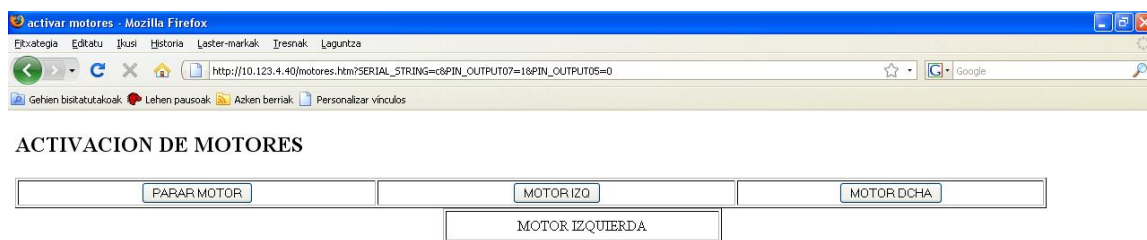
En la siguiente imagen podemos ver la pantalla del programa control de motores escrito con dreamweaver.



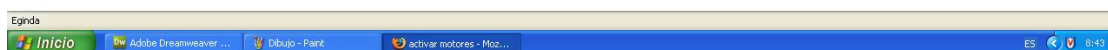
En la siguiente imagen podemos ver la pantalla de la aplicación del programa control de motores cuando el motor esta apagado.



En la siguiente imagen podemos ver la pantalla de la aplicación del programa control de motores cuando activamos motor izaq. En este caso veremos como el motor se activa hacia izquierdas y en la página nos muestra el mensaje motor izquierda.



En la siguiente imagen podemos ver la pantalla de la aplicación del programa control de motores cuando activamos motor dcha. En este caso veremos como el motor se activa hacia derechas y en la página nos muestra el mensaje motor derecha.



- PROGRAMA DEL MICROCONTROLADOR

En este apartado se vera el programa que contiene el microcontrolador. En este caso el microcontrolador recibe por el puerto serie un carácter que dependiendo del carácter que sea realizara una tarea concreta.

En este caso recibirá por el puerto serie tres posibles caracteres que controlaran un motor que esta conectado al puertob.

Si se recibe el carácter "a" mandara al puertob el valor 0x00 para que pare el motor.

Si recibe el carácter "b" mandara al puertob el valor 0x01 para que active el motor y gire a izquierdas.

Si recibe el carácter "c" mandara al puertob el valor 0x02 para que active el motor y gire a derechas.

```

/*****

Protocolo entre el PIC y el ezweb

*****/

// TxD ==> RC6   con RCV del ez web (patilla 9)
// RxD ==> RC7   con TX del ez web (patilla 8)

/*****

PROYECTO control desde web

*****/

#include <16F876A.h>
#use delay (clock=4000000)
#use rs232 (baud=9600,xmit=PIN_C6,rcv=PIN_C7)
//RC6 transmisor (TxD) y Rc7 receptor (RxD)

#fuses XT,NOWDT
#byte puertob=0x06
#define paro puertob=0x00 //defino el valor 0x00 como paro
#define izq puertob=0x01 //defino el valor 0x01 como izquierda
#define dcha puertob=0x02 //defino el valor 0x02 como derecha

void main(void)
{
    int8 letra;
    set_tris_b(0);          //pongo como salida RB0-RB5
    set_tris_c(0b10000000); //pongo como entrada RC7

    puertob=0x00;          // al inicio mando paro el motor

    while(1)
    {
        letra=getc();       //Recojo carácter por línea serie
        switch (letra)
        {
            case 'a':       // Si el valor recibido es "a"
                puertob=paro; // Paro el motor
                break;

```

```
case 'b':          // Si el valor recibido es "b"
    puertob=izq; // Activo motor a izquierdas
    break;
case 'c':          // Si el valor recibido es "c"
    puertob=dcha; // Activo motor a derechas
    break;
} //fin de switch
} //fin de while
} //fin de main
```