

## **ACTIVIDAD N°3:**

# **LECTURA DE ENTRADAS POR WEB**



La luz está:

---

La luminosidad es: 5.00

El porcentaje de luz es: 100.00%

## 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 realizaremos la lectura de distintas entradas conectadas al microservidor EZweb:

- Entradas digitales
- Entradas analógicas

En esta actividad realizaremos actividades que mostraran en una página web el resultado de las entradas digitales que recibe el ezweb. Esta opción nos permite visualizar el estado de distintos dispositivos digitales conectados al microservidor EZweb.

Por otro lado, también podremos visualizar valores analógicos que mostraremos en la página web, así como podremos realizar distintas operaciones según el valor recibido.

En definitiva, podremos visualizar en la página web el estado de las entradas conectadas al microservidor EZweb.

## 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 entradas digitales y analógicas.

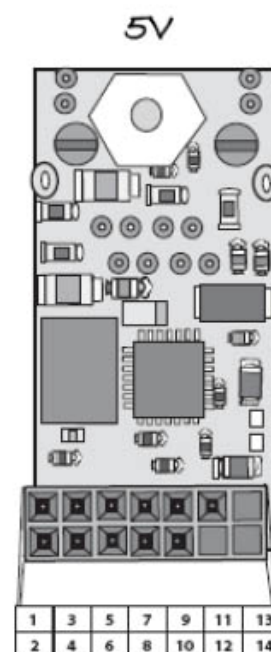
### 4.1 DESCRIPCION DEL MICROSERVIDOR EZWEB

#### 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 <sup>2</sup> C Temp Sensor SCL (clock) line/Digital I/O
11	Dallas DS1621 or DS1631 I <sup>2</sup> 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.

#### 4.1.2 EMPAQUETADO Y SUBIDA DE LA PAGINA WEB

Para poder trabajar con el ezweb, además de realizar el montaje, deberemos de crear una o varias páginas web. En nuestro caso la creación de páginas web la realizaremos con el software dreamweaver.

Una vez creadas las páginas web necesarias, deberemos empaquetar todas las páginas y generar un fichero. Este fichero deberá subirse al microservidor ezweb mediante tftp.

Para poder empaquetar las páginas creadas con dreamweaver, utilizaremos el software csfs.

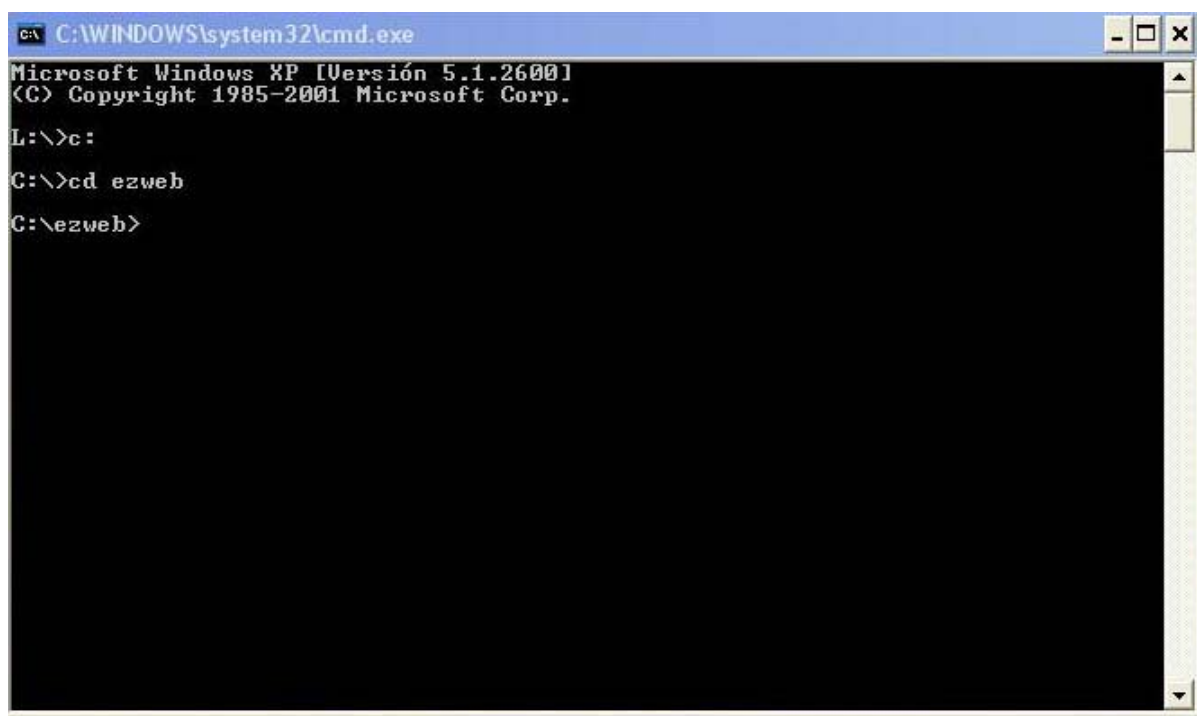
Todas las páginas que creamos deberán estar en un mismo directorio. En nuestro caso hemos creado un directorio llamado ezweb donde colocaremos por cada proyecto un nuevo directorio. El software CSFS que sirve para empaquetar las páginas, estará situado en el directorio ezweb.

Las páginas que creemos deberán de tener extensión htm.

La página de inicio se deberá llamar index.htm

Para realizar el empaquetado y subir la página seguiremos los siguientes pasos:

1. En la ventana de comandos nos colocaremos en el directorio en el que esta el software csfs. En nuestro caso en el directorio ezweb tal como se puede ver en la siguiente imagen:



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Versión 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.
L:\>c:
C:\>cd ezweb
C:\ezweb>
```

Para empaquetar las páginas web creadas deberemos ejecutar el siguiente comando:

**csfs input\_dir output\_file**

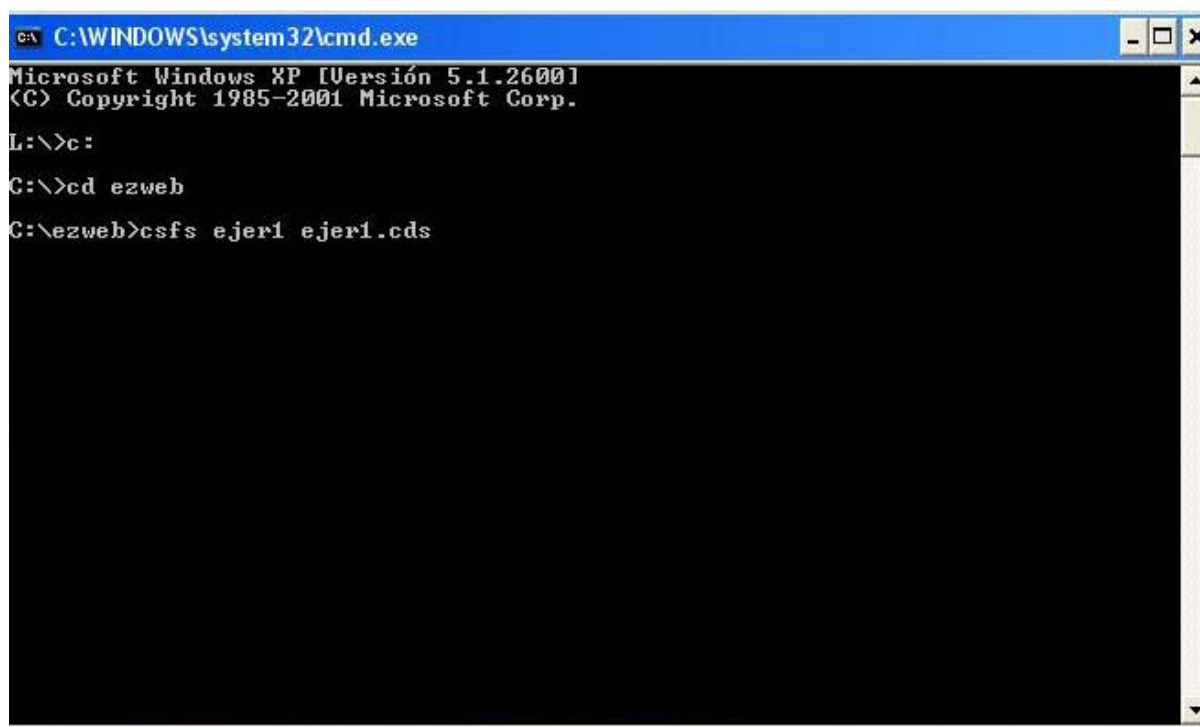
**input\_dir:** es el directorio donde están las páginas web que hemos creado.

**output\_file:** es el nombre del fichero que tenemos que generar. Este fichero tiene que tener extensión .cds

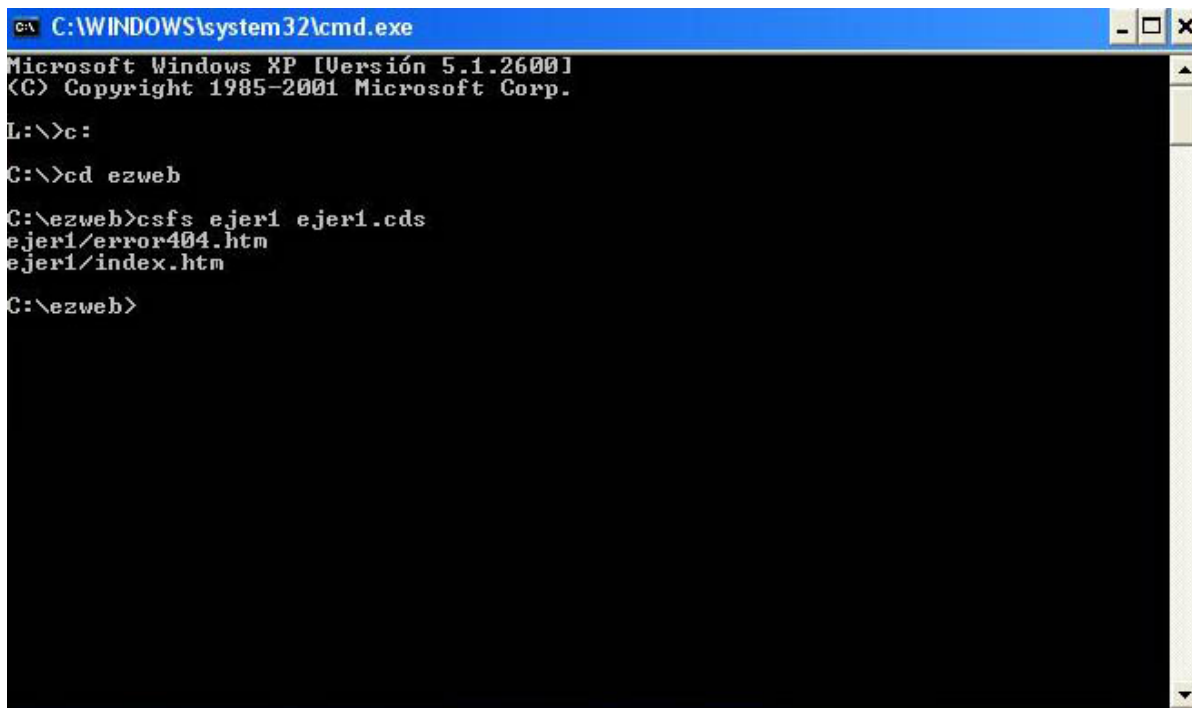
Para empaquetar las páginas que tenemos creadas en el directorio ejer1 ejecutaremos el comando de la siguiente manera:

**csfs ejer1 ejer1.cds**

Tal como podemos ver en la siguiente imagen:



Después de ejecutar este comando deberemos de ver la siguiente imagen:



