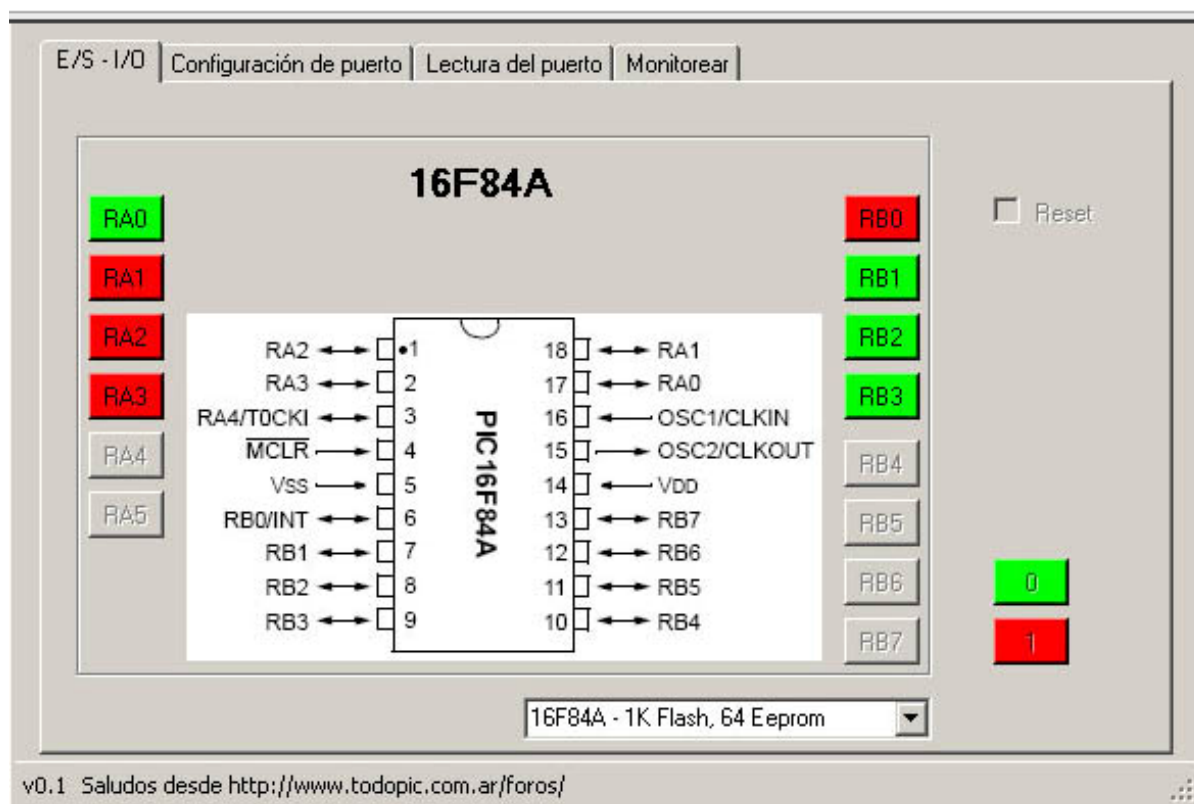


ACTIVIDAD N°4:

ACTIVACION DE SALIDAS POR WEB



©msdn.microsoft

1. DESCRIPCION

1.1 OBJETIVOS

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

1.2 METODOLOGIA

En esta actividad activaremos distintas salidas conectadas al microservidor EZweb, explicaremos como se programan botones para que activen distintos dispositivos y explicaremos la forma de ejecutar los distintos comandos de ejecución.

2 CONOCIMIENTOS PREVIOS

Para poder seguir con garantías esta unidad didáctica:

Se requiere:

- ♦ Conocimientos básicos de programación de páginas web.

Se recomienda:

- ♦ Conocimientos básicos sobre electrónica.

3 MATERIAL Y EQUIPAMIENTO NECESARIO

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

4 MONTAJE DEL PROTOTIPO

En este apartado realizaremos una descripción del patillaje del Microservidor Ezweb así como los comandos a utilizar con las salidas digitales y analógicas.

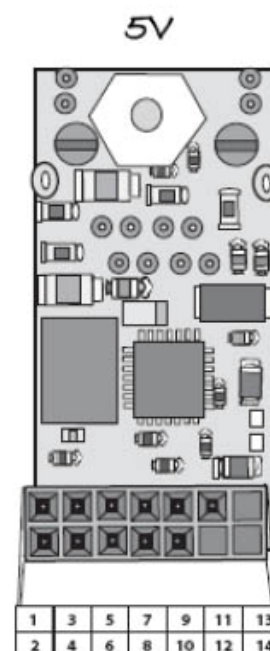
4.1 DESCRIPCION DEL MICROSERVIDOR EZWEB

4.1.1 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 11 patillas (desde la pata 1 a la 11) que pueden trabajar como entradas/salidas digitales.

Cuando las entradas/salidas digitales están configuradas como entradas digitales, cualesquiera de las otras características del pin (puerto serie, I2C, entradas analógicas) no están disponibles.

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.

5 COMANDOS E INSTRUCCIONES A UTILIZAR

5.1 INSTRUCCIONES O ETIQUETAS DINAMICAS

Las instrucciones que utilizaremos con el EZweb son los que a continuación mostramos:

Lista de instrucciones		
<!--(PIN x)-->	x es un número de pin.	El elemento es sustituido con un 0 si el pin está bajo y 1 si está alto. El pin debe estar en modo entrada.
<!--(PIN x ltag/htag)-->	x es un número de pin. ltag es el texto si el pin está bajo. htag es el texto si el pin está alto.	El elemento es sustituido con ltag si el pin está bajo y con htag si está alto. El pin debe estar en modo entrada.
<!--(PIN x ANALOG)-->	x es un número de pin.	El elemento es sustituido con un número en formato x.xx que representa el voltaje en el pin. El pin debe estar en modo entrada analógica.
<!--(PIN x ANALOG*a/b+cd,p)-->	x es un número de pin a es un número a multiplicar por el voltaje analógico (optional). b es un número a dividir por el voltaje analógico (optional). c es un número a sumar con el voltaje analógico (optional). d es un número a restar con el voltaje analógico (optional). p es el número de decimales.	El elemento es sustituido con un número que representa el resultado del cálculo. El pin debe estar en modo entrada analógica. Los operadores matemáticos (*/+/-) deben ser usados en este orden.
<!--(SERIAL DATA x y)-->	x es el index inicial de la última trama de datos serie capturada. y es el index final de la última trama de datos serie capturada.	El elemento es sustituido por la última trama de datos serie, desde el carácter index x hasta el carácter index y. Para leer toda la trama serie completa se utiliza la extensión READ HTML.
<!--(IF PIN x = n)-->	x es un número de pin n es 0 ó 1	Si la condición es falsa, se ignora el código hasta ELSE o ENDIF.
<!--(IF PIN x < a)-->	x es un número de pin	Si la condición es falsa, se ignora el código hasta ELSE o ENDIF.
<!--(IF PIN x > a)-->	a es un número en el rango (0.00-5.00) (5 V version)	Si la condición es falsa, se ignora el código hasta ELSE o ENDIF.
<!--(IF KEY = n)--> <!--(IF KEY != n)--> <!--(IF KEY < n)--> <!--(IF KEY <= n)--> <!--(IF KEY > n)--> <!--(IF KEY >= n)-->	KEY is un comando válido. n puede ser un entero o un decimal, dependiendo del comando utilizado.	Si el argumento del comando no cumple la condición, se ignora el código hasta ELSE o ENDIF.
<!--(ELSE)-->		Usado para la directiva IF
<!--(ENDIF)-->		Usado para la directiva IF/ELSE
<!--(EXEC KEY=VALUE)-->		Ejecuta un comando.
<!--(READ KEY)-->		Lee un comando.

5.2 COMANDOS

Los comandos a utilizar para leer entradas son los que a continuación describimos:

PIN_OUTPUTxx: Cuando se ejecuta este comando, mandaremos un "0" o un "1" a la salida digital que indicamos.

Este comando se ejecuta con la instrucción **<!--(EXEC KEY=VALUE)-->** donde la palabra KEY se sustituye por el comando a utilizar, y VALUE será el valor que queramos enviar.

Un ejemplo de aplicación de los comandos es el que a continuación mostramos.

<!--(EXEC PIN_OUTPUT01=0)--> Este comando mandara un "0" a la salida digital01.

6 EJEMPLOS DE APLICACIÓN. SOFTWARE

En este apartado realizaremos distintos ejemplos de aplicación, 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. Los comandos que utilizamos en estos ejemplos están realizados con etiquetas dinámicas.

Estos ejemplos de aplicación constarán de una pequeña descripción de la aplicación, de un esquema de conexiones y del programa de la página web.

El programa de la página web es un fichero o varios ficheros con extensión htm que tendremos que empaquetar con el software csfs y lo cargaremos en el microservidor mediante ftp.

