

Proyecto Fiona: Soporte-comunicador por Barrido Visual

Breve descripción

La idea de fabricar este soporte, se originó ante la necesidad de hacer física una herramienta virtual. Necesitábamos conseguir un medio físico para personas iniciadas (principalmente con Parálisis Cerebral) en la comunicación por barrido visual través del ordenador con el software "PLAPHOONS - (C) Jordi Lagares Roset - Proyecto Fressa 2007 - www.lagares.org". En líneas generales este sistema alternativo de comunicación, consta de unos "paneles" conceptuales, distribuidos en casillas (la cantidad de casillas es configurable, por ejemplo 2x2, 3x2...) en las que pondremos las imágenes que queremos. Esas casillas se bordearán de rojo y se mantendrán bordeadas durante un tiempo establecido, cuando pase ese tiempo, pasará la demarcación a la siguiente casilla. Si la niña quiere expresar el contenido de la casilla demarcada, pulsará un conmutador y se activará el mensaje hablado o gráfico. Necesitamos el soporte físico porque hay mucho material no informatizado, y hoy en día no todo el mundo tiene la capacidad de acceder a las nuevas tecnologías. Es un cajetín de madera de 55cmx45cm, en el que colocamos 9 leds (lucecitas). De esos leds, 8 forman una matriz de 4 x 2, que a través de diferentes combinaciones podrán ser modificadas las configuraciones de esa matriz, pudiéndose encender 2, 3, 4, 6, u 8 leds (dependiendo de la necesidad). Estos leds se encenderán de forma intermitente con un tiempo de latencia, en nuestro caso de 60'', emulando el barrido del "PLAPHOONS". Este circuito estará conectado a un conmutador. Cuando se pulse el conmutador se encenderá el led que queda, de modo que la persona sepa que ha provocado un cambio en el medio. También contamos con 5

interruptores, uno de encendido y los otros 4 nos servirán para configurar las matrices (las configuraciones aparecen explicadas mas adelante). Hemos colocado un botón rojo para reiniciar el barrido, ya sea con una nueva configuración o con la misma.

Para quién se hizo

Principalmente se ha creado pensando en Julia, una niña de 11 años que tiene parálisis cerebral. En el aspecto motor está muy limitada, consiguiendo mayor precisión de movimiento con la cabeza. Es ahí donde se le coloca el pulsador con la ayuda de un brazo articulado. Se está entrenando en barrido visual a través del ordenador para llevar a cabo su comunicación.

Propósito

Queremos que siga utilizando esta modalidad (barrido visual) en tareas en las que el software tiene limitaciones, por ejemplo en tareas de tamaños (las fotos en el programa aparecen todas del mismo tamaño). Básicamente es hacer física una herramienta virtual.

Materiales

Los materiales usados se exponen a continuación en dos bloques. Uno para la parte electrónica y el otro para la parte material.

Bloque 1:

- Placa de Circuito Impreso.
- 8 Diodos Led de color rojo.
- 1 Diodo Led de color verde.
- 4 Conmutadores.
- 1 Interruptor.
- 1 Pulsador.
- 1 Jack hembra.
- 1 Potenciómetro de 470K.
- 10 Resistencias de 330 Ohmios.
- 1 Resistencia de 10 K.
- 2 Condensadores de 22pF.
- 2 Condensadores de 100uF de 16V.
- 1 Condensador de 100uF de 25V.

- 1 Condensador de 10uF de 16V.
- 1 Condensador de 100nF.
- 1 Pic 16F84A.
- 1 Temporizador 555.
- 1 Regulador de Tensión LM7805.
- 2 Diodos 1N4148 (solo 1 si no se usa batería).
- 1 Cristal de Cuarzo de 8MHz.
- 1 Conector Hembra para la entrada del transformador.
- Cables.
- Mini-taladro.
- Soldador.
- Estaño.
- Pinzas.
- Destornilladores.

Bloque 2: (ancho x alto x profundidad)

- 2 Maderas de marquetería de 55cm x 45cm x 0.4cm.
- 2 Listones de 55cm x 4cm x 2cm.
- 2 Listones de 41cm x 4cm x 2cm.
- Puntillas sin cabeza.
- 13 Tornillos.
- 1 Asa de plástico (de una caja).

Fco. Javier Montero Rubio

Ana M^a Martínez Cabeza

Avd/ Rafael Alberti nº 20 C.P. 11570 La Barca de la Florida (Cádiz)

montero2210@gmail.com / analogopeda@gmail.com (Ana)

Esquema Eléctrico

El esquema eléctrico aunque parece bastante complicado, no lo es. Esta dividido en 3 partes. Una es el propio circuito que hace encender los 8 leds. Otra parte es el compuesto por el temporizador, que enciende el led verde y una última parte que es la fuente que transforma los 12V de entrada en 5 voltios que son los que alimenta el circuito. En este circuito se ha programado el pic16f84a de forma inversa, es decir hemos alimentado las entradas y cuando queremos hacer un cambio lo que realizamos es su desconexión (lo normal es que este desconectada y cuando queremos un cambio la conectamos).