```

C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Versión 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

L:\>c:

C:\>cd ezweb

C:\ezweb>csfs ejer1 ejer1.cds
ejer1/error404.htm
ejer1/index.htm

C:\ezweb>

```

En la imagen podemos ver las páginas que se han empaquetado generando el fichero ejer1.cds. En este caso constarían de las páginas index.htm y de la página error404.htm.

Una vez empaquetados las páginas deberemos subir las páginas al microservidor ezweb mediante tftp utilizando el siguiente comando:

**tftp -i IP\_address put Local\_File image.cds**

**IP\_address:** Aquí indicamos la dirección IP del dispositivo

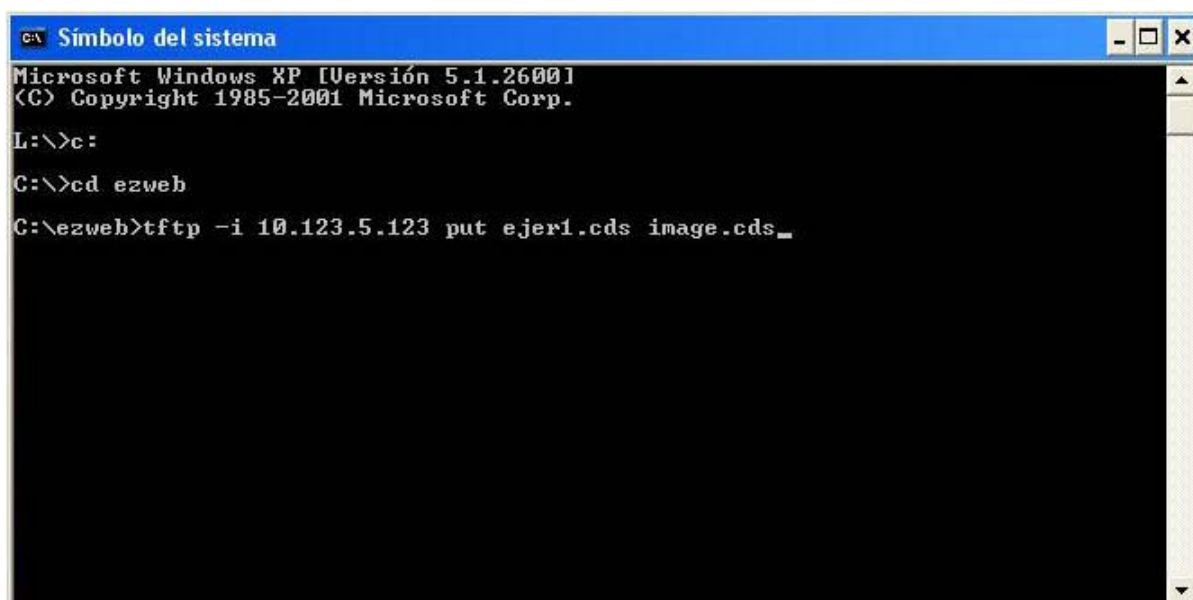
**Local\_File:** Aquí indicamos el fichero que se ha generado con el software csfs, es decir, el fichero creado con extensión .cds

Para subir el fichero que hemos generado ejecutaremos el comando de la siguiente manera:

**tftp -i 10.123.5.123 put ejer1.cds image.cds**

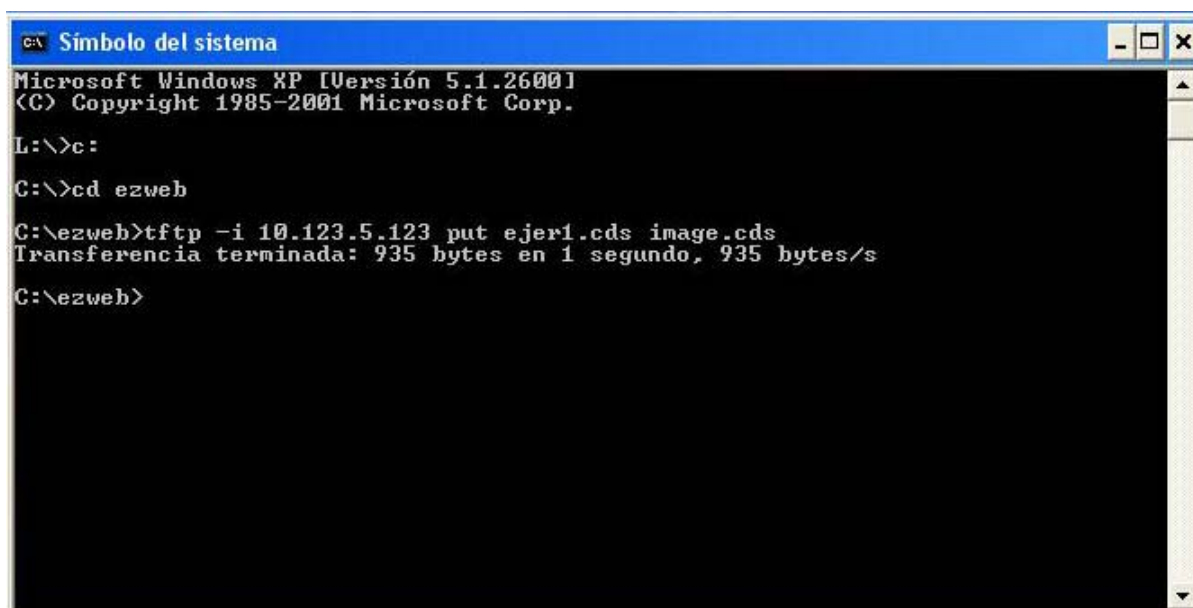
En nuestro caso la dirección del dispositivo es 10.123.5.123.

En la siguiente imagen podemos ver como hemos ejecutado el comando:



```
C:\>cd ezweb
C:\ezweb>tftp -i 10.123.5.123 put ejer1.cds image.cds_
```

Por ultimo, veremos la siguiente imagen cuando la página se ha subido al microservidor ezweb:



