

Encender una bombilla

Para llevar a cabo este ejemplo lo dividiremos en las siguientes partes:

- Conexiones
- Diagrama de flujo
- Programación en MSWLogo
- Programación en C

Conexiones

Necesitará una bombilla pequeña de 0,3 Amperios aproximadamente. Es conveniente usar un casquillo donde poner la bombilla y conectar un cable al polo positivo y otro al negativo del casquillo.

Las conexiones que se establecerán entre la bombilla y la tarjeta controladora CNICE se muestran en la siguiente tabla:

CONECTOR	SALIDA DIGITAL
0	Cable del polo positivo
1	Cable del polo negativo

Es indiferente si la conexión se hace al revés, es decir, se conecta el cable conectado al polo negativo del casquillo a la salida digital 0 y el otro cable a la salida 1. Esto es así porque la bombilla no tiene polaridad.

Se puede usar cualquiera de las otras salidas digitales para realizar la conexión entre la bombilla y la controladora.

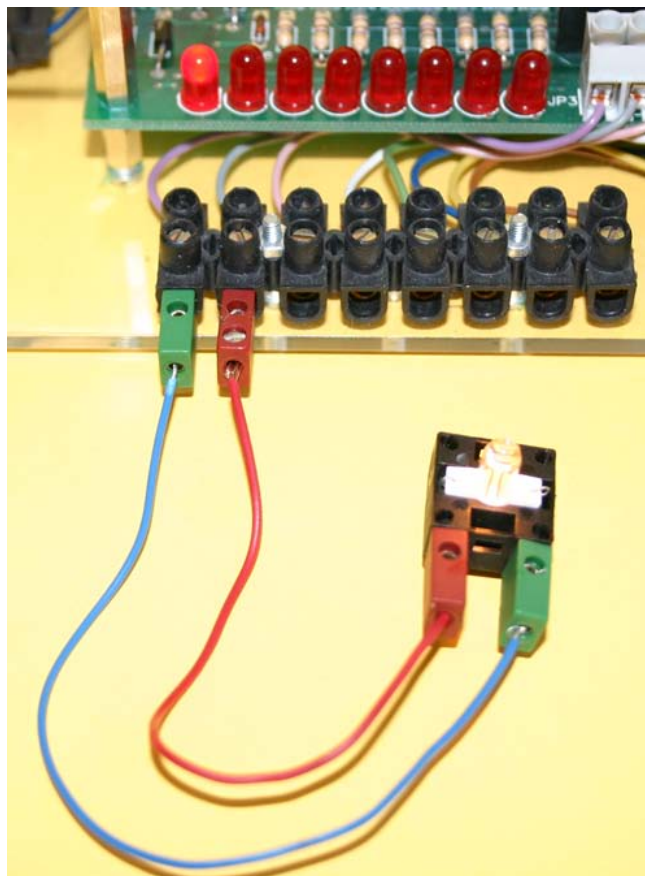
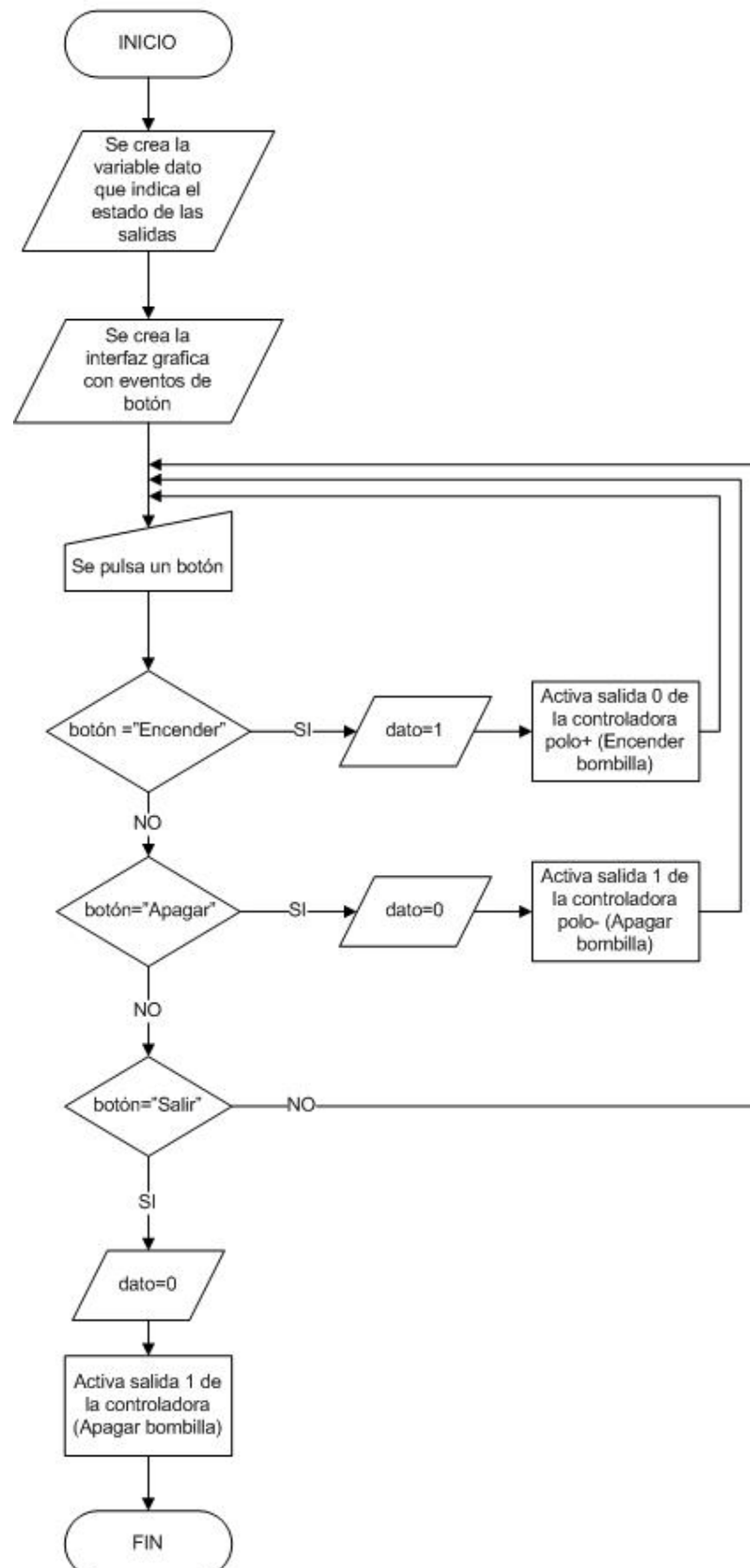