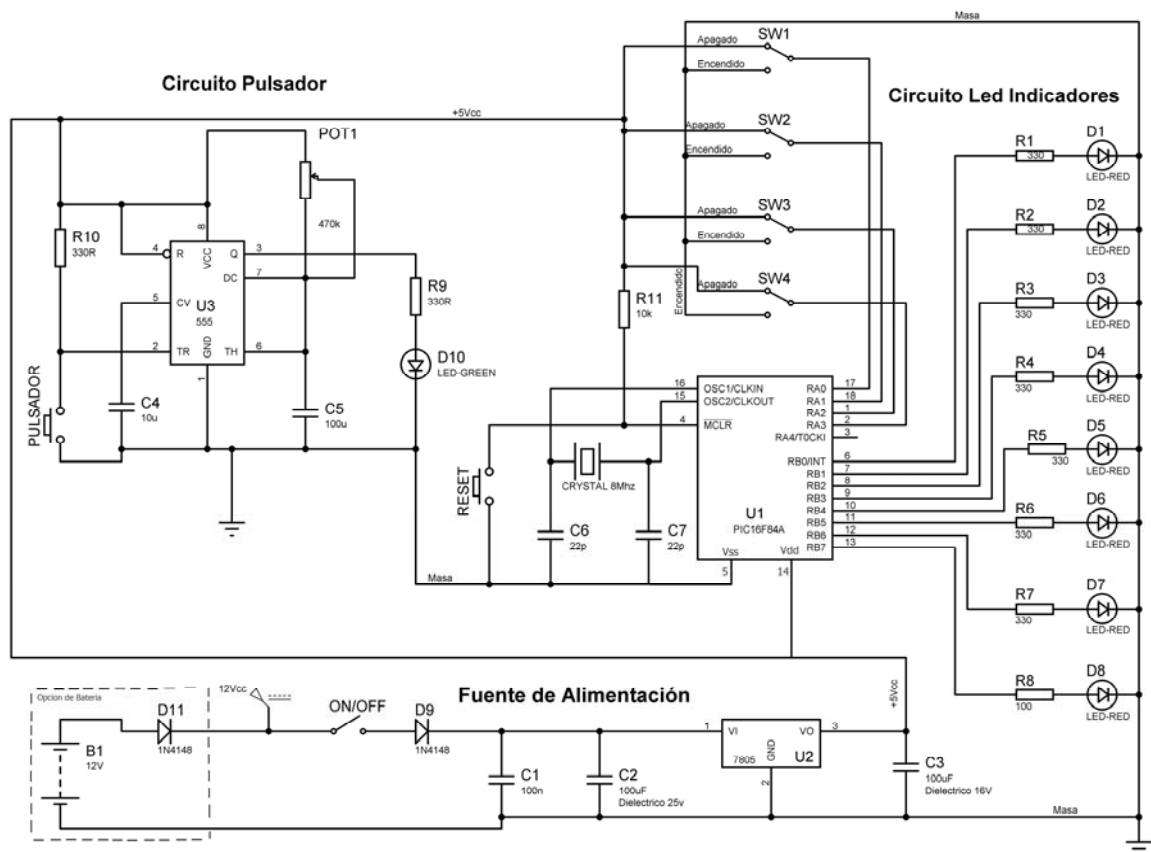


Imagen del esquema eléctrico.

Proceso de Fabricación

- Preparación del soporte

Distribuimos la madera de marquetería para colocar los 9 leds. En cada punto de colocación se marcan las patillas de los leds. Utilizamos un taladro pequeño y una broca de 1 mm. para hacer los agujeros. Vamos a taladrar la madera de forma que quepan los componentes. La madera de marquetería restante se unió a los listones, primero se apuntillaron los mas largos y luego los mas cortos. En el lateral izquierdo se hizo otro taladro para colocar el conector de toma de corriente. Posteriormente se pintaron las maderas de color blanco para mejorar el contraste visual. Una vez seca la pintura, se procede a colocar cada componente en su sitio.