```
C:\>cd ezweb
C:\ezweb>tftp -i 10.123.5.123 put ejer1.cds image.cds
Transferencia terminada: 935 bytes en 1 segundo, 935 bytes/s
C:\ezweb>
```



## 4.2 COMANDOS E INSTRUCCIONES A UTILIZAR

### 4.2.1 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.

### 4.2.2 COMANDOS

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

**PIN\_INPUTxx:** Cuando se manda leer este comando, nos devuelve el valor de la entrada digital. Un "0" en caso de que la entrada sea "0" y un "1" en caso contrario.

**PIN\_ANALOGxx:** Cuando se manda leer este comando, nos devuelve el valor de la tensión analógica especificada en la entrada analógica correspondiente.



Estos dos comandos se leen con la instrucción **<!--(READ KEY)-->** donde la palabra KEY se sustituye por el comando a utilizar.

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

**<!--(READ PIN\_INPUT01)-->** Este comando leerá el valor de la entrada digital01.

**<!--(READ PIN\_ANALOG01)-->** Este comando leerá el valor de la entrada analógica01.

## 5 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.

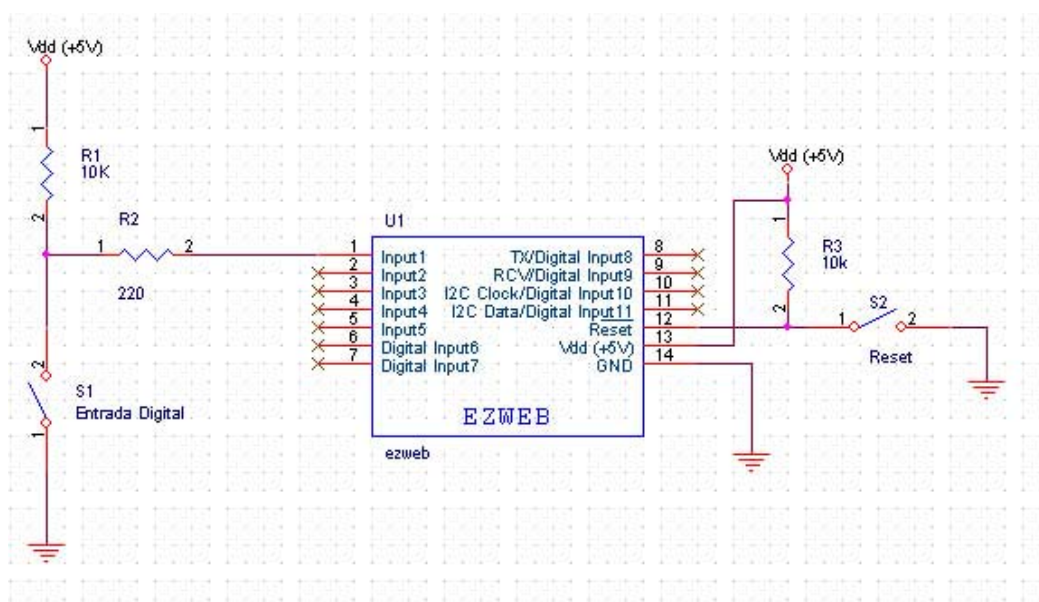
### 5.1 APLICACIÓN 1 (LECTURA DE ENTRADA DIGITAL)

#### 5.1.1 DESCRIPCION

La primera aplicación que realizaremos tiene como objetivo el leer una entrada digital, y visualizar en la página el valor de dicha entrada digital.

#### 5.1.2 ESQUEMA DE CONEXIONES

En la siguiente imagen mostraremos el esquema de conexiones para leer una entrada digital y mostrar su valor en la página web.



### 5.1.3 PROGRAMA

El programa de la aplicación 1 es el correspondiente a la creación de la página web en la cual usaremos uno de los comandos del Ezweb.

El comando a utilizar es `<!--(PIN X)-->` donde X es el número del pin donde conectamos la entrada digital.

En la página web se sustituirá el comando `<!--(PINX)-->` por el valor de la entrada digital que en este caso será "0" o "1".

El programa es el que a continuación describimos:

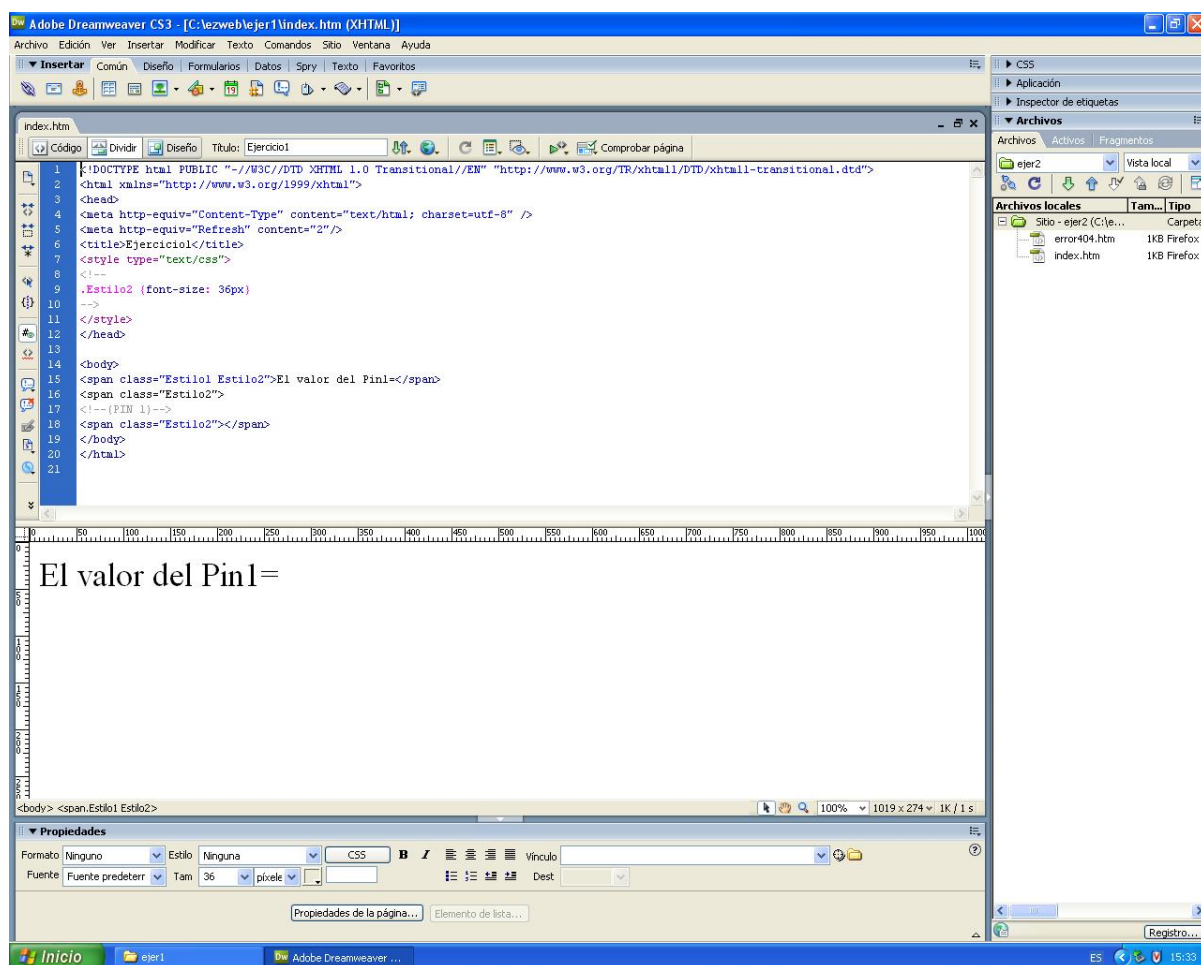
```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<meta http-equiv="Refresh" content="2"/>           //comando que sirve para que la
                                                    //página se refresque cada 2 //segundos.

<title>Ejercicio1</title>
<style type="text/css">
<!--
.Estilo1 {
    font-size: 18px;
    font-weight: bold;
}
-->
</style>
</head>

<body>
<span class="Estilo2">El valor del Pin1=
<span class="Estilo2">
<!--(PIN 1)-->           //Colocamos en la página web el siguiente texto:
                        //El valor del Pin1= //
                        //después colocamos el comando del ezweb que lee //el
                        //valor de la entrada digital conectada en la //entrada1.
                        //Cuando leamos la página el valor de la //entrada
                        //sustituirá al comando.