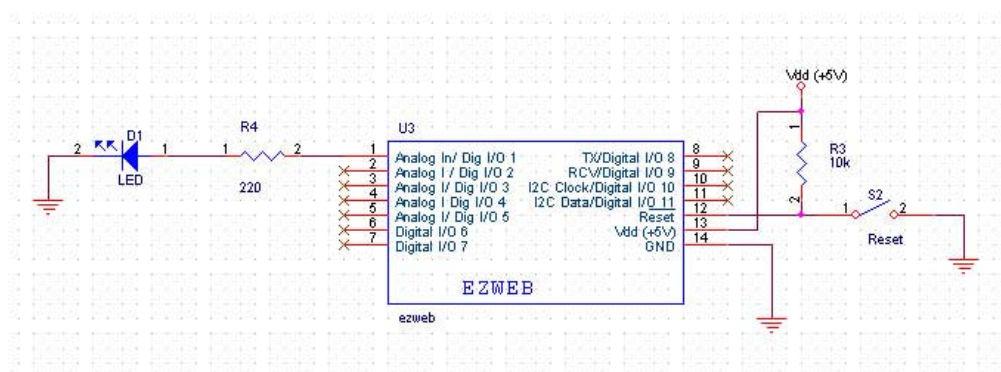
6.1 APLICACIÓN 1 (ACTIVACION DE SALIDA DIGITAL)

6.1.1 DESCRIPCION

La primera aplicación que realizaremos tiene como objetivo activar o desactivar una salida digital desde una pagina web.

6.1.2 ESQUEMA DE CONEXIONES

En la siguiente imagen mostraremos el esquema de conexiones para activar una salida desde una página web.



6.1.3 PROGRAMA

Esta aplicación consta de 2 páginas vinculadas entre sí. En cada página insertaremos el comando `<!--(EXEC PIN_OUTPUT1=VALUE)-->`. Cuando visualizamos la página `index.htm` ejecutaremos `<!--(EXEC PIN_OUTPUT1=0)-->` que sacará por la salida digital un "0", además en esta página tenemos un texto "ENCENDER LED" que vincularemos a otra página en la que tendremos el comando `<!--(EXEC PIN_OUTPUT1=1)-->` que nos activa la salida digital 1. En esta página también tendremos el texto "APAGAR LED" que nos servirá para crear un vínculo con la página `index.htm` que apaga el diodo led.

Esto nos permite que pulsando en un texto que se vincula a una página, activemos o desactivemos una salida digital.

El programa de la página `index.htm` es el que a continuación se muestra.

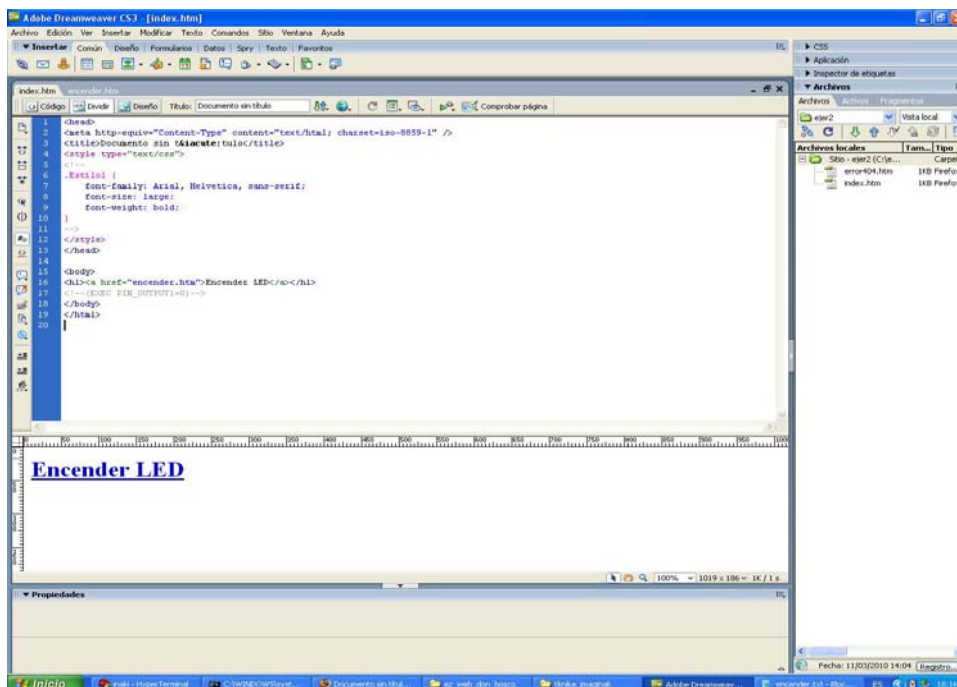
```
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Documento sin título</title>
<style type="text/css">
<!--
.Estilo1 {
    font-family: Arial, Helvetica, sans-serif;
    font-size: large;
    font-weight: bold;
}
-->
</style>
</head>

<body>
<h1><a href="encender.htm">Encender LED</a></h1>

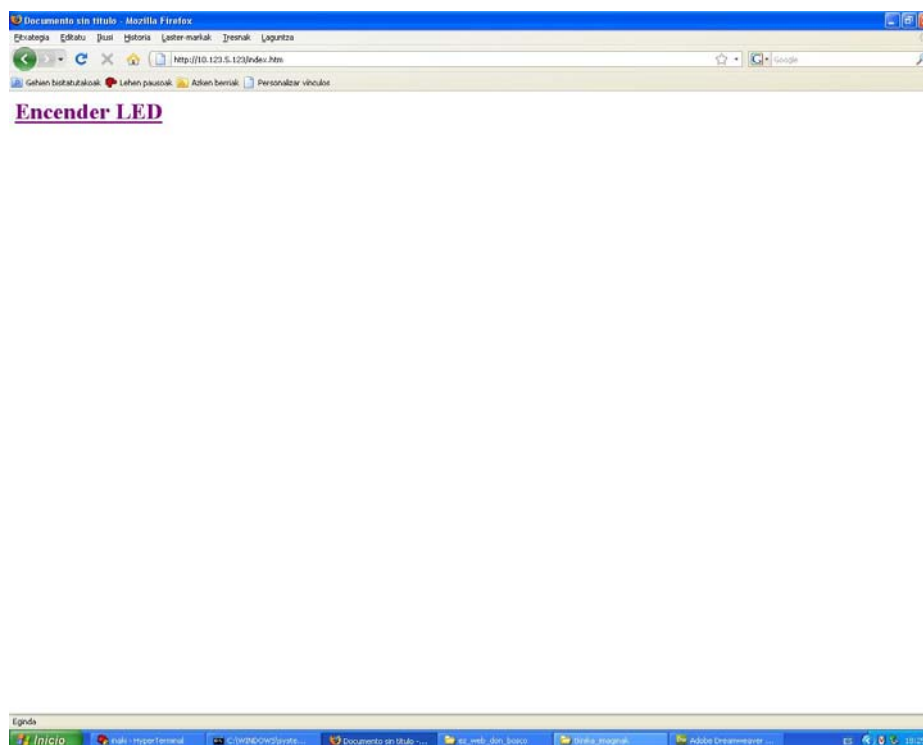
    //Vínculo a la página encender.htm

<!--(EXEC PIN_OUTPUT1=0)-->    //Comando sacar un "0" por la salida digital1
</body>
</html>
```


La siguiente imagen muestra la página escrita con dremweaver:



La siguiente imagen muestra la página tal como se ve en Internet:



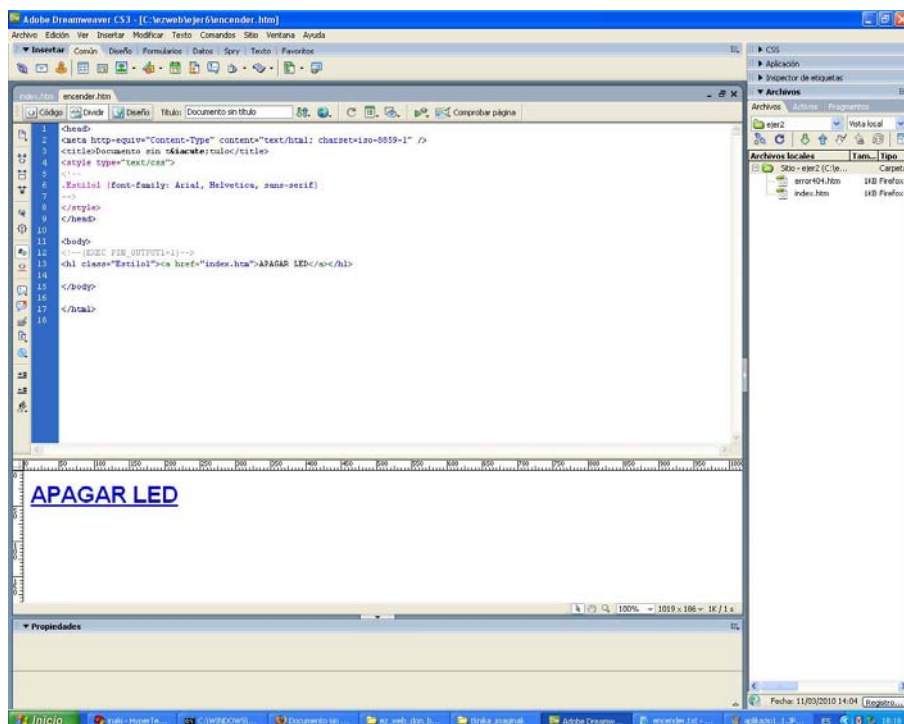
El programa de la página encender.htm es el que a continuación se muestra.

```
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Documento sin título</title>
<style type="text/css">
<!--
.Estilo1 {font-family: Arial, Helvetica, sans-serif}
-->
</style>
</head>

<body>
<!--(EXEC PIN_OUTPUT1=1)--> //Comando sacar un "1" por la salida digital1
<h1 class="Estilo1"><a href="index.htm">APAGAR LED</a></h1>
//Vinculo a la página index.htm

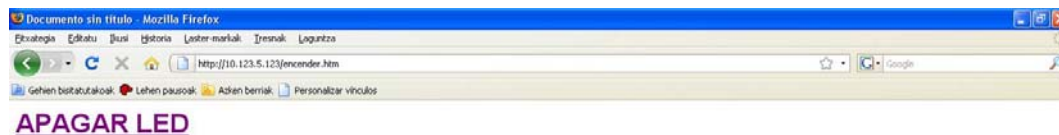
</body>
</html>
```

