

CEROS Y UNOS.

Los ordenadores son dispositivos que internamente trabajan usando el denominado sistema binario de numeración. En él sólo se dispone de dos dígitos, el "cero" y el "uno".

0 "cero" significa ausencia de corriente eléctrica, desactivado, desconectado,...
1 "uno" significa presencia de corriente eléctrica, activado, conectado,...

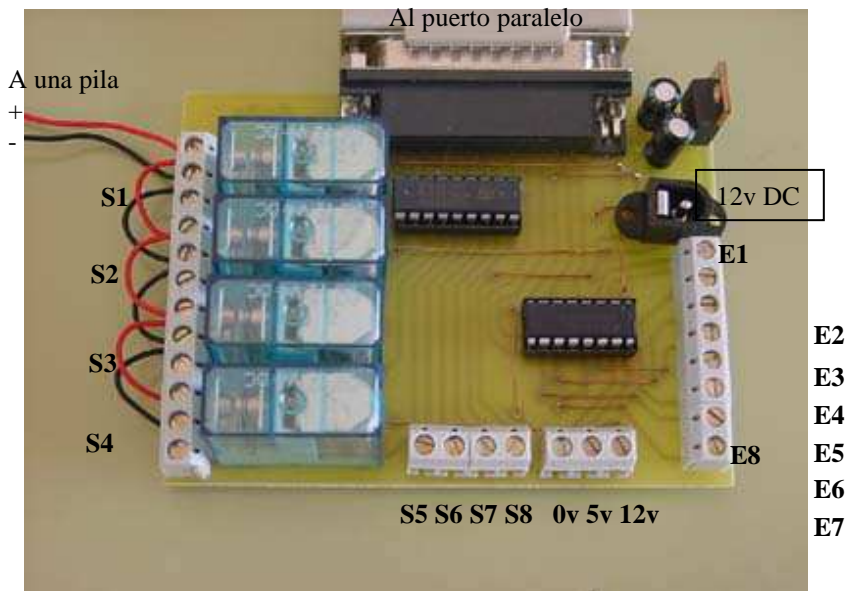
Ellos entienden estos dos dígitos y pueden suministrarlos para activar y desactivar elementos externos (motores, bombillas, timbres, electroimanes,...). Del mismo modo, también aceptarían órdenes externas dadas en dicho formato, mediante interruptores o pulsadores que permitan o no el paso de la corriente eléctrica.

PLACA CONTROLADORA.

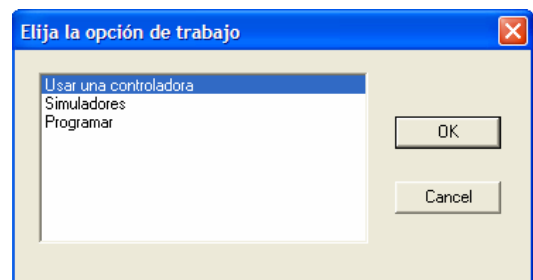
Para conectar el ordenador con el mundo exterior es necesario una placa controladora que proporcione las salidas y las entradas oportunas. En nuestro caso disponemos de una placa controladora que

dispone de 8 salidas y 8 entradas, en ambos casos digitales; es decir, trabajan exclusivamente con "ceros" (tensión eléctrica próxima a 0v) y "unos" (tensión eléctrica de 5v). La unión con el ordenador se realiza a través del puerto paralelo. La placa se alimenta mediante una fuente externa de 12v.

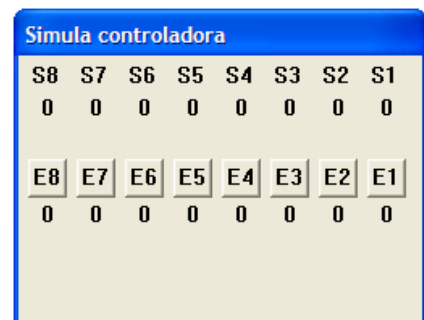
Las salidas S1, S2, S3 y S4 están formadas por sendos relés, tal como se aprecia en la figura anterior. S5, S6, S7 y S8 se corresponden con salidas a transistor en colector abierto.



Para utilizarla seguiremos necesitando MSWLogo. Con él realizaremos los programas que hagan funcionar los dispositivos que conectemos a las salidas y entradas de la placa. Para ello, cuando se inicia MSWLogo y aparece la ventana "Elija una opción", tenemos que seleccionar "Usar una controladora". De esta manera se cargarán unos procedimientos especiales (salidas y leerentradas) que nos permitirán comunicarnos con la placa.



Normalmente, antes de probar un programa directamente en la controladora, lo simularemos utilizando sim8e8s (se elige la opción simuladores y se selecciona sim8e8s de la lista que aparece). Nos aparecerá en pantalla una ventana que responderá al programa que realicemos poniendo a uno o a cero las salidas oportunas y simulando las entradas mediante botones que podemos pinchar con el ratón.



INSTRUCCIONES DE CONTROL.

salidas :n

Activa (pone a 1) las salidas correspondientes al número indicado, según la tabla adjunta. Se pueden conectar varias salidas a la vez sumando los números correspondientes a cada una de ellas.