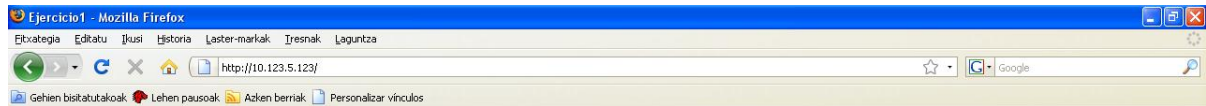
<span class="Estilo2"></span>
</body>
</html>
```

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 un "1".

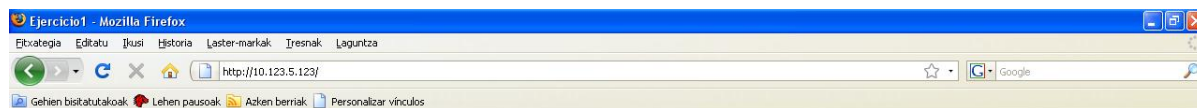


El valor del Pin1= 1





Y en la siguiente imagen podremos ver la aplicación cuando en la entrada introducimos un "0"



El valor del Pin1=0



## 5.2 APLICACIÓN 2 (LECTURA DE ENTRADA DIGITAL CON ETIQUETAS)

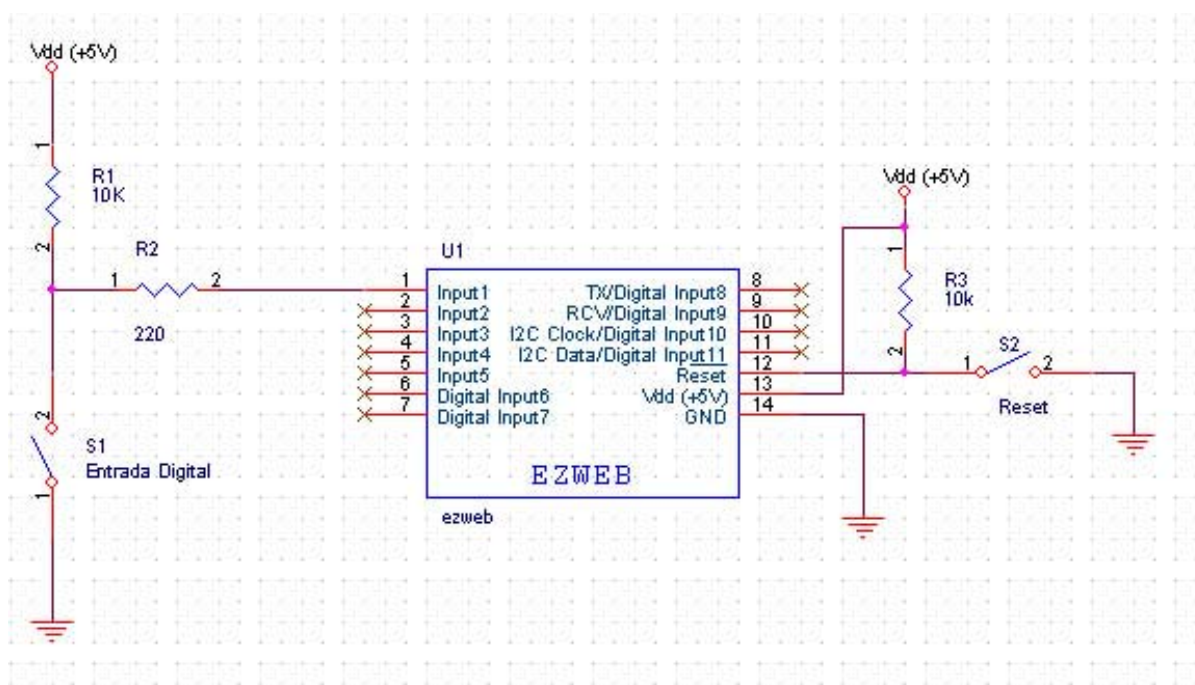
### 5.2.1 DESCRIPCION

En esta 2ª aplicación volveremos a realizar el mismo montaje pero en la página web mostraremos un texto dependiendo del valor que leamos en la entrada digital.

En este caso mostraremos el texto desactivado cuando leamos un "0" y mostraremos el texto "activado" cuando leamos un "1".

### 5.2.2 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 leyendo una entrada digital y añadimos al programa un texto dependiendo el valor que estamos recibiendo.

### 5.2.3 PROGRAMA

El programa de la aplicación 2 es el correspondiente a la creación de la pagina web en la cual usaremos uno de los comandos del Ezweb.

El comando a utilizar es `<!—(PIN X ltag/htag)-->` donde X es el número del pin donde conectamos la entrada digital, ltag es la etiqueta que queremos ver cuando la entrada vale "0" y htag es la etiqueta correspondiente al pin cuando este vale "1"

En la página web colocara la etiqueta correspondiente dependiendo del valor que reciba. En caso de recibir un "0" el programa asignara la etiqueta puesta en lugar de ltag y en caso de leer un "1" colocara la etiqueta que esta puesta en lugar de htag.

El programa es el que a continuación describimos:

```
<!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=utf-8" />
<meta http-equiv="Refresh" content="2"/>           //comando que sirve para que la página se
                                                    //refresque cada 2 segundos.

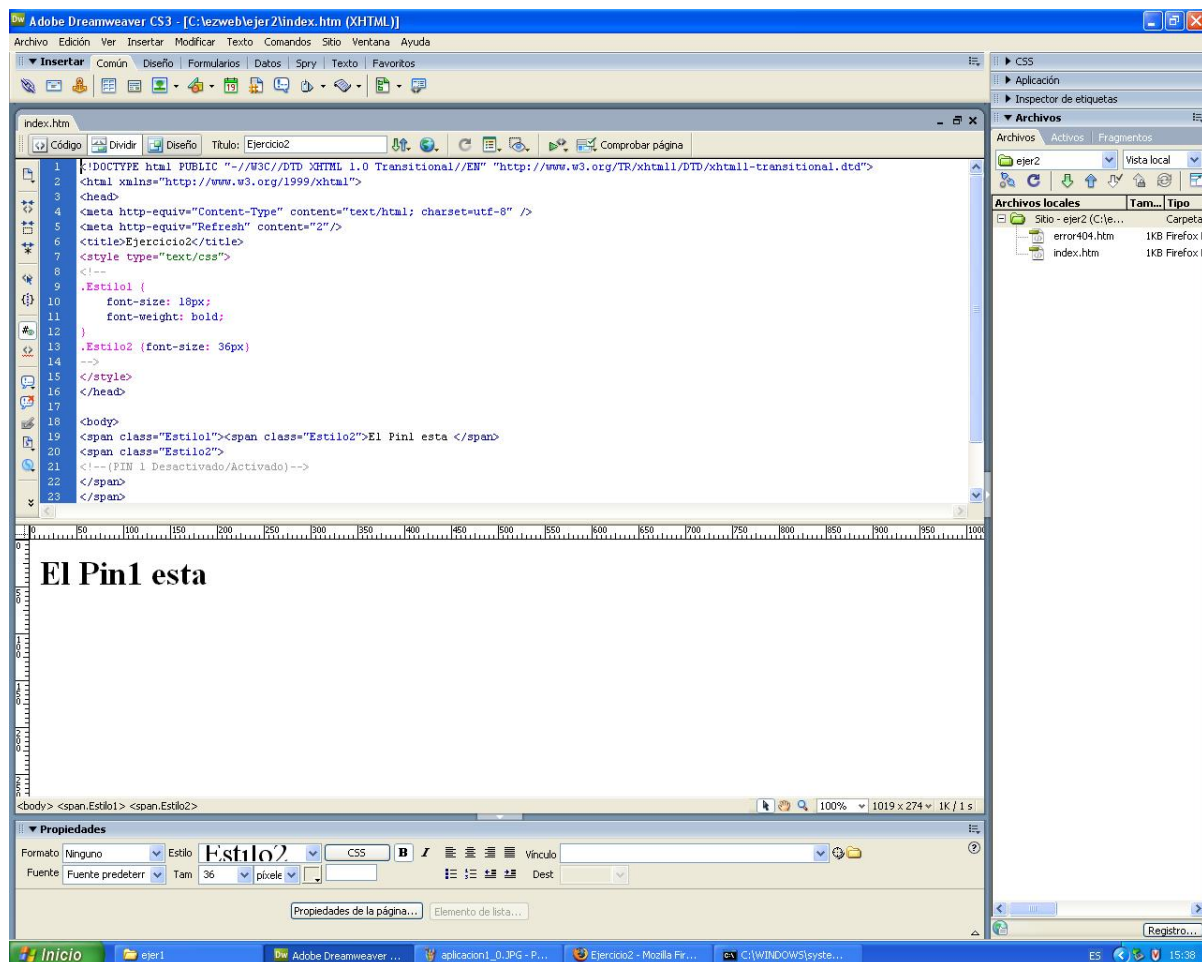
<title>Ejercicio2</title>
<style type="text/css">
<!--
.Estilo1 {
    font-size: 18px;
    font-weight: bold;
}
.Estilo2 {font-size: 36px}
-->
</style>
</head>

<body>
<span class="Estilo1"><span class="Estilo2">El Pin1 esta </span>
<span class="Estilo2">
<!--(PIN 1 Desactivado/Activado)-->           //Colocamos en la pagina web el siguiente texto:
                                                    //El Pin1 esta
                                                    //después colocamos el comando del ezweb que lee //el
                                                    //valor de la entrada digital conectada en la //entrada1 y
                                                    //dependiendo de su valor le asigna una //etiqueta. En este
                                                    //caso cuando la entrada vale "0" le asigna la etiqueta
                                                    //Desactivado. Si la entrada vale //1" le asigna la etiqueta
                                                    //Activado.