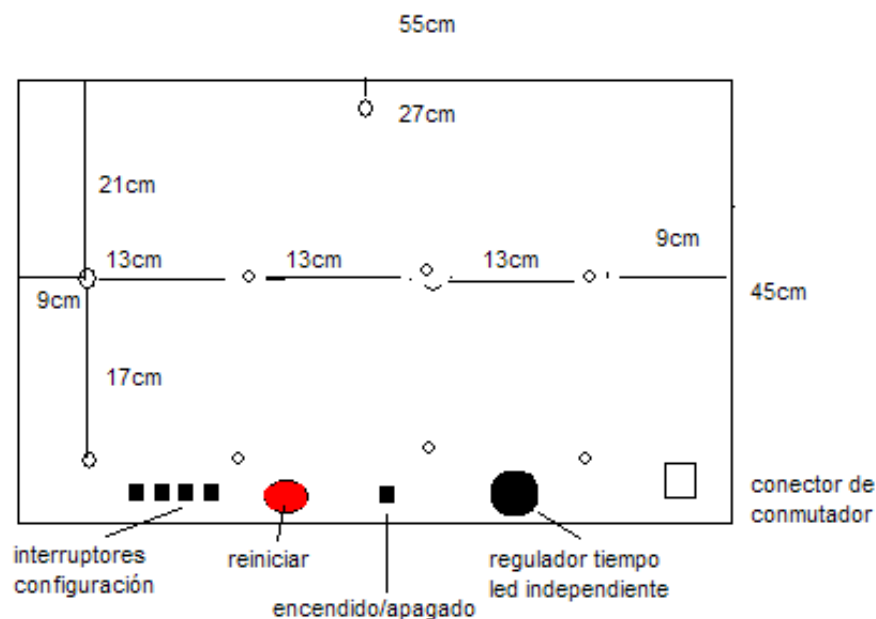


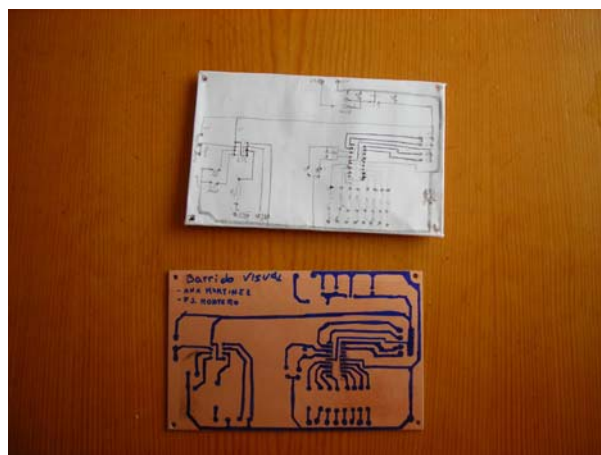
Imagen del esquema de la caja con medidas.

En la madera de marquetería de la parte posterior atornillaremos la placa con sus correspondientes separadores. Podemos colocar el asa de plástico de una caja en la parte superior.