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





La siguiente imagen muestra la página tal como se ve en Internet:



6.2 APLICACIÓN 2 (ACTIVACION DE SALIDAS DIGITALES MEDIANTE BOTONES)

6.2.1 DESCRIPCION

En esta 2ª aplicación volveremos a realizar el mismo montaje, pero en la página web mostraremos dos botones, uno de ON para que active la salida digital y otro de OFF que desactivara la salida digital.

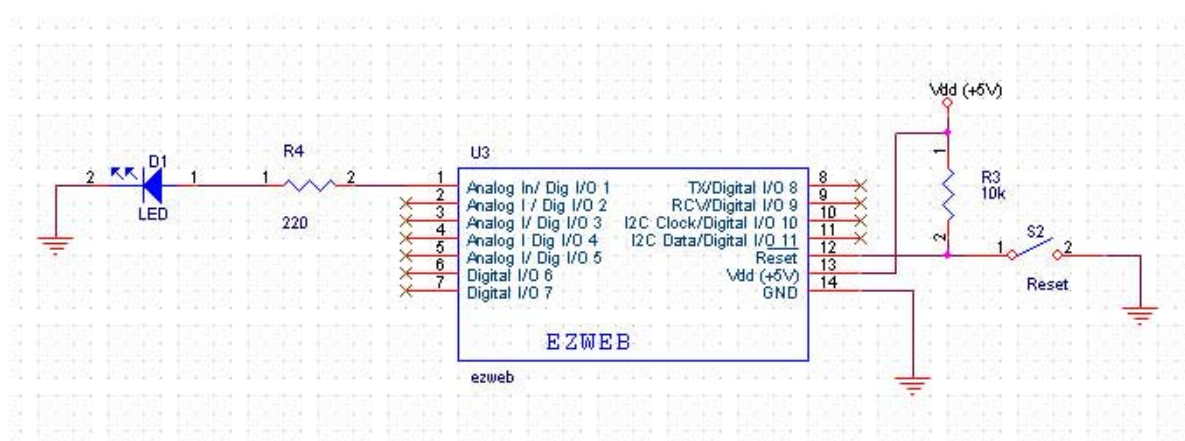
A su vez la aplicación muestra un texto "ON" si se activa la salida digital y un texto "OFF" si se desactiva la salida digital.

Para los casos en los que tenemos que poner botones para que ejecuten algún comando, necesitamos crear formularios.

En las páginas que usemos botones, los insertaremos en tablas en las que los bordes no se visualicen, de tal forma que nos posibilitará colocar el botón en la posición que mas nos interese.

6.2.2 ESQUEMA DE CONEXIONES

En este caso tendremos el siguiente esquema de conexiones.



Tal como podemos comprobar este montaje es igual que el anterior. En este caso solo cambiamos la programación de la página web ya que seguimos activando una salida digital. En este caso en vez de que se active mediante un texto vinculado, lo activaremos o desactivaremos mediante dos botones, uno para que se active la salida y otro para desactivarlo.

5.2.3 PROGRAMA

En el programa de la aplicación, al utilizar botones necesitaremos crear formularios.

En lenguaje htm, para cada botón necesitamos crear un formulario. Cuando creamos el formulario, el programa correspondiente al formulario es el que a continuación se muestra:

```
<form id="form1" name="form1" method="get" action="">
    <input name="PIN_OUTPUT1" type="hidden" value="1" />
    <input name="" type="submit" value="ON" />
</form>
```

En la programación destacaremos los siguientes apartados:

Name: es el nombre del formulario

Method: En este apartado pondremos el método de enviar el formulario. En este caso existen dos métodos de enviar el formulario:

Método get: Con este método enviaremos el formulario y además lo visualizamos en la cabecera (donde se visualiza la dirección de la página). En principio este método nos viene bien por que nos permite ver los comandos que se están enviando al pulsar el botón.

Método post: Con este método enviamos el formulario pero no se visualizara en la cabecera (donde se visualiza la dirección de la página).

Input name: Aquí pondremos el comando que queremos enviar. En el ejemplo de arriba tenemos dos Input, el primero, el correspondiente al comando del ezweb, PIN_OUTPUT1 (activar salida 1) y el segundo, el correspondiente al botón que aparece entre comillas ya que el botón no envía ningún comando.

Type: En este caso indicamos el tipo de entrada que se utiliza. En el ejemplo de arriba el primer input es de tipo hidden (oculto), ya que es la entrada correspondiente al ezweb. Y el segundo input es de tipo submit (enviar) con el que indicamos que al pulsar el botón enviaremos todos los input del formulario.

Value: En este apartado indicaremos el valor a enviar. En el ejemplo de arriba, en el primer caso enviamos un "1". En el segundo caso enviaremos la palabra ON.

En definitiva cuando enviamos un formulario, lo que hacemos es enviar

input name=value

Siguiendo el ejemplo de arriba lo que enviamos es el comando de activar la salida 1

PIN_OUTPUT1=1

También enviamos el comando del botón que envía

=ON

En nuestro programa tendremos dos botones por lo que deberemos de tener dos formularios. Con un formulario enviaremos al ezweb el comando PIN_OUTPUT1=1 que activara la salida 1 y con el otro formulario enviaremos al ezweb el comando PIN_OUTPUT1=0 para desactivar la salida digital1.

Nuestro programa también leerá la salida digital y dependiendo del valor que leamos, mostrara el texto ON u OFF. En caso de leer un "0" visualizaremos el texto OFF y en caso de leer un "1" mostraremos el texto ON.

El programa de la aplicación 2 nos queda de la siguiente manera:

```
<!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>Documento sin título</title>
<style type="text/css">
<!--
.Estilo1 {color: #FF0000}
.Estilo2 {color: #00FF00}
-->
</style>
</head>

<body>
<table width="50%" border="0">
<tr>
<td width="38%"><div align="center">
<form id="form1" name="form1" method="get" action="">
<input name="PIN_OUTPUT1" type="hidden" value="0" />
<input name="" type="submit" value="OFF" />
</form>
//formulario correspondiente al botón OFF
</div></td>
<td width="32%"><div align="center">
<form id="form1" name="form1" method="get" action="">
<input name="PIN_OUTPUT1" type="hidden" value="1" />
<input name="" type="submit" value="ON" />
</form>
//formulario correspondiente al botón ON
</div></td>
<td width="30%"><div align="center"></div></td>
</tr>
<tr>
<td height="48"><!--(IF PIN_OUTPUT1=0)--><div align="center"
class="Estilo1">OFF</div><!--(ENDIF)--></td>
//si la salida digital1 es "0" visualiza la palabra OFF
<td><!--(IF PIN_OUTPUT1=1)--><div align="center" class="Estilo2">ON</div><!--
(ENDIF)--></td>
```

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

```
//si la salida digital1 es "1" visualiza la palabra ON
```