</span>
</span>
```

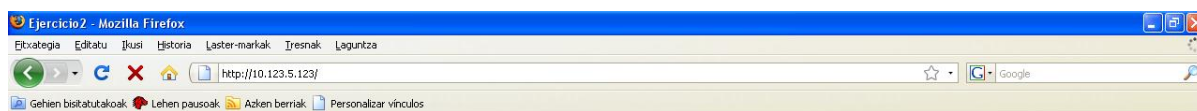


En la siguiente imagen podremos ver el programa escrito con dreamweaver:





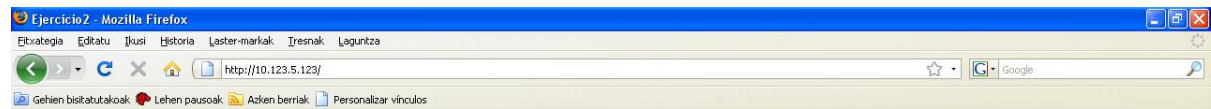
En la siguiente imagen podemos ver el ejemplo de la aplicación cuando en la entrada introducimos un "1".



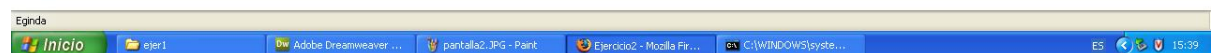
**El Pin1 esta Activado**



Y en la siguiente imagen podremos ver la aplicación cuando en la entrada introducimos un "0"



**El Pin1 esta Desactivado**



### 5.3 APLICACIÓN 3 (LECTURA DE ENTRADA DIGITAL CON IMAGENES)

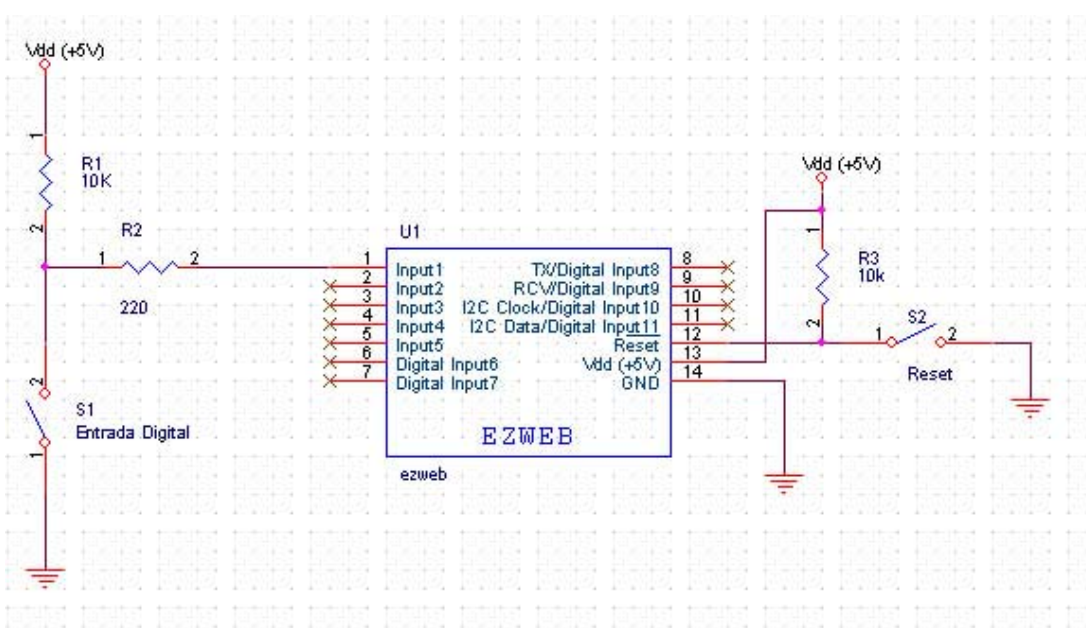
#### 5.3.1 DESCRIPCION

En esta 3ª aplicación volveremos a realizar el mismo montaje pero en la página web mostraremos una imagen.

Mostraremos un interruptor abierto en caso de que leamos en la entrada digital un "0" y mostraremos un "1" en caso de leer un "1" en la entrada digital.

#### 5.3.2 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 leyendo una entrada digital y añadimos al programa la parte que dependiendo del valor de la entrada digital mostraremos una u otra imagen.

En este caso, cuando recibimos en la entrada digital un "0", mostraremos la imagen de un interruptor abierto:



En el caso de recibir en la entrada digital un "1", mostraremos la imagen de un interruptor cerrado:



### 5.3.3 PROGRAMA

El programa de la aplicación 3 es el correspondiente a la creación de la página web en la cual usaremos uno de los comandos del Ezweb.

En este caso debemos leer el valor introducido en la entrada digital 1 y lo compararemos con un valor. Para comparar con un valor determinado usaremos el comando `<!--(IF PIN x = valor)-->`. Si se cumple el valor le diremos que ejecute el comando correspondiente. Con este comando valor solo puede tomar valores digitales, esto es, "0" o "1".

Para que ejecute la opción (si no) utilizaremos el comando `<!--(ELSE)-->`. Este comando se ejecutará cuando la opción IF anteriormente puesta no se cumple.

Para terminar colocaremos el comando `<!--(ENDIF)-->`.

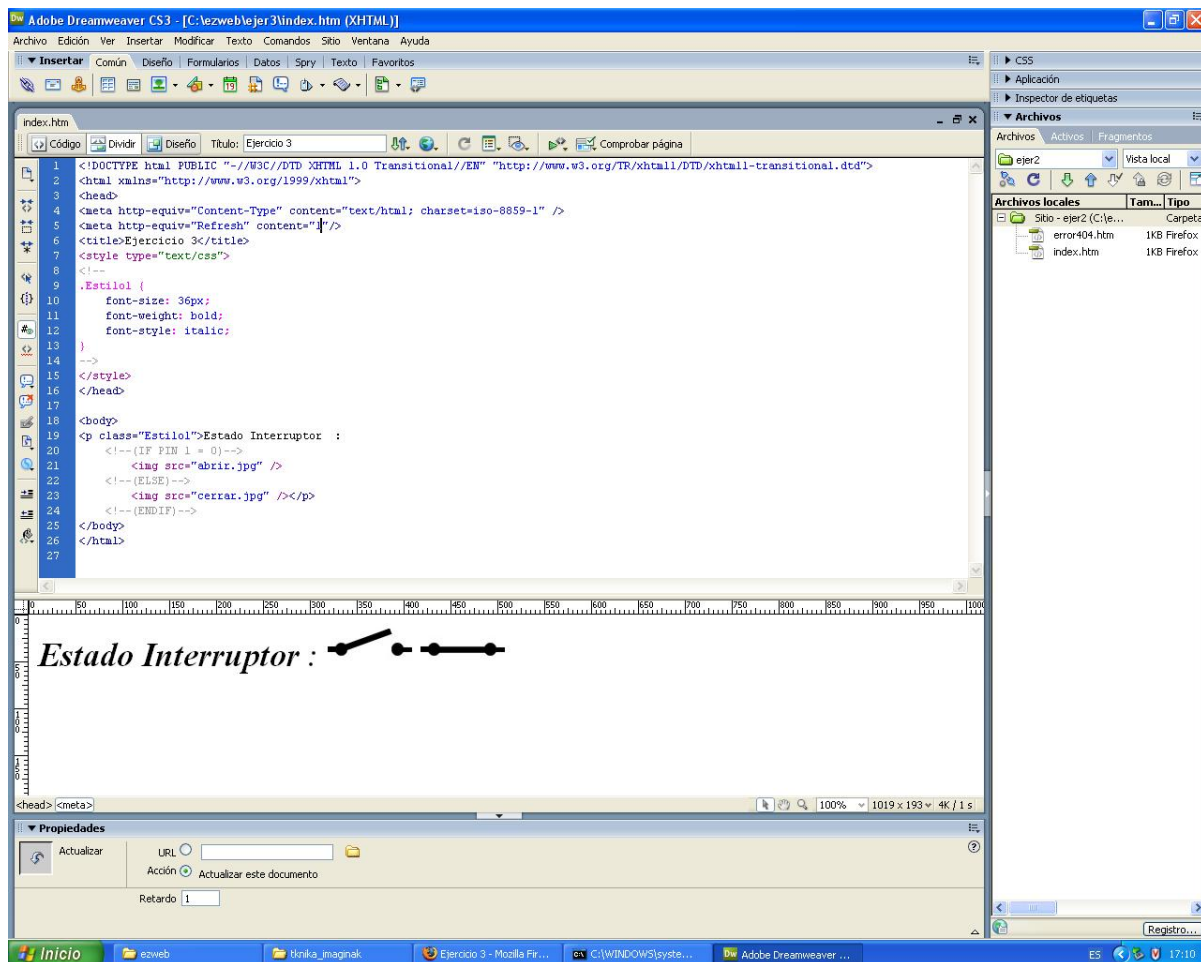
El programa de la aplicación 3 se muestra a continuación:

```
<!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" />
<meta http-equiv="Refresh" content="1"/>           //comando que sirve para que la //página
                                                    se refresque cada 1 //segundos.