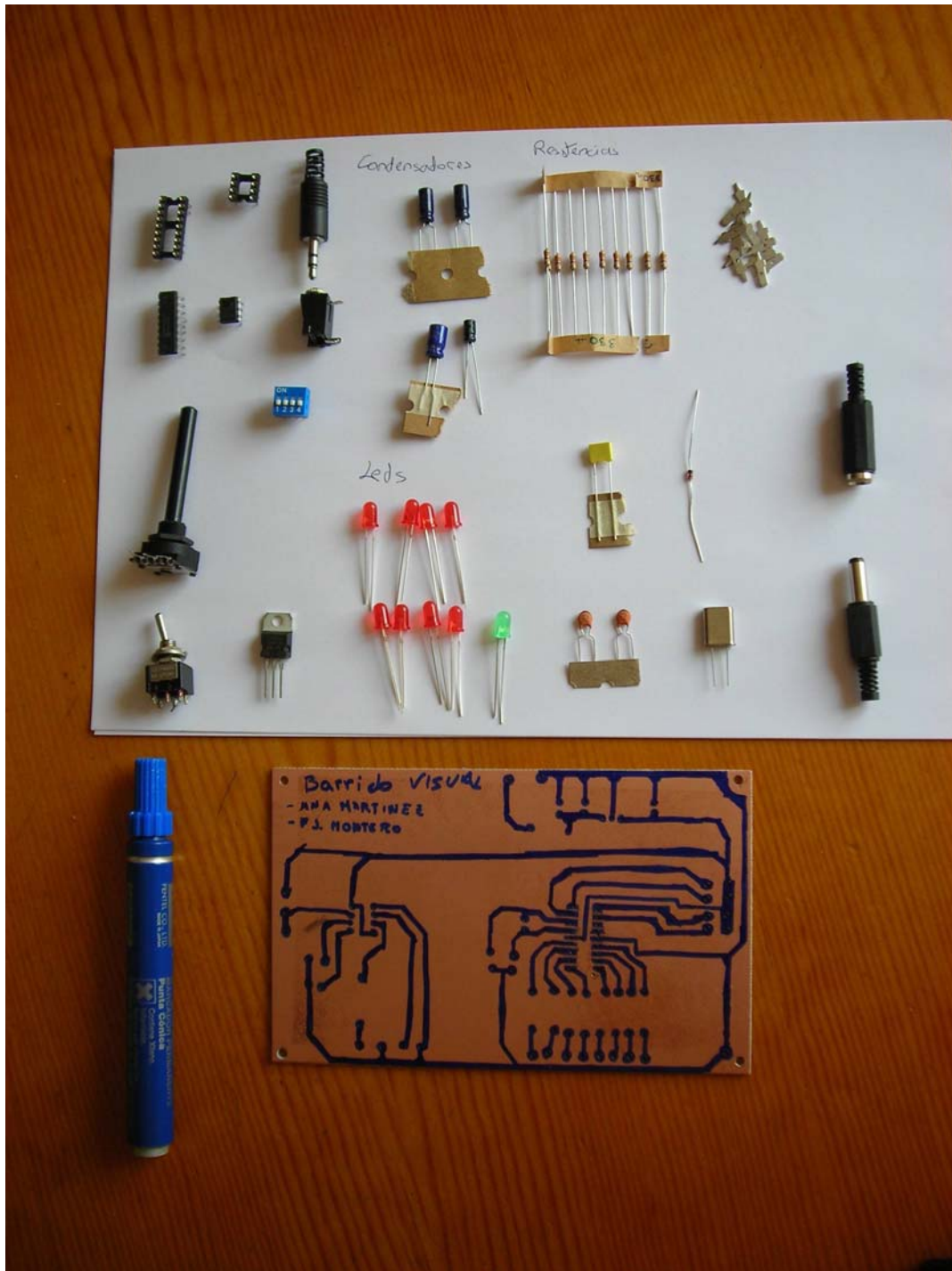


Fotografía de la apariencia externa del comunicador.

- Preparación de la placa



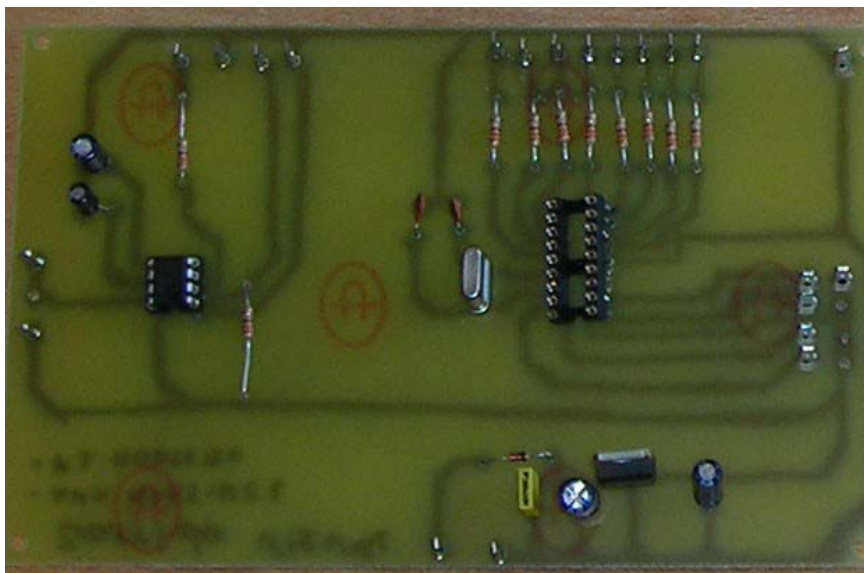
Fotografía de la placa preparada para su quemado.



Fotografía de la placa junto con los componentes que llevará.

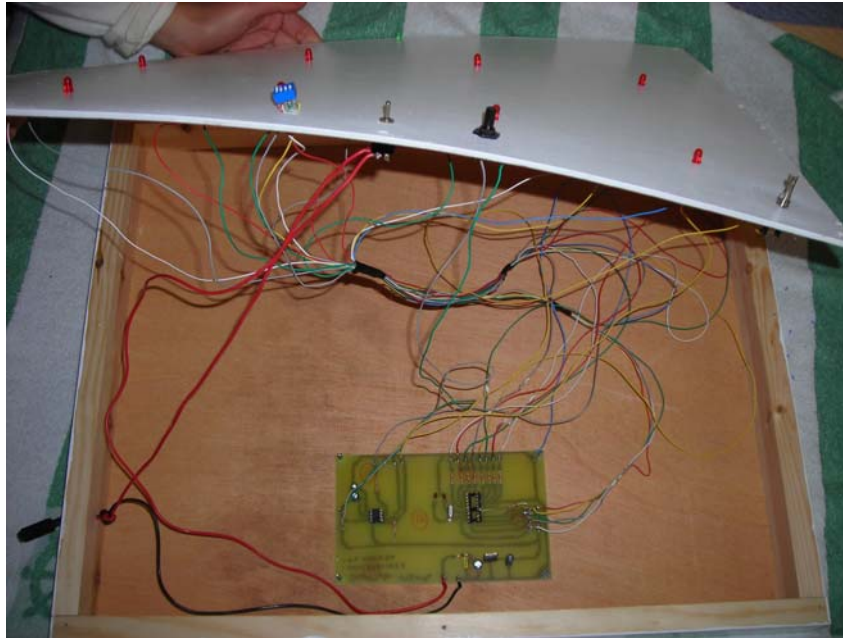
A partir del esquema eléctrico vamos a fabricar la placa.