S8	S7	S6	S5	S4	S3	S2	S1
128	64	32	16	8	4	2	1

Ejemplo:

```
salidas 0   Desconecta todas las salidas.
salidas 8   Activa la salida 4, desconecta todas las demás.
salidas 3   Activa las salidas 1 y 2, desconecta todas las demás.
salidas 38  Activa las salidas 6, 3 y 2, desconecta todas las demás.
```

Al ejecutar esta instrucción se actualizan a 0 o a 1 unas variables correspondientes a cada una de las salidas ("s1, "s2,..., "s8) y, por tanto, si es necesario, podremos preguntar por su estado desde cualquier lugar de nuestros programas.

Ejemplo: si :s1=1 [salidas 0] ;si s1 está en 1 desactiva todas las salidas
 si :s1=0 [salidas 1] ;si s1 está en 0 activa la salida 1 y desactiva las demás

También disponemos de la variable "sn que guarda, en todo momento, el número decimal correspondiente al estado de las ocho salidas.

Ejemplo: si :s3=0 [salidas :sn+4] ;activa la salida 3 sin modificar el resto

 si :s3=1 [salidas :sn-4] ;desactiva la salida 3 sin modificar el resto

leerentradas

Realiza un chequeo del valor (0 o 1) que hay en cada una de las entradas, almacenándolo en las variables: "e1, "e2, "e3, ..., "e8. Para saber, posteriormente, en que estado se encuentra una determinada entrada tenemos que preguntar por el valor de la variable correspondiente.

Ejemplo: leerentradas
 si :e1 = 1 [salidas 1] ;si la entrada 1 vale 1 (5v) activa la salida 1

Al igual que con las salidas, también se crea la variable "en para almacenar el número decimal correspondiente al estado de todas las entradas.

EL CONTROL DEL TIEMPO. ESPERA.

En los programas de control es muy frecuente que tengamos la necesidad de realizar temporizaciones, es decir, controlar el tiempo que un dispositivo está encendido o apagado. MSWLogo cuenta con una instrucción específica para este trabajo, se trata de espera:

espera tiempo

Detiene la ejecución de un procedimiento durante el tiempo especificado (éste se expresa en 1/60 segundos), una vez transcurrido continúa con la ejecución del resto de instrucciones.

Ejemplos: espera 1 Detiene el programa durante 1/60 segundos (0'0166 segundos).
 espera 60 Detiene el programa durante 1 segundo.
 espera 300 Detiene el programa durante 5 segundos.
 espera 3600 Detiene el programa durante 1 minuto.

ACTIVIDADES: (utilizar sim8e8s para comprobar las actividades)

1. Indicar que hace el siguiente programa:

```
para actividad1
salidas 3 espera 600
salidas 2 espera 300
salidas 0
fin
```

2. Indicar que hace el siguiente programa:

```
para actividad3
ponfoco [Pantalla de MSWLogo]
ponteclado [control]
fin
```

```
para control
haz "tecla car lc
si :tecla = "a [salidas 1 espera 480 salidas 0]
fin
```

3. Control marcha-paro de un motor. Se dispone de un motor conectado a la salida 1 y se quiere controlar con la teclas m (marcha) y p (parada). Programa el control de dicho motor.

4. Un motor, conectado a las salidas S1 y S2, debe girar a derecha durante 5 segundos, posteriormente parar durante 2 seg. Y finalmente girar a izquierda durante 10 seg. antes de detenerse. Realizar el programa adecuado, teniendo en cuenta que el inicio del ciclo tiene lugar accionando la tecla i.

5. Tenemos 8 bombillas conectadas a las 8 salidas de la placa controladora y queremos que se enciendan progresivamente, de una en una, a intervalos de 0'5 s. Una vez que estén todas encendidas permanecerán así 5 s. Transcurrido dicho tiempo se apagarán todas y volverá a iniciarse el ciclo.

6. Se dispone de un motor, conectado a las salidas S1 y S2, y queremos controlarlo con la teclas d (giro a la derecha), i (giro a la izquierda) y p (parada). Realizar el programa para controlar dicho motor.

7. Programar el funcionamiento de un semáforo para un paso de peatones. Las conexiones a la placa controladora serán las siguientes:

Peatones		Coches			Tiempo
Rojo	Verde	Rojo	Ambar	Verde	
S5	S4	S3	S2	S1	
1	0	0	0	1	30 seg
1	0	0	1	0	5 seg
0	1	1	0	0	15 seg

8. Realizar un programa para controlar, con la teclas correspondientes del teclado numérico, el movimiento (derecha, izquierda, adelante y atrás) del vehículo que utilizamos en el buscador de luz. Dispone de dos motores conectados a las salidas S1 y S2 el motor derecho y S3, S4 el motor izquierdo.

9. Indicar que hace el siguiente programa:

```
para actividad2
leerentradas
si :el = 1 [salidas 1 espera 480 salidas 0]
actividad2
fin
```