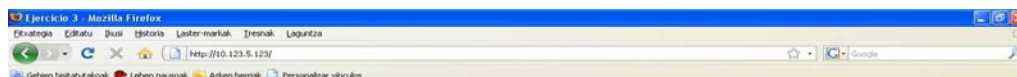
<title>Ejercicio 3</title>
<style type="text/css">
<!--
.Estilo1 {
    font-size: 36px;
    font-weight: bold;
    font-style: italic;
}
-->
</style>
</head>

<body>
<p class="Estilo1">Estado Interruptor :
    <!--(IF PIN 1 = 0)-->           //si la entrada digital 1 vale "0"
            //colocar la imagen que se llama abrir.
    <!--(ELSE)-->           //Si no colocar la imagen que se llama
        </p> //cerrar.
    <!--(ENDIF)-->
</body>
</html>
```

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



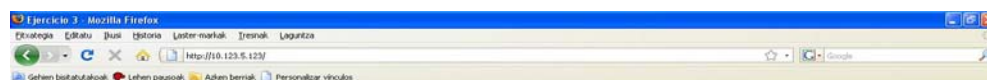
En la siguiente imagen podemos ver el ejemplo de la aplicación cuando en la entrada introducimos un "1".



*Estado Interruptor :* —●—



En la siguiente imagen podemos ver el ejemplo de la aplicación cuando en la entrada introducimos un "0":



*Estado Interruptor :* ●—





## 5.4 APLICACIÓN 4 (LECTURA DE ENTRADA ANALÓGICA)

### 5.4.1 DESCRIPCION

En esta 4ª aplicación realizaremos la lectura de una entrada analógica y la mostraremos con distintos colores.

Si la entrada analógica es mayor de 2,5V, mostraremos el valor analógico en color verde, en caso contrario mostraremos el valor de la entrada analógica en color rojo.

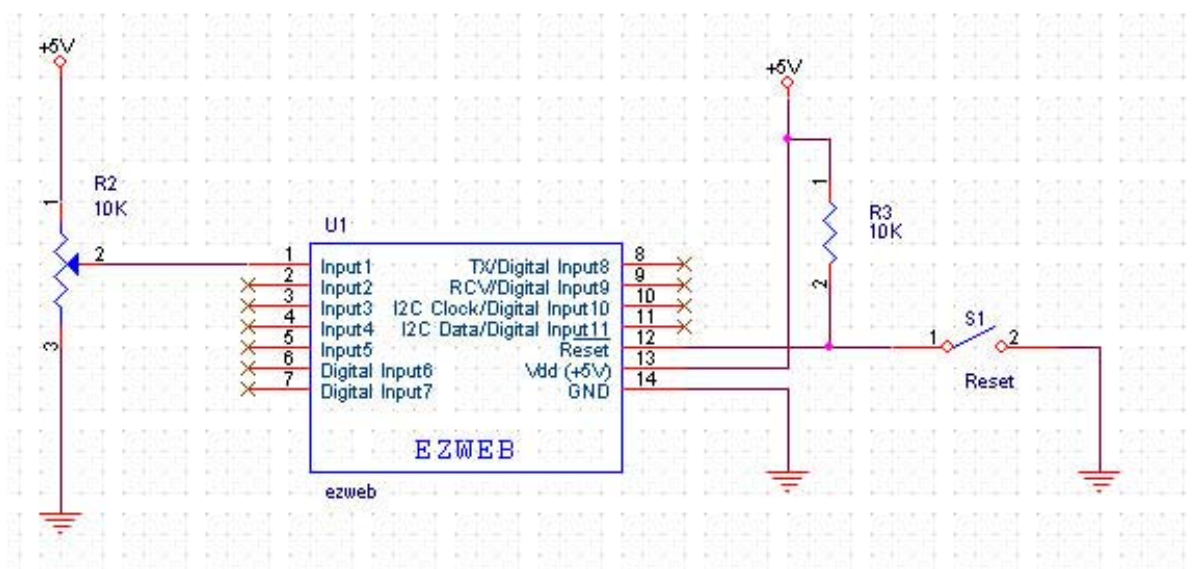
Recordar que en los casos de entradas analógicas, las entradas analógicas deben tener prioridad con respecto las digitales.  
Se debe de usar la primera entrada analógica libre. En este caso al usar una entrada analógica se debe conectar en la pata 1, que es la entrada analógica 1.

Las entradas analógicas tienen el rango GND-Vdd con una resolución de 0,02V en el módulo de 5V.

### 5.4.2 ESQUEMA DE CONEXIONES

Tal como podemos en la siguiente imagen, en esta aplicación usaremos un potenciómetro para introducir una tensión analógica que variara entre 0 y 5 voltios de continua.

En este caso tendremos el siguiente esquema de conexiones.



### 5.4.3 PROGRAMA

El programa de la aplicación 4 es el correspondiente a la creación de la página web en la cual usaremos uno de los comandos del Ezweb para poder leer las entradas analógicas.