1. Dibujamos el esquema en un papel superpuesto encima de la placa.
2. Con el papel dibujado encima de la placa, taladramos los agujeritos en los que irán soldados los componentes.
3. Dibujamos el esquema en la placa de forma que ningún cable o línea que no deba estar unida se cruce (e.e. en el esquema aparecen líneas cruzadas que físicamente no lo están). Lo pintaremos con rotulador permanente (*edin 3000*).
4. Verificamos que el esquema de la placa concuerde con el esquema eléctrico.
5. Elaboramos un ácido para posteriormente proceder a su quemado. Este ácido lo hacemos con 25% agua fuerte, 25% agua oxigenada, 50% agua. Una vez hecho, sumergimos la placa en el ácido, hasta comprobar que se ha eliminado todo el cobre de la placa, excepto lo pintado.
5. Soldamos cada componente en su sitio correspondiente.
6. Montamos la placa en la caja.

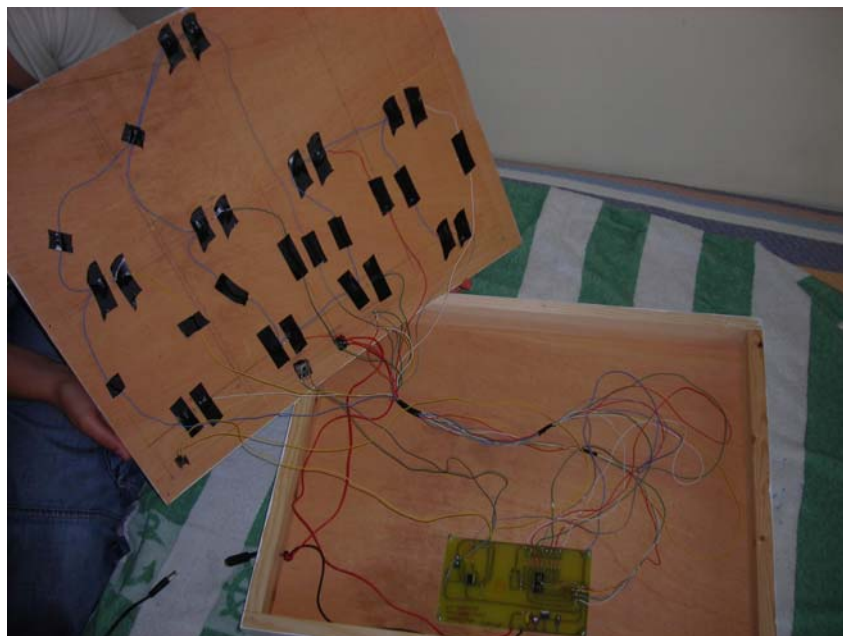


Fotografía de la placa quemada y con los componentes montados.

- Unión de componentes del panel a la placa.



Fotografía del montaje de la placa en el interior del comunicador.



Vista de la conexión de la placa a los leds.

1. Conectaremos entre sí todas las patillas cortas de los leds a través de un cable. Ese cable irá al polo negativo de la placa (Masa), de este modo todas las patillas estarán conectadas a través de un solo cable.
2. La patilla larga de cada led ira conectada con un cable independiente, en cada led, (cada patilla el suyo). Estas se conectarán al lugar correspondiente de la placa (eso se sabe mirando el esquema eléctrico).
3. El resto de los componentes se sueldan en su lugar correspondiente en la placa.

- Programación del PIC16F84A

El código de nuestro programa se ha introducido en un cd que acompaña a esta memoria. Dicho código esta en c y puede ser abierto por cualquier editor de texto como notepad, ultraedit, etc..., no en un procesador de texto.

A continuación ponemos el código.

```

/*****/
/* Nombre: Barrido Visual. */
/* Descripcion: Soporte para sistema de comunicación alternativo/aumentativo */
/* a través de un barrido visual. */
/* */
/* Autores: Francisco Javier Montero Rubio y Ana Maria Martínez Cabeza */
/*****/
/* */
/* Nota: La instrucción delay_ms(), espera el tiempo en microsegundos que este */
/* encendido o apagado el led, es decir es la instrucción de retardo. */
/* */
/* Las asignaciones están en hexadecimal. Esto se consigue pasando el estado */
/* de los puertos a hexadecimal. El bit mas significativo es el puerto 4 de la */
/* entrada a y el puerto 8 de la salida b */
/*****/
//////////////////////////////// DIRECTIVAS DE PREPROCESADO //////////////////////////////////
// libreria para el manejo del pic16f84a
#include <16F84A.h>
// declara la frecuencia del cristal
#use delay(clock=8000000)
//configura los fusibles
// HS es la configuracion del oscilador la cual indica High speed es decir

```

```
// frecuencias altas esto es a partir de 8 Mhz.  
// NOWDT deshabilita el Watch Dog Timer  
// NOPUT deshabilita le reseteo de power up timer  
// NOPROTECT deshabilita la proteccion del codigo del pic.  
#fuses HS,NOWDT,NOPUT,NOPROTECT  
// asignamos a variable port_b el espacio memoria 0x06 que es la dirección de port_b  
// y 0x06 que es la dirección del port_a  
#byte port_b=0x06  
#byte port_a=0x05
```

```
//////////////////// Programa Principal //////////////////////  
void main(){  
// Declaramos el puerto B como salidas  
set_tris_b(0x00);  
// Declaramos el puerto a como entradas  
set_tris_a(0xFF);  
// Bucle infinito para que las instrucciones que tiene  
siempre se ejecuten  
while (true){  
//Leemos el estado de los pulsadores, y según esto  
entramos en un caso o en otro.  
switch (port_a){  
//Modo Demo, este modo no es útil, solo es  
para la comprobación de que todos los leds  
funcionan correctamente. Todos los pulsadores  
en la posición apagada.  
case 0x0F:{  
port_b=0x01; //led 1  
delay_ms(100);  
port_b=0x02; //led 2  
delay_ms(100);  
port_b=0x04; //led 3  
delay_ms(100);  
port_b=0x08; //led 4  
delay_ms(100);  
port_b=0x10; //led 5  
delay_ms(100);  
port_b=0x20; //led 6  
delay_ms(100);  
port_b=0x40; //led 7  
delay_ms(100);  
port_b=0x80; //led 8  
delay_ms(100);  
port_b=0x40; //led 7  
delay_ms(100);  
port_b=0x20; //led 6  
delay_ms(100);  
port_b=0x10; //led 5  
delay_ms(100);  
port_b=0x08; //led 4  
delay_ms(100);  
port_b=0x04; //led 3  
delay_ms(100);  
port_b=0x02; //led 2  
delay_ms(100);  
port_b=0x01; //led 1  
delay_ms(100);  
port_b=0x00; //Todos Apagados
```

```
delay_ms(250);
port_b=0x01; //led 1
delay_ms(250);
port_b=0x00;
port_b=0x02; //led 2
delay_ms(250);
port_b=0x10; //led 5
delay_ms(250);
port_b=0x20; //led 6
delay_ms(250);
port_b=0x04; //led 3
delay_ms(250);
port_b=0x08; //led 4
delay_ms(250);
port_b=0x40; //led 7
delay_ms(250);
port_b=0x80; //led 8
delay_ms(250);
port_b=0x00; //Todos Apagados
delay_ms(250);
port_b=0x01; //led 1
delay_ms(250);
port_b=0x08; //led 4
delay_ms(250);
port_b=0x10; //led 5
delay_ms(250);
port_b=0x80; //led 8
delay_ms(250);
port_b=0x00;
delay_ms(250);
port_b=0xFF;
delay_ms(250);
port_b=0x00;
delay_ms(250);
port_b=0xFF;
delay_ms(1000);
break;
}
//Para el LLed 1 y 2
//Interruptor 1 encendido
case 0x0E:{
port_b=0x01; //Led 1
delay_ms(60000);
port_b=0x02; //Led 2
delay_ms(60000);
break;
}
//Para el led 1, 2 y 3
//Interruptor 2 encendido
case 0x0D:{
port_b=0x01; //Led 1
delay_ms(60000);
port_b=0x02; //Led 2
delay_ms(60000);
port_b=0x04; // Led 3
```

```
delay_ms(60000);
break;
}
//Para el led 1, 2, 3 y 4
//Interruptor 3 encendido
case 0x0B:{
port_b=0x01; //Led 1
delay_ms(60000);
port_b=0x02; //Led 2
delay_ms(60000);
port_b=0x04; //Led 3
delay_ms(60000);
port_b=0x08; //Led 4
delay_ms(60000);
break;
}
port_b=0x00; //Todos Apagados
delay_ms(250);
port_b=0x01; //led 1
delay_ms(250);
port_b=0x08; //led 4
delay_ms(250);
port_b=0x10; //led 5
delay_ms(250);
port_b=0x80; //led 8
delay_ms(250);
port_b=0x00;
delay_ms(250);
port_b=0xFF;
delay_ms(250);
port_b=0x00;
delay_ms(250);
port_b=0xFF;
delay_ms(1000);
break;
}
//Para el Led 1 y 2
//Interruptor 1 encendido
case 0x0E:{
port_b=0x01; //Led 1
delay_ms(60000);
port_b=0x02; //Led 2
delay_ms(60000);
break;
}
//Para el led 1, 2 y 3
//Interruptor 2 encendido
case 0x0D:{
port_b=0x01; //Led 1
delay_ms(60000);
port_b=0x02; //Led 2
delay_ms(60000);
port_b=0x04; // Led 3
delay_ms(60000);
```



```
break;
}
//Para el led 1, 2, 3 y 4
//Interruptor 3 encendido
case 0x0B:{
port_b=0x01; //Led 1
delay_ms(60000);
port_b=0x02; //Led 2
delay_ms(60000);
port_b=0x04; //Led 3
delay_ms(60000);
port_b=0x08; //Led 4
delay_ms(60000);
break;
}

//Para el led 1, 2, 5 y 6
//Interruptor 4 encendido
case 0x07:{
port_b=0x01; //led 1
delay_ms(60000);
port_b=0x02; //led 2
delay_ms(60000);
port_b=0x10; //led 5
delay_ms(60000);
port_b=0x20; //led 6
delay_ms(60000);
break;
}
//Para el led 1,2,3,5,6 y 7
//Interruptor 1 y 2 encendido
case 0x0C:{
port_b=0x01; //Led 1
delay_ms(60000);
port_b=0x02; //Led 2
delay_ms(60000);
port_b=0x04; //Led 3
delay_ms(60000);
port_b=0x10; //led 5
delay_ms(60000);
port_b=0x20; //led 6
delay_ms(60000);
port_b=0x40; //led 7
delay_ms(60000);
break;
}
//Para el led 1,2,3,4,5,6,7 y 8
//Interruptor 1,2,3 y 4 encendidos
case 0x00:{
port_b=0x01; //led 1
delay_ms(60000);
port_b=0x02; //led 2
delay_ms(60000);
port_b=0x04; //led 3
```

```
delay_ms(60000);
port_b=0x08; //led 4
delay_ms(60000);
port_b=0x10; //led 5
delay_ms(60000);
port_b=0x20; //led 6
delay_ms(60000);
port_b=0x40; //led 7
delay_ms(60000);
port_b=0x80; //led 8
delay_ms(60000);
break;
}

//Para el 1 y 5
//interruptor 3 y 4 encendido
case 0x03:{
port_b=0x01; //led 1
delay_ms(60000);
port_b=0x10; //led 5
delay_ms(60000);
break;
}
//Para el 1,2,3,4,5,6,7 y 8 pero con 30
segundos
//Interruptor 1,2,3 encendidos
case 0x08:{
port_b=0x01; //led 1
delay_ms(30000);
port_b=0x02; //led 2
delay_ms(30000);
port_b=0x04; //led 3
delay_ms(30000);
port_b=0x08; //led 4
delay_ms(30000);
port_b=0x10; //led 5
delay_ms(30000);
port_b=0x20; //led 6
delay_ms(30000);
port_b=0x40; //led 7
delay_ms(30000);
port_b=0x80; //led 8
delay_ms(30000);
break;
}
//Para el led 1,4
//Interruptor 2,3 y 4 encendidos
case 0x01:{
port_b=0x01; //led 1
delay_ms(60000);
port_b=0x08; //led 4
delay_ms(60000);
break;
}
}
```

```
//Para el led 1,4,5,8
//Interruptor 2 y 4 encendidos
case 0x05:{
port_b=0x01; //led 1
delay_ms(60000);
port_b=0x08; //led 4
delay_ms(60000);
port_b=0x10; //led 5
delay_ms(60000);
port_b=0x80; //led 8
delay_ms(60000);
break;
}

//Para el led 1,3,5 y 7
//Interruptor 2 y 3
case 0x06:{
port_b=0x01; //led 1
delay_ms(60000);
port_b=0x04; //led 3
delay_ms(60000);
port_b=0x10; //led 5
delay_ms(60000);
port_b=0x40; //led 7
delay_ms(60000);
break;
}

//Para el led 1y2 , 3y4, 5y6, 7y8
encendidos de dos en dos
//Interruptor 1 3 y 4
case 0x02:{
port_b=0x03; //led 1y2
delay_ms(60000);
port_b=0x0C; //led 3y4
delay_ms(60000);
port_b=0x30; //led 5y6
delay_ms(60000);
port_b=0xC0; //led 7y8
delay_ms(60000);
break;
}

//En esta parte se hace un parpadeo de todos los
led. Esto indica que no se ha configurado ninguna
de las opciones anteriores.
default:{
port_b=0xff;
delay_ms(500);
port_b=0x00;
delay_ms(500);
port_b=0xff;
delay_ms(500);
port_b=0x00;
delay_ms(500);
port_b=0xff;
}
```

```

delay_ms(500);
port_b=0x00;
delay_ms(500);
port_b=0xff;
delay_ms(500);
port_b=0x00;
delay_ms(500);
break;
}
}
} // Fin de bucle while
} // Fin del programa principal main

```

Interruptores

En la siguiente tabla se muestran las posibles combinaciones de los diferentes interruptores que actualmente se encuentran programadas. Existen mas configuraciones que no están programadas y por lo tanto se podrán utilizar en un futuro si fuese necesario. Si se establece una combinación no programada, el hardware nos avisara haciendo que todos los leds parpadeen 4 veces. Si ocurriese esto, verificaríamos la posición de los leds y pondríamos la correcta, resetearíamos (en el botón rojo) y comenzaría a funcionar correctamente. Existe un modo que se le ha denominado Modo Demo. En este modo se realizara una serie de combinaciones de led. Estas combinaciones se utilizan para ver si todos los leds funcionan correctamente. Además de este modo, si ponemos una configuración que no este programada, se encenderán todos los leds. De esta forma también podríamos verificar que todos los leds estén funcionando correctamente. A continuación se expone la tabla con las diferentes combinaciones actualmente programadas.

Pulsadores	Leds	Nota
Todos Apagados	Modo Demo	Verificación de todos los leds
Interruptor 1 Encendido	Leds 1 y 2	Tiempo de encendido cada led 60 segundos
Interruptor 2	Encendido Leds 1, 2 y 3	Tiempo de encendido cada led 60 segundos

Interruptor 3	Encendido Leds 1, 2, 3 y 4	Tiempo de encendido cada led 60 segundos
Interruptor 4	Encendido Leds 1, 2, 5 y 6	Tiempo de encendido cada led 60 segundos
Interruptor 1 y 2 Encendido	Leds 1, 2, 3, 5, 6 y 7	Tiempo de encendido cada led 60 segundos
Interruptor 1, 2, 3 y 4 Encendido	Leds 1, 2, 3, 4, 5, 6, 7 y 8	Tiempo de encendido cada led 60 segundos
Interruptor 3 y 4 Encendido	Leds 1 y 6	Tiempo de encendido cada led 60 segundos
Interruptor 1, 2, y 3 Encendido	Leds 1, 2, 3, 4, 5, 6, 7 y 8	En este modo, el tiempo de encendido es de 30segundos
Interruptor 2, 3 y 4 Encendido	Leds 1 y 4	Tiempo de encendido cada led 60 segundos
Interruptor 2 y 4 Encendido	Leds 1, 4, 5 y 8	Tiempo de encendido cada led 60 segundos
Interruptor 1 y 4 Encendido	Leds 1, 3, 5 y 7	Tiempo de encendido cada led 60 segundos
Interruptor 1, 3, y 4 Encendido	Leds 1y2, 3y4, 5y6, 7y8	En este modo se enciende de dos en dos y dura 60 segundos
Interruptor 1 y 3	No configurado	
Interruptor 2 y 3	No configurado	
Interruptor 1, 2 y 4	No configurado	
Si no se utiliza ninguna de estas configuraciones, parpadearan todos los leds a la vez. Esto indica que los interruptores no están en la posición correcta.		

Posibles Modificaciones

En nuestro caso, el hardware ha sido diseñado para la utilización de una fuente de alimentación (transformador) externa que convierta los 220 V de corriente alterna (un enchufe normal y corriente) a 12 V en continua. En el esquema que acompaña a esta memoria se le ha introducido la opción de hacerlo funcionar con una pila o batería de 12 voltios.

Otra modificación que se le puede realizar es un circuito oscilador que varíe la

frecuencia del cristal de cuarzo. Al variar esta frecuencia variamos el tiempo que están encendido o apagados los leds. Si hiciésemos esto en la programación tendríamos que hacer cambios, tales como indicar que tipo de oscilador se le va a poner.

Además de esa modificación se le pueden realizar muchas, tales como forma de encendido de los leds, etc....

ANEXO

Software

El software necesario para la elaboración es el siguiente.

Proteus: Este programa ha sido utilizado para el diseño y simulación del circuito electrónico que hemos utilizado. Este software se puede encontrar en la siguiente dirección. Nosotros hemos utilizado la versión 7.2.

<http://www.ieeprotes.com/descarga.html>

CCS C: Dicho programa es el utilizado para la programación del pic. Esta programación se ha realizado en lenguaje C. Este mismo programa genera los archivos necesarios para introducirlos directamente en el pic. Este programa se puede descargar de Internet de forma gratuita en la siguiente dirección

<http://www.ccsinfo.com/>

WINPIC800: Este software ha sido utilizado para introducir el código generado por CCS C en el pic. Para poder programar nuestro pic necesitaremos un hardware específico. Este hardware es el T-20 un programador de bajo coste. Una vez conectado al puerto serie de nuestro equipo, deberemos configurar el software winpic800, para ello pulsaremos en configuración y a continuación en Hardware. Seleccionamos TE20 y pulsamos en guardar esta configuración. Una vez realizado estos pasos, seleccionamos el pic que queremos grabar (en nuestro caso es el PIC 16F84A) y a continuación pulsamos en abrir y seleccionamos el fichero .hex que será introducido en el pic. Una vez realizado esto procederemos a dar a programar todo y esperamos.

Cuando este proceso termine, ya tenemos nuestro pic programado y podremos introducirlo en nuestro hardware. Este software se puede conseguir en la siguiente dirección:

<http://www.winpic800.com//descargas/WinPic800.zip>

Contenido del CD

Junto a esta memoria se incorpora un cd con la memoria en formato digital, así como todos los archivos de programación y esquemas eléctricos de la placa de circuito impreso de nuestro hardware.

Los archivos que se incorporan son los siguientes.

- Memoria
- Archivo de programación en C
- Archivo de programación en .hex
- Esquema eléctrico en formato jpg.
- Esquema eléctrico para Proteus.
- Fotos y Video