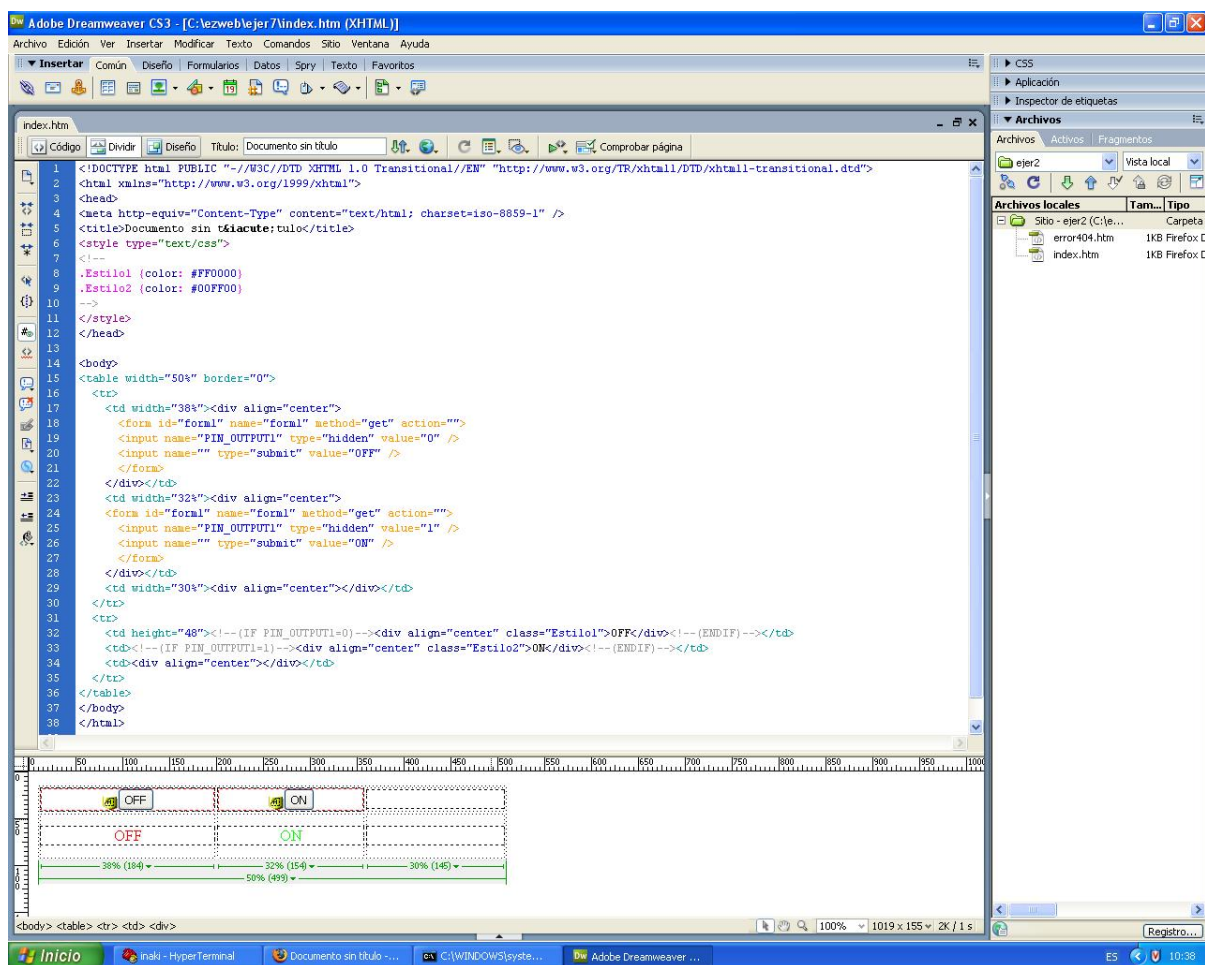
```
</tr>
```

```
</table>
```

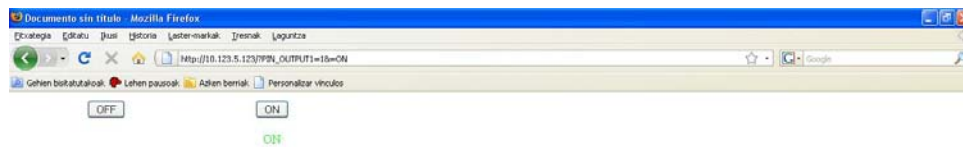
```
</body>
```

```
</html>
```

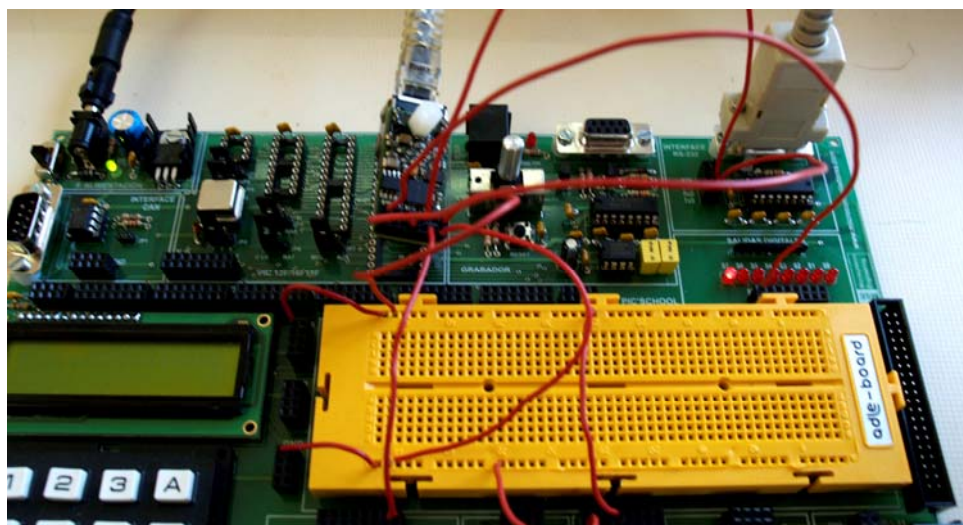
En la siguiente imagen podremos ver el programa escrito con dreamweaver:



En la siguiente imagen podemos ver el ejemplo de la aplicación después de pulsar el pulsador ON:



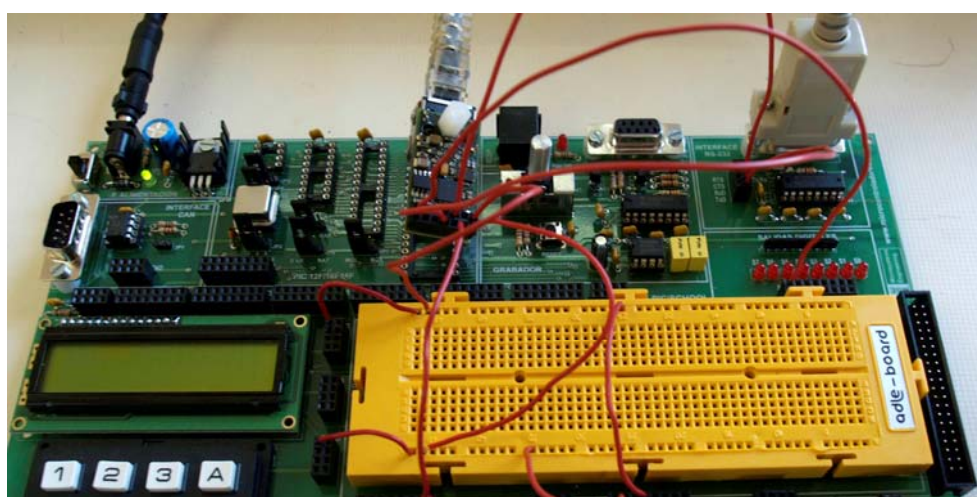
En la siguiente imagen podemos ver la aplicación en marcha después de pulsar el botón de ON:



En la siguiente imagen podemos ver el ejemplo de la aplicación después de pulsar el pulsador OFF:



En la siguiente imagen podemos ver la aplicación en marcha después de pulsar el botón de OFF:



6.3 APLICACIÓN 3 (ACTIVACION DE VARIAS SALIDAS DIGITALES)

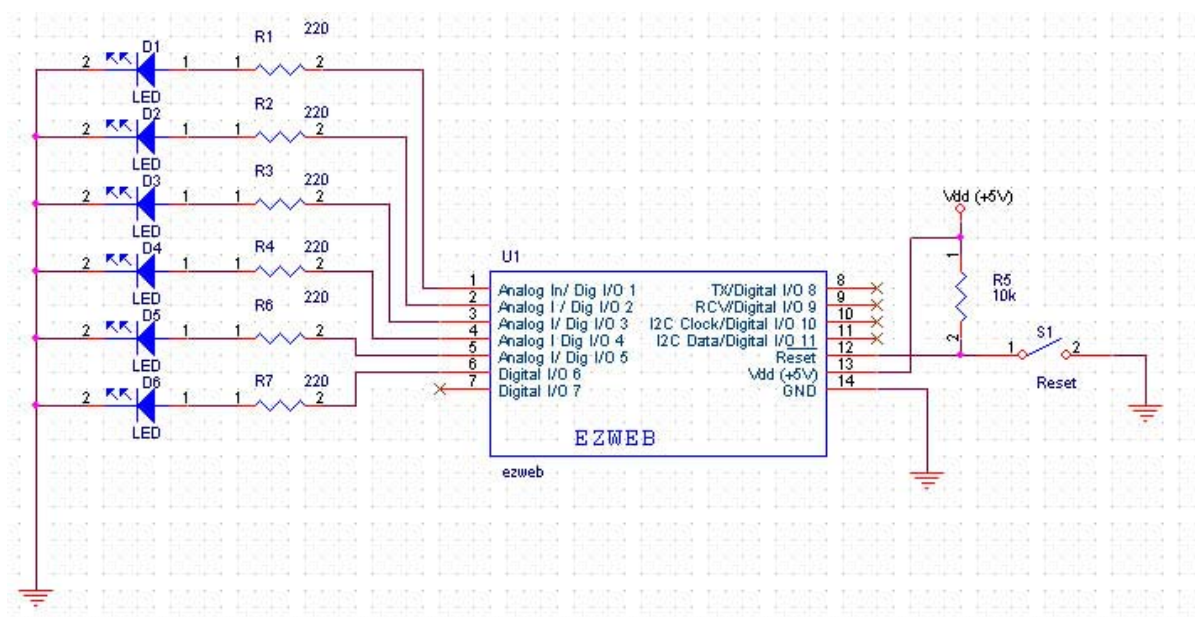
6.3.1 DESCRIPCION

En esta 3ª aplicación activaremos más de una salida digital.