```
<!--(PIN x ANALOG)-->
```

En la página web, este comando se sustituye por el valor recibido en la entrada analógica con dos decimales.

Para asignar un color a las letras, deberemos de asignar el valor en hexadecimal del color al parámetro font color.

En esta 4 aplicación, aparte de leer el valor analógico, también debemos de realizar una comparación con un valor 2,5V (valor a partir del cual cambiamos de color). Esta comparación la realizaremos con el comando <!--(IF PIN 1>2,5)-->.

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>Ejercicio 4</title>
<style type="text/css">
<!--
.Estilo2 {
    font-size: 24px;
    color: #000099;
}
.Estilo3 {color: #FF0000}
.Estilo4 {color: #00FF00}
-->
</style>
<meta http-equiv="Refresh" content="1"/> //refresco de página cada segundo
</head>

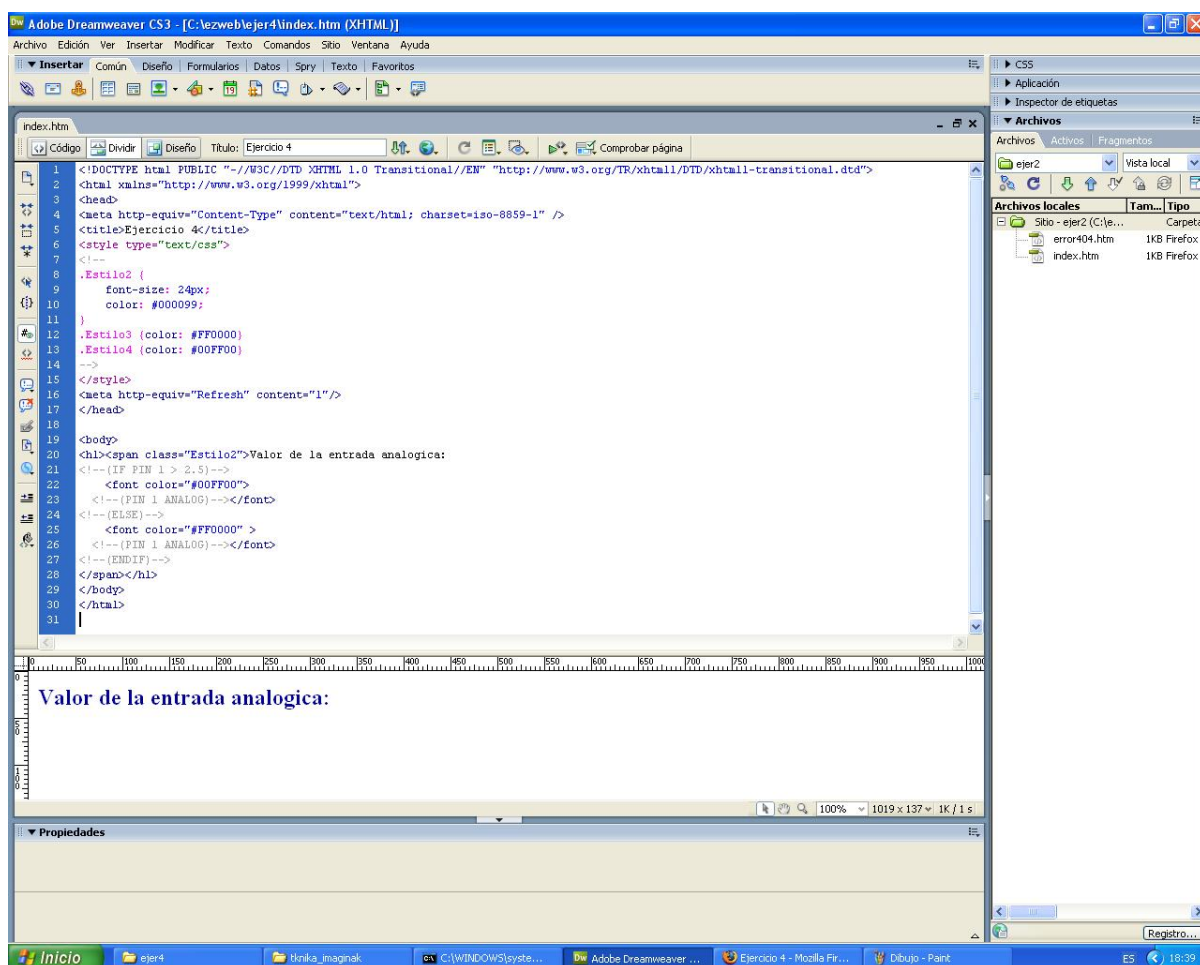
<body>
<h1><span class="Estilo2">Valor de la entrada analógica:
<!--(IF PIN 1 > 2.5)--> //Si el valor leído es mayor que 2,5V mostrar en
    <font color="#00FF00"> //color verde (color de texto)
    <!--(PIN 1 ANALOG)--></font> //valor de la entrada analógica
<!--(ELSE)--> //Si no, mostrar
    <font color="#FF0000" > //color rojo (color de texto)
```

```

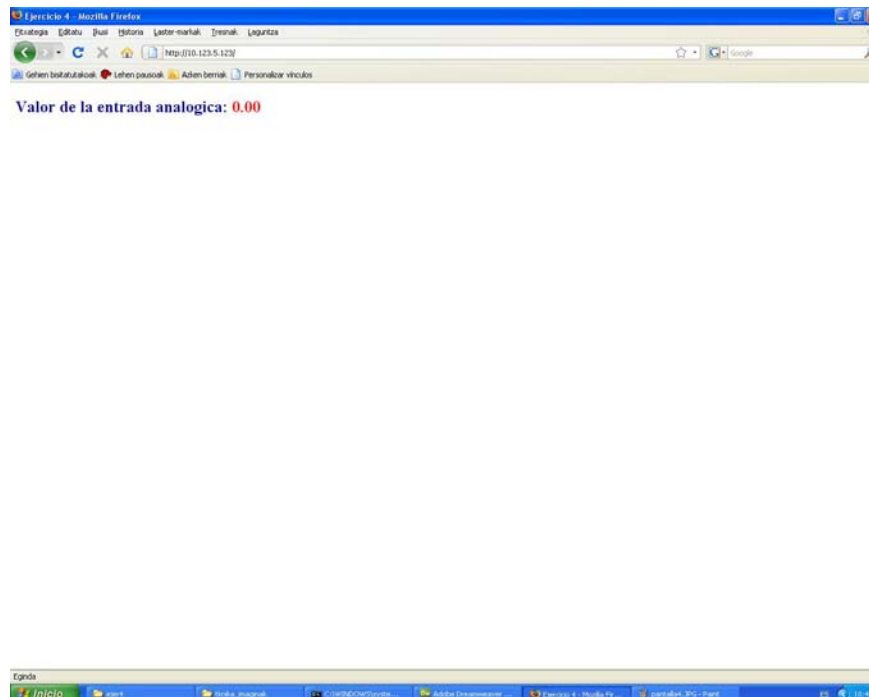
<!--(PIN 1 ANALOG)--></font> //valor de la entrada analógica
<!--(ENDIF)-->
</span></h1>
</body>
</html>

```

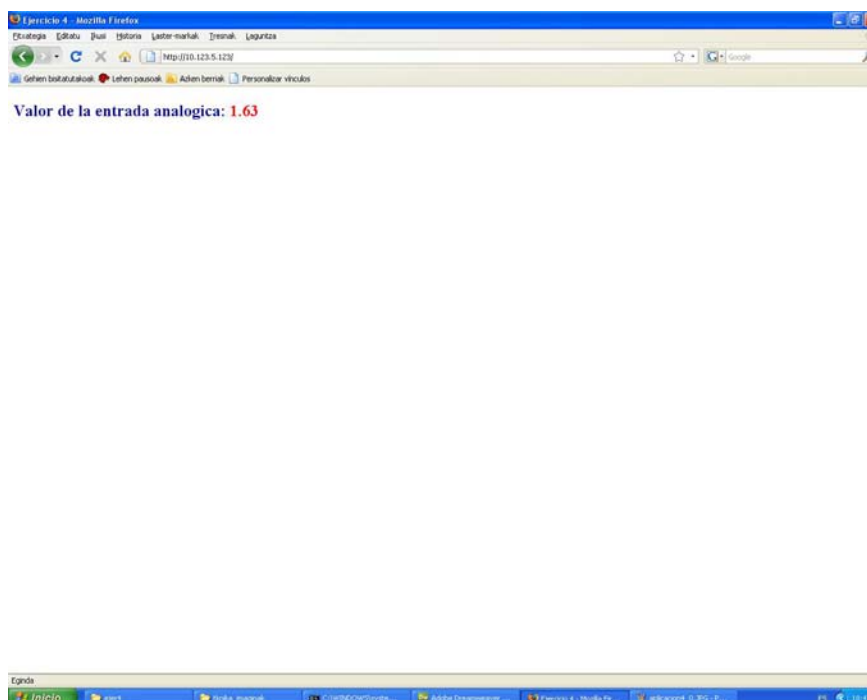
En la siguiente imagen podremos ver el programa escrito con dreamweaver:



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



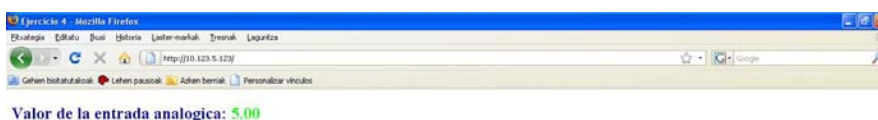
En la siguiente imagen podemos ver el ejemplo de la aplicación cuando en la entrada introducimos una señal analógica de 1,63V:



En la siguiente imagen podemos ver el ejemplo de la aplicación cuando en la entrada introducimos una señal analógica de 3,20V:



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



## 5.5 APLICACIÓN 5 (LECTURA DE ENTRADA ANALÓGICA CON IMAGENES)

### 5.5.1 DESCRIPCION

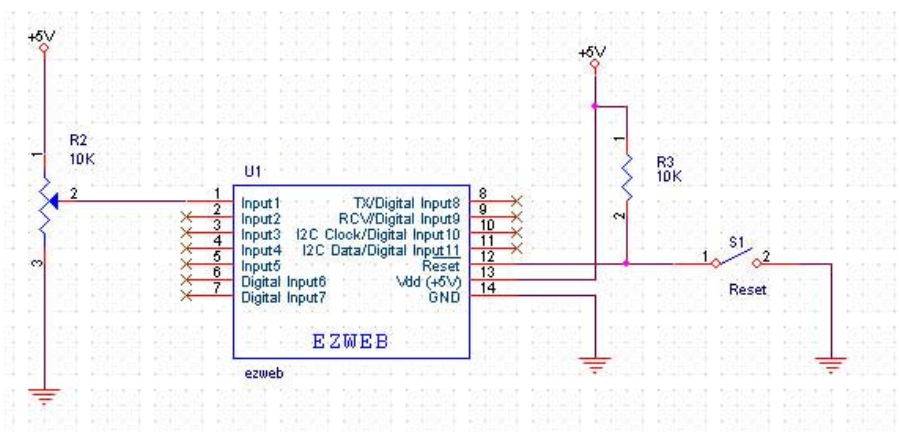
En esta 5ª aplicación realizaremos la lectura de una entrada analógica y realizaremos un ajuste de valores dependiendo de la entrada analógica que recibamos.

También se mostrara una imagen dependiendo de la entrada analógica que recibamos. Como usaremos el comando `<--(IF PIN x = n)-->`, n solo puede ser un valor digital. En este caso la señal analógica funcionara como si fuera una digital.

Cuando realicemos la lectura de la entrada analógica con el comando `<!--(PIN X ANALOG)-->`, recibiremos el valor de la señal analógica.

### 5.5.2 ESQUEMA DE CONEXIONES

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



### 5.5.3 PROGRAMA

En esta aplicación veremos un nuevo comando que nos permitirá realizar operaciones aritméticas con la señal analógica.