Diagrama de Flujo

BOMBILLA



Programación en MSWLogo

La programación de este caso se estructura en los siguientes pasos:

1.- Se implementa un procedimiento para crear la ventana gráfica del programa. Para ello se utiliza la función **creaventana**. Para crear un marco dentro de la ventana se utiliza la función **creagroupbox**. Por último dentro del marco se crean los botones con la función **creaboton**. Dentro de cada botón se establecerán entre los corchetes las funciones que se han de ejecutar una vez presionado el botón. Además de esto, se mostrará la imagen de la bombilla con **cargadib**.

Ejemplo:

```
para VentanaSeleccion
```

```
creaventana "trabajo "EncBombilla [BOMBILLA] 40 30 125 75 []
```

```
creagroupbox "EncBombilla "E_Ventana 5 0 113 60
```

```
creaboton "EncBombilla "Encender "Encender 25 20 35 10 [EncenderBombilla]
```

```
creaboton "EncBombilla "Apagar "Apagar 70 20 35 10 [ApagarBombilla]
```

```
creaboton "EncBombilla "Salir "Salir 45 40 35 10 [ApagarBombilla borraVentana  
"EncBombilla]
```

```
cargadib "Sinluz.bmp
```

```
fin
```

2.- Se crean varios procedimientos para encender y apagar la bombilla. Para encender la bombilla se utilizará la función **conecta** y a continuación el número **1** que indica que se activen el par de salidas digitales 0 y 1, y para apagar la bombilla se utiliza la función **desconecta**, similar es su sintaxis a la función anterior.