Mostraremos una serie de interruptores que activaran una salida digital cada una.

6.3.2 ESQUEMA DE CONEXIONES

En este caso tendremos el siguiente esquema de conexiones.



En esta aplicación tenemos 7 pares de botones. Los 6 primeros pares, activan o desactivan cada una de las 6 salidas digitales que tenemos activadas. Y el 7 par de botones activara o desactivara todas las salidas digitales a la vez.

6.3.3 PROGRAMA

El programa de la aplicación 3 consiste en repetir varias veces el programa de la aplicación 2. Por cada botón enviaremos un formulario en el que se tiene que enviar un comando al ezweb. En el caso del botón que sirve para activar todas las salidas digitales, el formulario deberá de enviar varios comandos al ezweb.

El programa de la aplicación3 se muestra a continuación:

```
<!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" />
```

```
//Si la salida digital1 es 0 visualizar texto OFF
```

```

<input name="PIN_OUTPUT2" type="hidden" value="1" />
<input name="" type="submit" value="ON" />
</form></td>

<td><form id="form2" name="form2" method="get" action="">
<input name="PIN_OUTPUT2" type="hidden" value="0" />
<input name="" type="submit" value="OFF" />
</form>
</td>

<td><!--(IF PIN_OUTPUT2=1)--><span class="Estilo1">ON</span><!--(ENDIF)-->
<!--(IF PIN_OUTPUT2=0)--><span class="Estilo2">OFF</span><!--(ENDIF)-->
</td>
</tr>

```

// igual que los botones de la salida digital1 pero para la salida digital 2

```

<tr>
<td>OUTPUT3</td>

<td><form id="form1" name="form1" method="get" action="">
<input name="PIN_OUTPUT3" type="hidden" value="1" />
<input name="" type="submit" value="ON" />
</form>
</td>

<td><form id="form2" name="form2" method="get" action="">
<input name="PIN_OUTPUT3" type="hidden" value="0" />
<input name="" type="submit" value="OFF" />
</form>
</td>

<td><!--(IF PIN_OUTPUT3=1)--><span class="Estilo1">ON</span><!--(ENDIF)-->
<!--(IF PIN_OUTPUT3=0)--><span class="Estilo2">OFF</span><!--(ENDIF)--></td>
</tr>

```

// igual que los botones de la salida digital1 pero para la salida digital 3

```

<tr>
<td>OUTPUT4</td>

<td><form id="form1" name="form1" method="get" action="">
<input name="PIN_OUTPUT4" type="hidden" value="1" />
<input name="" type="submit" value="ON" />
</form>
</td>

<td><form id="form2" name="form2" method="get" action="">

```

```

<input name="PIN_OUTPUT4" type="hidden" value="0" />
<input name="" type="submit" value="OFF" />
</form>
</td>
<td><!--(IF PIN_OUTPUT4=1)--><span class="Estilo1">ON</span><!--(ENDIF)-->
      <!--(IF PIN_OUTPUT4=0)--><span class="Estilo2">OFF</span><!--(ENDIF)-->
</td>
</tr>

```

// igual que los botones de la salida digital1 pero para la salida digital 4

```

<tr>
<td>OUTPUT5</td>
<td><form id="form1" name="form1" method="get" action="">
<input name="PIN_OUTPUT5" type="hidden" value="1" />
<input name="" type="submit" value="ON" />
</form></td>
<td><form id="form2" name="form2" method="get" action="">
<input name="PIN_OUTPUT5" type="hidden" value="0" />
<input name="" type="submit" value="OFF" />
</form></td>
<td><!--(IF PIN_OUTPUT5=1)--><span class="Estilo1">ON</span><!--(ENDIF)-->
      <!--(IF PIN_OUTPUT5=0)--><span class="Estilo2">OFF</span><!--(ENDIF)-->
</td>
</tr>

```

// igual que los botones de la salida digital1 pero para la salida digital 5

```

<tr>
<td>OUTPUT6</td>
<td><form id="form1" name="form1" method="get" action="">
<input name="PIN_OUTPUT6" type="hidden" value="1" />
<input name="" type="submit" value="ON" />
</form>
</td>
<td><form id="form2" name="form2" method="get" action="">
<input name="PIN_OUTPUT6" type="hidden" value="0" />
<input name="" type="submit" value="OFF" />
</form>
</td>
<td><!--(IF PIN_OUTPUT6=1)--><span class="Estilo1">ON</span><!--(ENDIF)-->

```

```
<!--(IF PIN_OUTPUT6=0)--><span class="Estilo2">OFF</span><!--(ENDIF)-->
```

```
</td>
```

```
</tr>
```

// igual que los botones de la salida digital1 pero para la salida digital 6

```
<tr>
```

```
<td>Todas las salidas </td>
```

```
<td><form id="form3" name="form3" method="get" action="">
```

```
<input name="PIN_OUTPUT1" type="hidden" value="1" />
```

```
<input name="PIN_OUTPUT2" type="hidden" value="1" />
```

```
<input name="PIN_OUTPUT3" type="hidden" value="1" />
```

```
<input name="PIN_OUTPUT4" type="hidden" value="1" />
```

```
<input name="PIN_OUTPUT5" type="hidden" value="1" />
```

```
<input name="PIN_OUTPUT6" type="hidden" value="1" />
```

```
<input name="" type="submit" value="ON" />
```

```
</form>
```

```
</td>
```

**// En este caso si pulso el botón de todas las salidas tendré que mandar todos
//comandos PIN_OUTPUTX=1**

```
<td><form id="form4" name="form4" method="get" action="">
```

```
<input name="PIN_OUTPUT1" type="hidden" value="0" />
```

```
<input name="PIN_OUTPUT2" type="hidden" value="0" />
```

```
<input name="PIN_OUTPUT3" type="hidden" value="0" />
```

```
<input name="PIN_OUTPUT4" type="hidden" value="0" />
```

```
<input name="PIN_OUTPUT5" type="hidden" value="0" />
```

```
<input name="PIN_OUTPUT6" type="hidden" value="0" />
```

```
<input name="" type="submit" value="OFF" />
```

```
</form>
```

```
</td>
```

**// En este caso si pulso el botón de todas las salidas tendre que mandar todos
//comandos PIN_OUTPUTX=0**

```
<td><!--(IF PIN_OUTPUT1=1)--><span class="Estilo1">ALL ON</span><!--(ENDIF)-->
```

```
<!--(IF PIN_OUTPUT1=0)--><span class="Estilo2">ALL OFF</span><!--(ENDIF)-->
```