En este caso necesitamos realizar un ajuste de la señal analógica para que podamos visualizar el porcentaje de luz y reflejarlo en tanto por ciento. Para ello necesitamos multiplicar la señal analógica por 20.

El comando a utilizar es el que se muestra a continuación:

`<!--(PIN x ANALOG*a/b+c-d,p)-->` donde después de la palabra ANALOG mostraremos la operación aritmética necesaria y p es el número de decimales con el que queremos representar el resultado.

En este ejemplo, necesitamos multiplicar por 20 y queremos representar con 0 decimales. Por lo tanto el comando necesario queda de la siguiente manera:

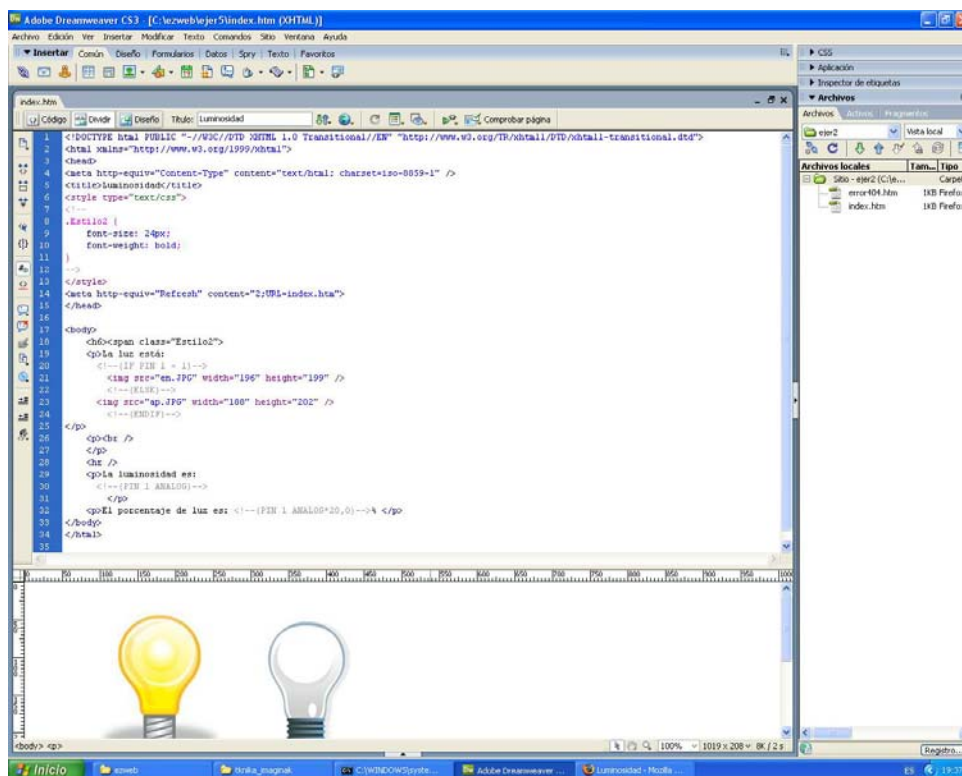
`<!--(PIN x ANALOG*20,0)-->`

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

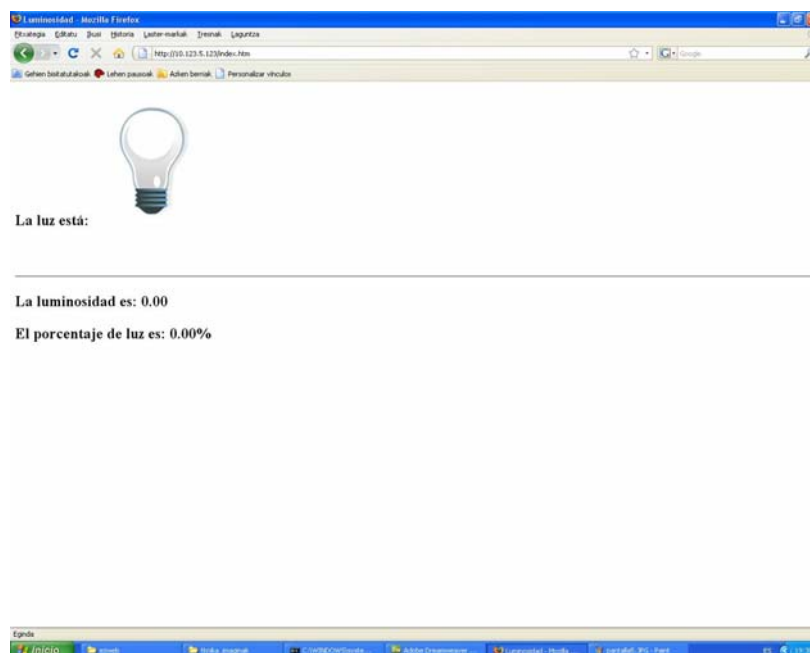
```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Luminosidad</title>
<style type="text/css">
<!--
.Estilo2 {
    font-size: 24px;
    font-weight: bold;
}
-->
</style>
<meta http-equiv="Refresh" content="2;URL=index.htm"> //refresco de web cada
                                                    //2 segundos
</head>
<body>
    <h6><span class="Estilo2">
    <p>La luz está:
        <!--(IF PIN 1 = 1)-->          // Si PIN 1 es 1 (toma como entrada digital)
             //muestra imagen de
                                                    //lampara encendida
        <!--(ELSE)-->                // Si no muestra
             // imagen lámpara apagada
        <!--(ENDIF)-->
    </p>
    <p><br />
</p>
    <hr />
    <p>La luminosidad es:
        <!--(PIN 1 ANALOG)-->        // mostrar valor de entrada analogica
            </p>
    <p>El porcentaje de luz es: <!--(PIN 1 ANALOG*20,0)-->% </p>
                                                    // Se realiza el ajuste de entrada analógica multiplicando por 20 y
                                                    // mostrando el resultado con 0 decimales.
</body>
</html>
```



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

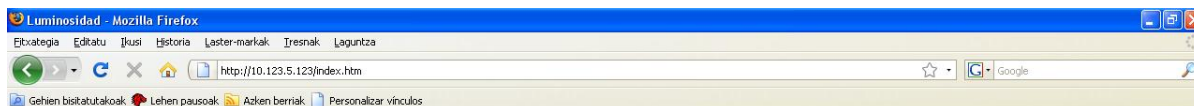


En la siguiente imagen podemos ver el ejemplo de la aplicación cuando en la entrada introducimos una señal analógica de 0V:



En la siguiente imagen podemos ver el ejemplo de la aplicación cuando en la entrada introducimos una señal analógica de 0,49V:

En la imagen podemos ver se realiza la lectura de la entrada analógica que nos da un valor de 0,49V, pero la imagen de la lámpara esta apagada por que para la imagen hemos tomado un comando que toma la entrada como digital.



La luz está:

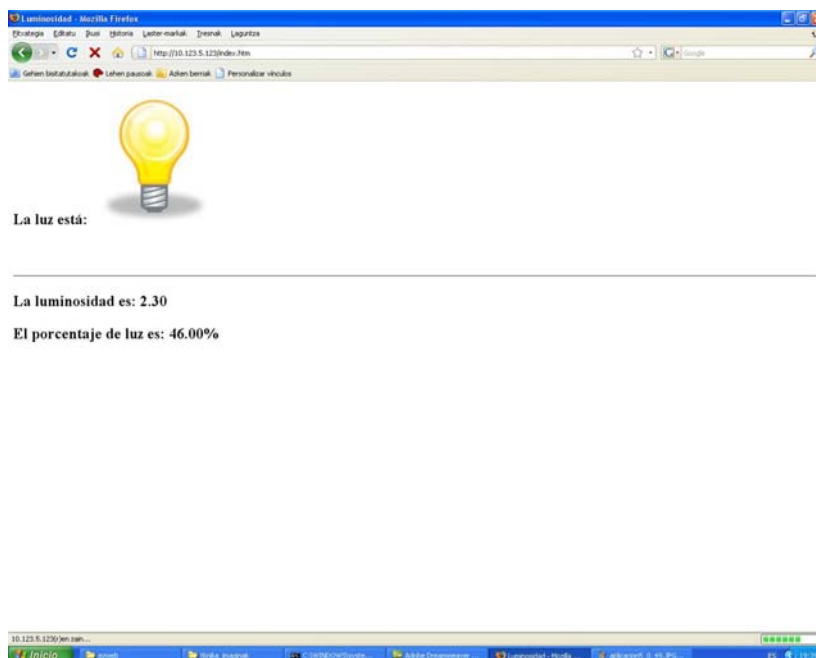
---

La luminosidad es: 0.49

El porcentaje de luz es: 9.80%



En la siguiente imagen podemos ver el ejemplo de la aplicación cuando en la entrada introducimos una señal analógica de 2,30V:



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