Ejemplo:

```
para EncenderBombilla
```

```
cargadib "conluz.bmp
```

```
CONECTA 1
```

```
fin
```

```
para ApagarBombilla
```

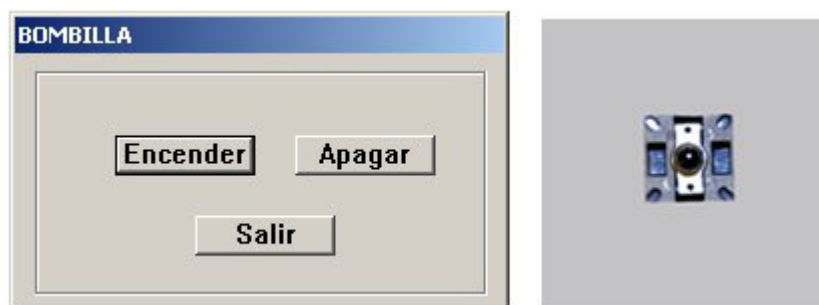
```
cargadib "Sinluz.bmp
```

```
DESCONECTA 1
```

```
fin
```

3.- Se llama al procedimiento que crea la ventana gráfica fuera de cualquier procedimiento, para que se cargue la aplicación gráfica nada más cargar el fichero de logo.

Descargue el fichero programado en **MSWLogo**, descomprímalo y guárdelo en un directorio aparte. Contiene el fichero de código en MSWLogo (bombilla.lgo) y las imágenes de la bombilla. Ejecute el compilador MSWLogo versión 6.5a en castellano. Vaya al menú del programa, Archivo/Abrir y seleccione el fichero bombilla.lgo que se descargó previamente. Se visualizará la siguiente pantalla:



Pulsando los botones que se le presentan podrá encender y apagar la bombilla o salir de la aplicación.

Programación en C

La programación de este caso se estructura en los siguientes pasos:

- 1.- Se crea un nuevo proyecto
- 2.- Se añaden al proyecto los archivos io.h, io.cpp, Primitivas_CNICE.CPP y Primitivas_CNICE.HPP y SDL.h (librería que permite añadir imágenes en la aplicación programada con C).
- 3.- Se crea el archivo main.c donde se incluirán las funciones necesarias para crear las ventanas
- 4.- Dentro del archivo main.c creado anteriormente, se añade la declaración a las funciones de la biblioteca io.dll de la siguiente manera:

```
#include "io.h"
```

También se añade la declaración a las funciones de la biblioteca SDL.dll de la siguiente manera:

```
#include <SDL.h>
```

- 5.- En nuestro archivo main.c se define la siguiente función que permite activar o desactivar las entradas digitales de la controladora.

```
void Escribir_Salidas_Digitales(int Dato)
{
    LoadIODLL();
    PortOut(0x37A,0x7);
    PortOut(0x378,Dato);
}
```

- 6.- Se crearán dos botones en nuestra ventana, uno para encender la bombilla y otro para apagarla, y en cada uno de ellos se incluye la llamada a la función anterior para cambiar el estado de las salidas de la siguiente manera:

```
switch(LOWORD(wParam))
{
    case 1: // botón Encender
        conluz = SDL_LoadBMP("conluz.bmp");
        screen = SDL_SetVideoMode( 146, 146, 0, SDL_NOFRAME);
        if( screen == NULL ) {
            printf( "Error al entrar a modo grafico: %s\n", SDL_GetError() );
            SDL_Quit();
            return -1;
        }
        SDL_BlitSurface(conluz, NULL,screen,NULL);
        SDL_Flip(screen);
        Escribir_Salidas_Digitales(1);
        break;
    case 2: // botón Apagar
```

Interfaz de control de dispositivos externos por ordenador a través de puerto paralelo

```
sinluz = SDL_LoadBMP("sinluz.bmp");
screen = SDL_SetVideoMode( 146, 146, 0, SDL_NOFRAME );
if( screen == NULL ) {
    printf( "Error al entrar a modo grafico: %s\n", SDL_GetError() );
    SDL_Quit();
    return -1;
}
SDL_BlitSurface(sinluz, NULL,screen, NULL);
SDL_Flip(screen);
Escribir_Salidas_Digitales(0);
break;
case 3: // botón Salir
    Escribir_Salidas_Digitales(0);
    SendMessage(hwnd, WM_CLOSE, 0, 0);
    break;

default:
    break;
}
```

Explicación de las funciones de la librería SDL.

Cada vez que se quiera mostrar una imagen se indicara en nuestro proyecto lo siguiente:

```
foto = SDL_LoadBMP("imagen.bmp");
screen = SDL_SetVideoMode(200, 308, 0, SDL_NOFRAME );
if( screen == NULL ) {
    printf( "Error al entrar a modo grafico: %s\n", SDL_GetError() );
    SDL_Quit();
    return -1;
}
rect.x=0;
rect.y=0;
rect.w=primera1->w;
rect.h=primera1->h;
destino.x=0;
destino.y=0;
SDL_BlitSurface(primera1, &rect, screen, &destino);
SDL_Flip(screen);
```

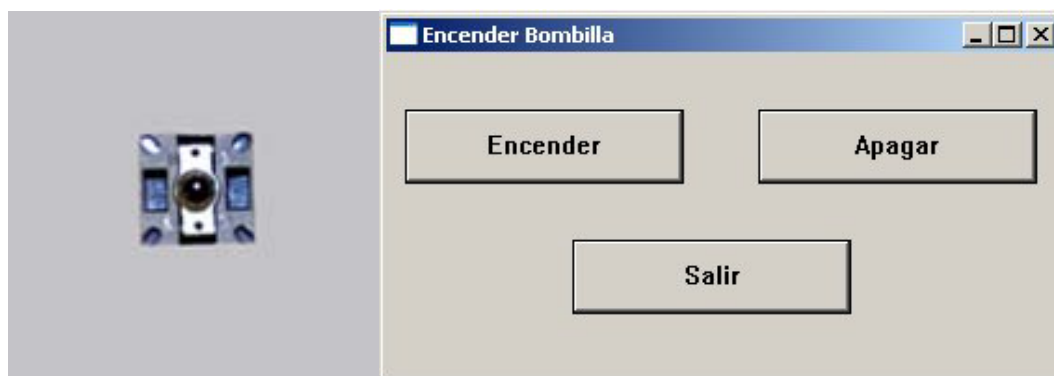
donde **foto** y **screen** son del tipo `SDL_Surface` y **rect** y **destino** es del tipo `SDL_Rect`.

- `SDL_LoadBMP`: carga la imagen .bmp que queramos
- `SDL_SetVideoMode` (int width, int height, int bpp, Uint32 flags): configure un modo de video con una anchura (width), una altura (height) y unos bits-por-pixeles. El parámetro flags indica el tipo de ventana que se quiere. En nuestro caso una ventana sin titulo no borde.
- `SDL_BlitSurface`(imagen, &rect, screen, &destino): pega desde la imagen, la porción seleccionada por rect sobre la superficie screen en el destino indicado por destino.
- `SDL_Flip`(screen): muestra la imagen que se ha seleccionado.

7.- Una vez creados los botones con la función que les corresponden, se compila comprobando que no hay ningún error.

8.- Una vez que se ha comprobado que no hay ningún error en nuestro código, se ejecuta y se comprueba en funcionamiento de la bombilla. Al ejecutar el proyecto se creará el fichero **bombilla.exe**

Descargue los diferentes archivos que forman todo el proyecto programado en C, descompríalos y guárdelos en un directorio aparte. Ejecute el fichero **bombilla.exe**. Se visualizará la siguiente pantalla:



Pulsando los botones que se le presentan podrá encender y apagar la bombilla o salir de la aplicación.

Nota:

En la aplicación programada con C, la imagen puede no aparecer al lado de la ventana. En este caso basta con mover nuestra ventana y se verá correctamente la imagen.