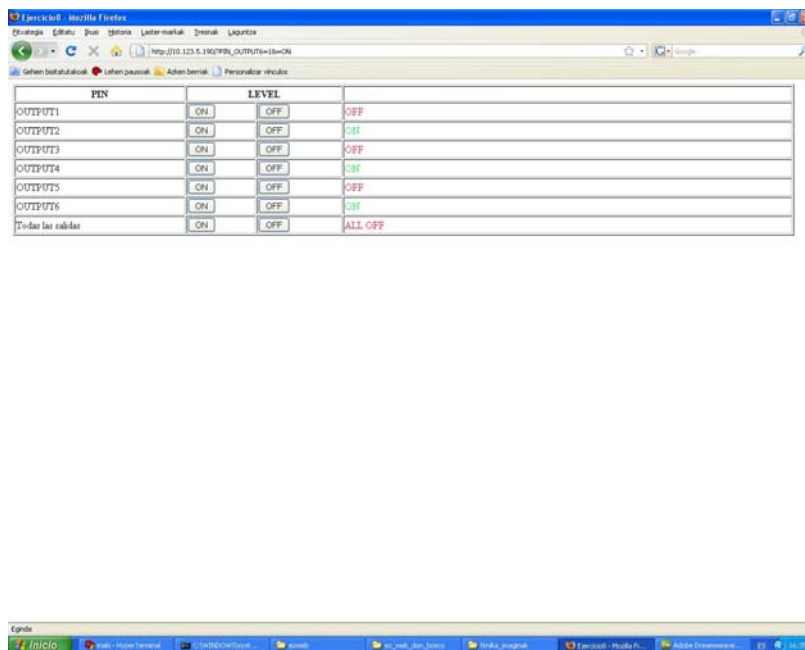
```
</td>
```

```
</tr>
```

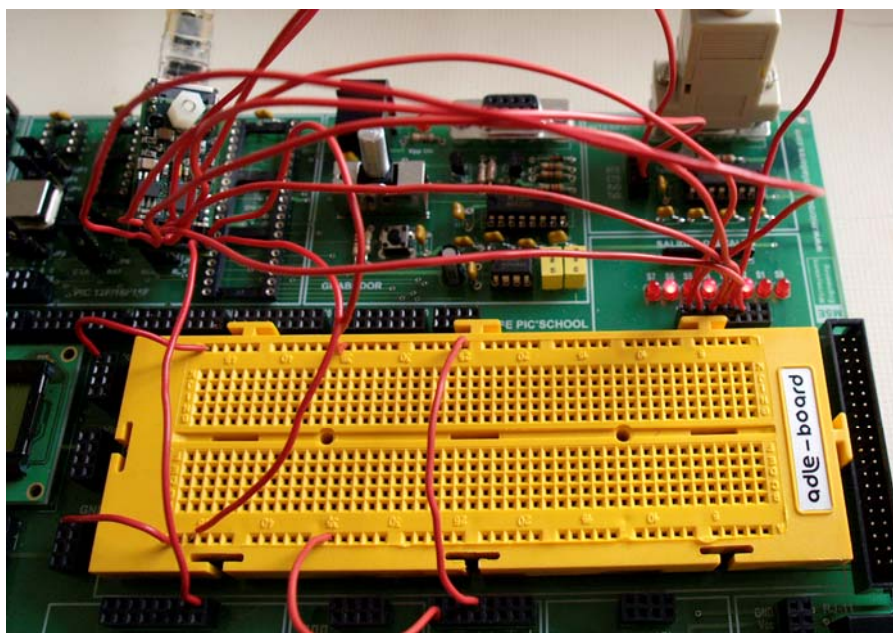
// Visualizare el texto ALL ON o ALL OFF dependiendo lo que vale la salida digital 1

```
</table> </body> </html>
```


En la siguiente imagen podemos ver el ejemplo de la aplicación cuando pulsamos los botones de las salidas digitales 2,4 y 6".



En la siguiente imagen podemos ver la aplicación en marcha después de pulsar los botones de las salidas digitales 2,4 y 6:



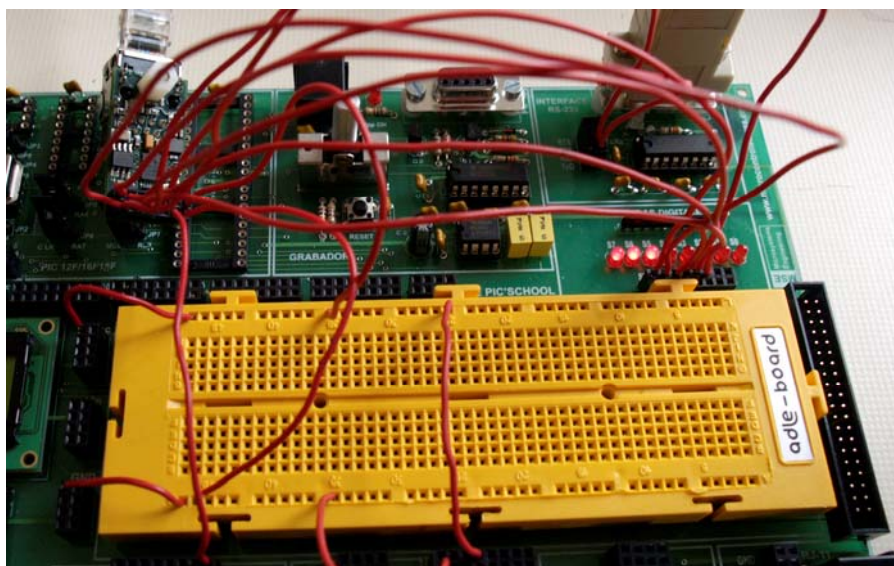
En la siguiente imagen podemos ver el ejemplo de la aplicación cuando pulsamos los botones de las salidas digitales 1,2 y 3.



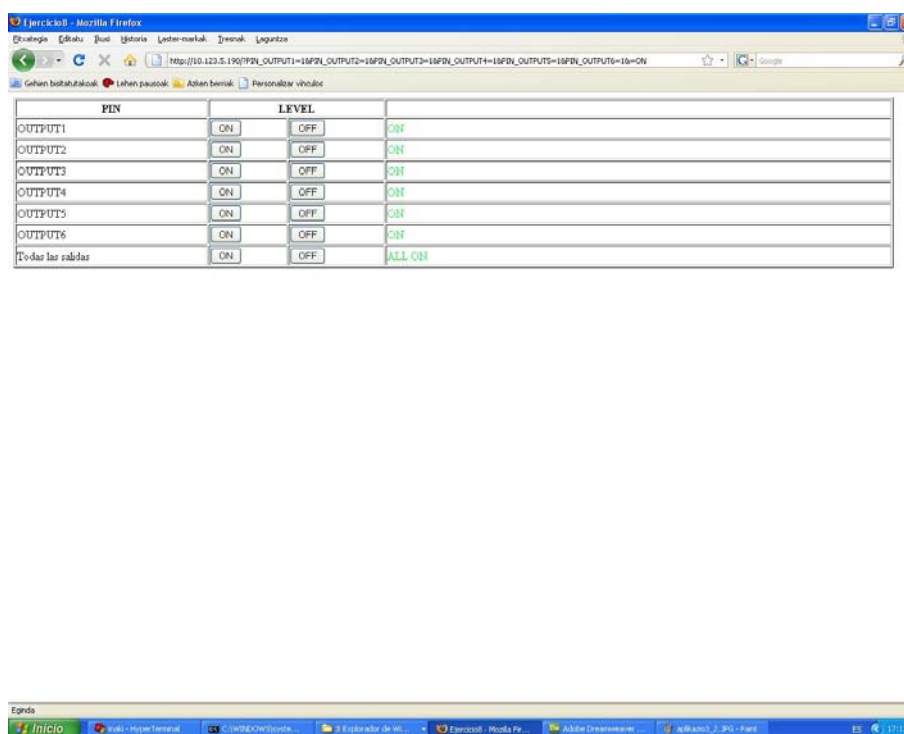
PIN	LEVEL		
OUTPUT1	ON	OFF	ON
OUTPUT2	ON	OFF	ON
OUTPUT3	ON	OFF	ON
OUTPUT4	ON	OFF	OFF
OUTPUT5	ON	OFF	OFF
OUTPUT6	ON	OFF	OFF
Todas las salidas	ON	OFF	ALL ON



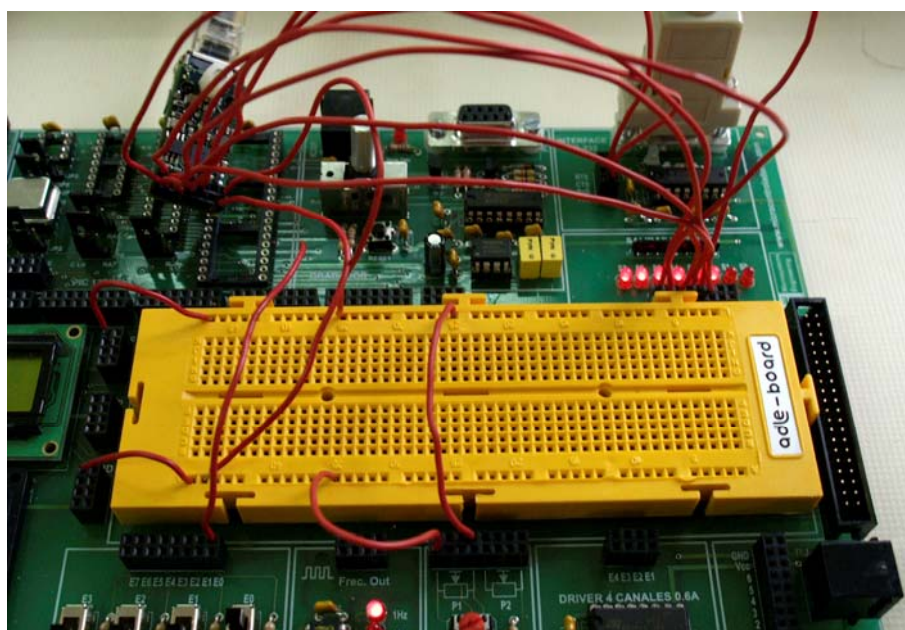
En la siguiente imagen podemos ver la aplicación en marcha después de pulsar los botones de las salidas digitales 1,2 y 3:



En la siguiente imagen podemos ver el ejemplo de la aplicación cuando pulsamos el botón de activar todas las salidas:



En la siguiente imagen podemos ver la aplicación en marcha después de pulsar el botón de activar todas las salidas:



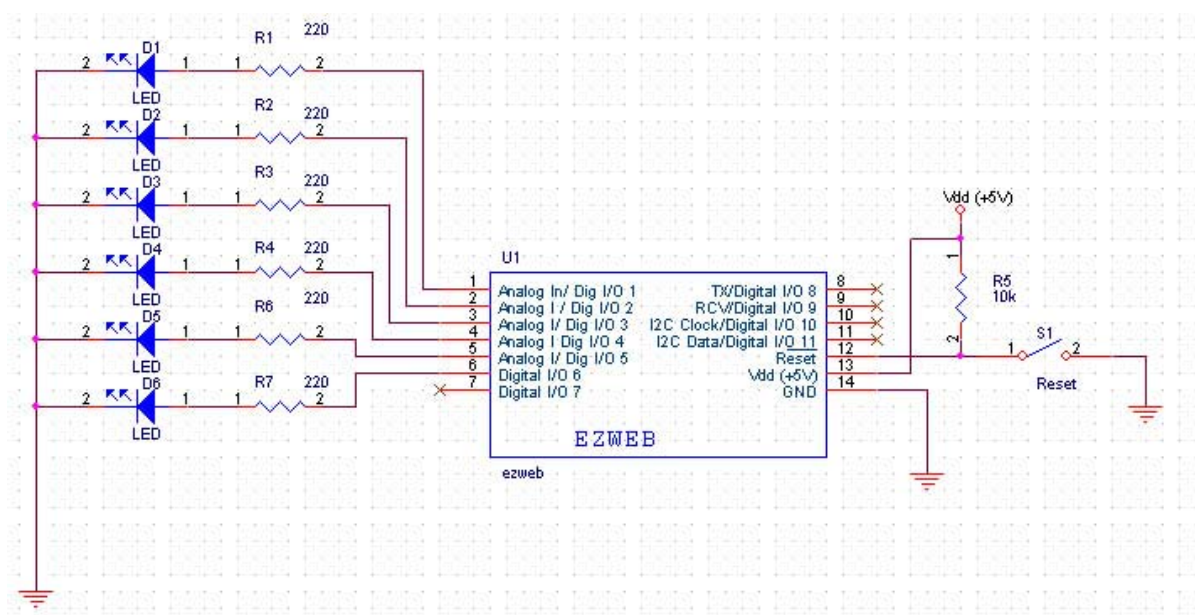
6.4 APLICACIÓN 4 (ACTIVACION DE SALIDAS UTILIZANDO CHECKBOX)

6.4.1 DESCRIPCION

En esta 4ª aplicación seguimos activando salidas pero el formato de las variables de entrada son en formato checkbox.

6.4.2 ESQUEMA DE CONEXIONES

El esquema de conexiones es igual que la aplicación 3 ya que consiste en activar distintas salidas dependiendo de la opción que se elija en el checkbox. Por lo tanto la diferencia estará en la programación de la página web.



6.4.3 PROGRAMA

En el programa de la aplicación 4 hemos introducido una variante a la hora de introducir datos. Para activar y desactivar las salidas, utilizaremos los checkbox.

Para activar y desactivar las salidas tenemos que seguir enviando formularios, pero a la hora de indicar el tipo de formulario hay que indicar que es de tipo checkbox.

El programa de la aplicación 4 es el que a continuación se muestra:

```
<!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>ejercicio4</title>
</head>
<body>
<table width="50%" border="0">
<tr>
<th scope="col">OUTPUTS</th>
</tr>
<tr>
<td><form name="form1" method="get" action="">
<input name="PIN_OUTPUT1" type="checkbox" value="1"><label>PIN1 ON</label><br />
<input name="PIN_OUTPUT2" type="checkbox" value="1"><label>PIN2 ON</label><br />
<input name="PIN_OUTPUT3" type="checkbox" value="1"><label>PIN3 ON</label><br />
<input name="PIN_OUTPUT4" type="checkbox" value="1"><label>PIN4 ON</label><br />
<input name="PIN_OUTPUT5" type="checkbox" value="1"><label>PIN5 ON</label><br />
<input name="PIN_OUTPUT6" type="checkbox" value="1"><label>PIN6 ON</label><br />
//formulario de tipo checkbox para activar las distintas salidas
<input name="" type="submit" value="ENVIAR" />
//formulario de tipo submit correspondiente al botón para enviar los
//comandos al ezweb y activar las distintas salidas
</form>
</td>
</tr>
<tr>
<td><form name="form2" method="get" action="">
<input name="PIN_OUTPUT1" type="checkbox" value="0"><label>PIN1 OFF</label><br />
<input name="PIN_OUTPUT2" type="checkbox" value="0"><label>PIN2 OFF</label><br />
```

<input name="PIN_OUTPUT3" type="checkbox" value="0"><label>PIN3 OFF</label>

<input name="PIN_OUTPUT4" type="checkbox" value="0"><label>PIN4 OFF</label>

<input name="PIN_OUTPUT5" type="checkbox" value="0"><label>PIN5 OFF</label>

<input name="PIN_OUTPUT6" type="checkbox" value="0"><label>PIN6 OFF</label>

//formulario de tipo checkbox para desactivar las distintas salidas

<input name="" type="submit" value="ENVIAR" />

//formulario de tipo submit correspondiente al botón para enviar los

//comandos al ezweb y desactivar las distintas salidas

</form></td>

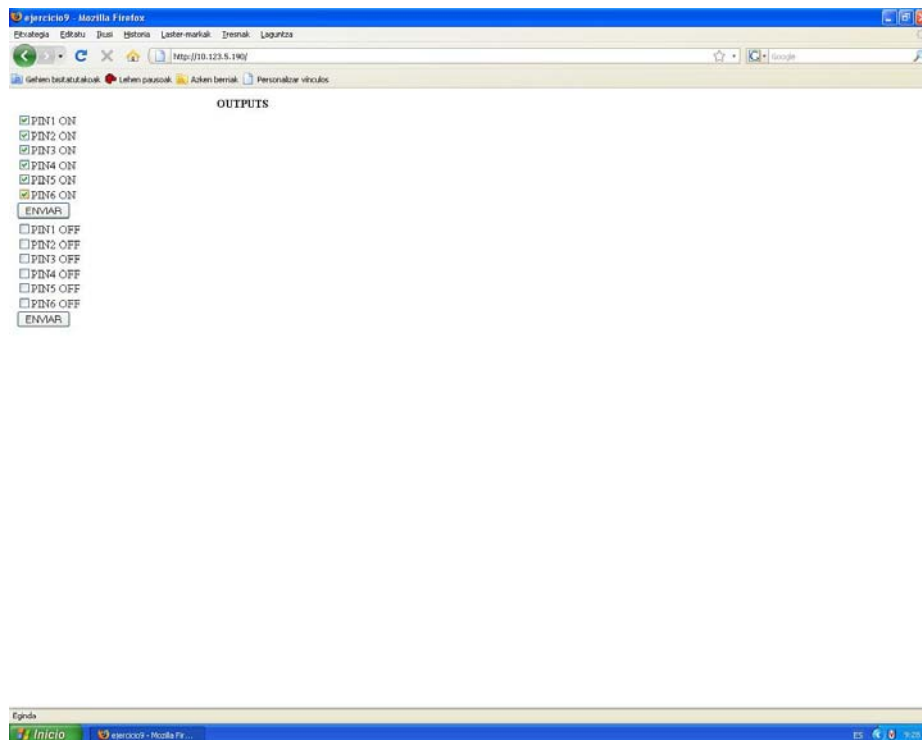
</tr>

</table>

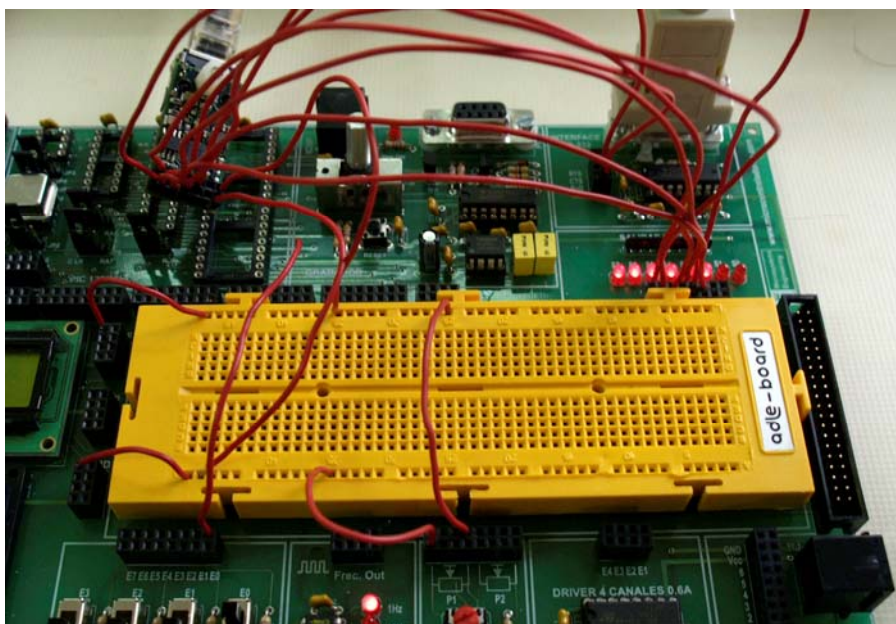
</body>

</html>

En la siguiente imagen podemos ver el ejemplo de la aplicación para activar todas las salidas:



En la siguiente imagen podemos ver la aplicación en marcha después de pulsar el botón de activar todas las salidas:



6.5 APLICACIÓN 5

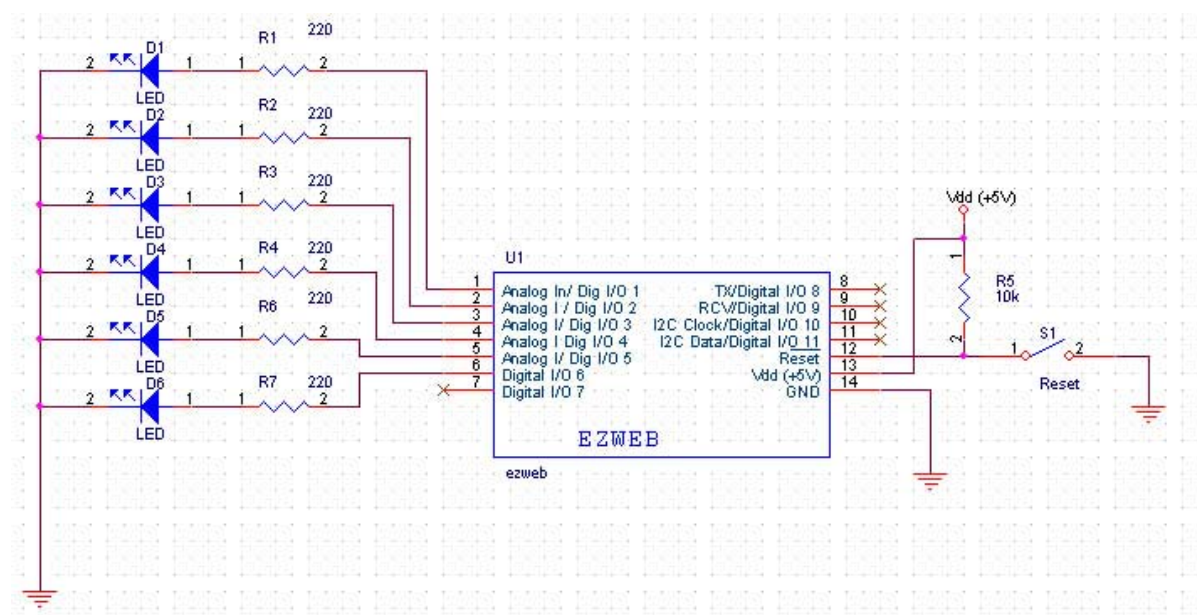
(ACTIVACION DE SALIDAS CON FORMULARIOS TIPO RADIO)

6.5.1 DESCRIPCION

En esta 5ª aplicación seguimos activando salidas pero el formato de las variables de entrada son en formato radio.

6.5.2 ESQUEMA DE CONEXIONES

En esta aplicación utilizaremos el mismo esquema que en la aplicación 4. El esquema es el que a continuación se muestra.



6.5.3 PROGRAMA

En el programa de la aplicación 5 hemos introducido una variante a la hora de introducir datos. Para activar y desactivar las salidas, utilizaremos los formularios de tipo radio.

Para activar y desactivar las salidas tenemos que seguir enviando formularios, pero a la hora de indicar el tipo de formulario hay que indicar que es de tipo radio.

El programa de la aplicación 5 es el que a continuación se muestra:

```
<!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>salida</title>
</head>

<body>
<table width="50%" border="1">
<tr>
<th scope="col">OUTPUTS</th>
</tr>
<tr>
<td><form id="form1" name="form1" method="get" action="">
<input name="PIN_OUTPUT1" type="radio" value="1" /><label>PIN1 ON</label>
<input name="PIN_OUTPUT1" type="radio" value="0" /><label>PIN1 OFF</label>
//formulario de tipo radio para activar y desactivar la salida digital1
<input name="" type="submit" value="ENVIAR" />
//formulario de tipo submit correspondiente al botón para enviar los //comandos al ezweb
</form>
</td>
</tr>
<tr>
<td><form id="form1" name="form1" method="get" action="">
<input name="PIN_OUTPUT2" type="radio" value="1" /><label>PIN2 ON</label>
<input name="PIN_OUTPUT2" type="radio" value="0" /><label>PIN2 OFF</label>
//formulario de tipo radio para activar y desactivar la salida digital2
<input name="" type="submit" value="ENVIAR" />

```

//formulario de tipo submit correspondiente al botón para enviar los //comandos al ezweb

</form></td>

</tr>

<tr>

<td><form id="form1" name="form1" method="get" action="">

<input name="PIN_OUTPUT3" type="radio" value="1" /><label>PIN3 ON</label>

<input name="PIN_OUTPUT3" type="radio" value="0" /><label>PIN3 OFF</label>

//formulario de tipo radio para activar y desactivar la salida digital3

<input name="" type="submit" value="ENVIAR" />

//formulario de tipo submit correspondiente al botón para enviar los //comandos al ezweb

</form></td>

</tr>

<tr>

<td><form id="form1" name="form1" method="get" action="">

<input name="PIN_OUTPUT4" type="radio" value="1" /><label>PIN4 ON</label>

<input name="PIN_OUTPUT4" type="radio" value="0" /><label>PIN4 OFF</label>

//formulario de tipo radio para activar y desactivar la salida digital4

<input name="" type="submit" value="ENVIAR" />

//formulario de tipo submit correspondiente al botón para enviar los //comandos al ezweb

</form></td>

</tr>

<tr>

<td><form id="form1" name="form1" method="get" action="">

<input name="PIN_OUTPUT5" type="radio" value="1" /><label>PIN5 ON</label>

<input name="PIN_OUTPUT5" type="radio" value="0" /><label>PIN5 OFF</label>

//formulario de tipo radio para activar y desactivar la salida digital5

<input name="" type="submit" value="ENVIAR" />

//formulario de tipo submit correspondiente al botón para enviar los //comandos al ezweb

</form></td>

</tr>

<tr>

<td><form id="form1" name="form1" method="get" action="">

<input name="PIN_OUTPUT6" type="radio" value="1" /><label>PIN6 ON</label>

<input name="PIN_OUTPUT6" type="radio" value="0" /><label>PIN6 OFF</label>

//formulario de tipo radio para activar y desactivar la salida digital6



<input name="" type="submit" value="ENVIAR" />

//formulario de tipo submit correspondiente al botón para enviar los //comandos al ezweb

</form></td>

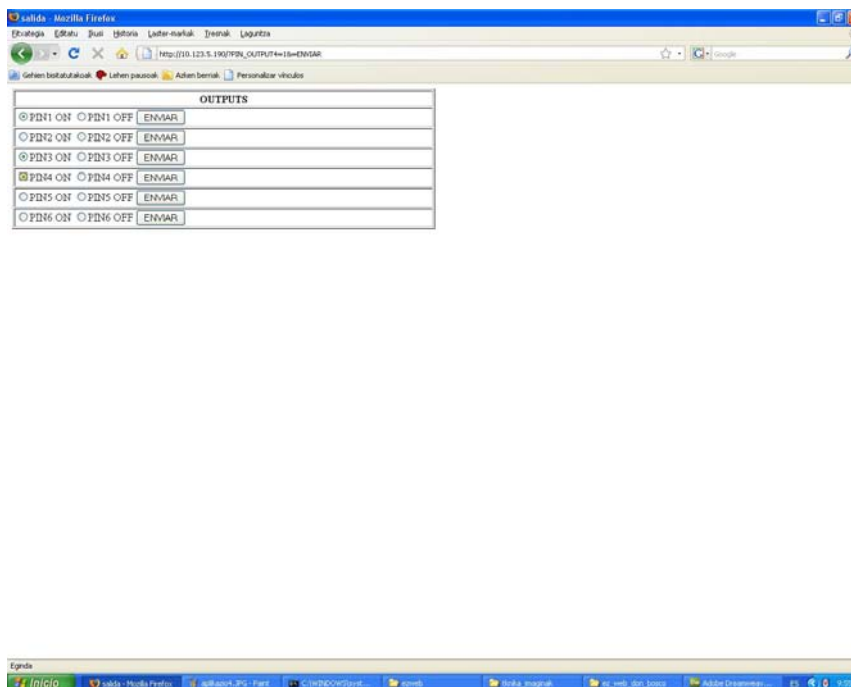
</tr>

</table>

</body>

</html>

En la siguiente imagen podemos ver el ejemplo de la aplicación para activar las salidas 1,3 y 4:



En la siguiente imagen podemos ver la aplicación en marcha después de pulsar el botón los botones de activar las salidas 1,3 y 4:

