

***ENTORNOS DE TRABAJO EN  
MSWLOGO PARA LAS CONTROLADORAS  
ENCONOR Y CNICE***

# ÍNDICE

<b>1</b>	<b>INTRODUCCIÓN</b>	<b>3</b>
<b>2</b>	<b>ENTORNO DE TRABAJO PARA LA CONTROLADORA ENCONOR</b>	<b>5</b>
2.1	Procedimientos para establecer el canal de comunicación:	10
2.2	Primitivas de control:	11
2.2.1	Conjunto de primitivas de control diseñadas para trabajar con los actuadores conectados a las salidas digitales:	11
2.2.2	Conjunto de primitivas de control diseñadas para trabajar con las entradas digitales:	14
2.2.3	Conjunto de primitivas de control diseñadas para trabajar con las entradas analógicas:	15
2.2.4	Conjunto de primitivas de control diseñadas para trabajar con las salidas analógicas:	17
2.3	Procedimientos para crear el entorno gráfico:	18
<b>3</b>	<b>ENTORNO DE TRABAJO PARA LA CONTROLADORA CNICE</b>	<b>19</b>
3.1	Primitivas de control	22
3.1.1	Conjunto de primitivas de control diseñadas para trabajar con los actuadores conectados a las salidas digitales:	22
3.1.2	Conjunto de primitivas de control diseñadas para trabajar con las entradas digitales:	26
3.1.3	Conjunto de primitivas de control diseñadas para trabajar con las entradas analógicas:	27
3.2	Procedimientos para crear el entorno gráfico:	29

# 1 INTRODUCCIÓN

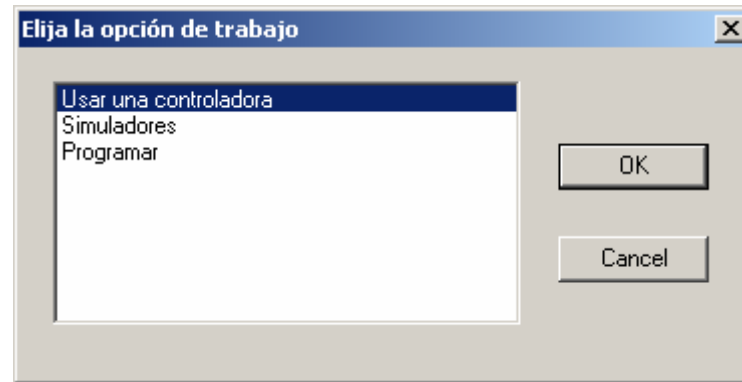
Esta versión de MSWLogo permite la carga, al abrirse, de un entorno para manejar las controladoras ENCONOR y CNICE. Para conseguirlo se han hecho unas pequeñas modificaciones en los proyectos de Borland C++ del compilador de MSWLogo (ambos proyectos son similares, la diferencia entre ambos reside en que uno se utiliza para la depuración). Estos proyectos están en el archivo **LOGO32.IDE** o en el archivo **LOGO32D.IDE**. En el archivo **MAIN.CPP** es donde se deben realizar las modificaciones que consisten en introducir una línea que permita la carga del archivo que contiene el entorno para las controladoras (**Primitivas.LGO**).

```
void TMyApp::InitInstance()
{
    TApplication::InitInstance();
    HAccTable = LoadAccelerators("LOGOACC");
    fileload("ENTORNO/Primitivas.LGO");
}
```

El archivo *Primitivas.LGO* debe guardarse en una carpeta llamada *ENTORNO* (dentro de la carpeta *MSWLogo/bc5*) junto con el resto de archivos necesarios para crear el entorno de trabajo para las controladoras. La carpeta *ENTORNO* se creará y se dejará donde se tenga el proyecto del MSWLogo.

Una vez hecho esto se deben compilar los proyectos, creando así los nuevos ejecutables (**logo32.exe** para el proyecto *LOGO32.IDE* –este es el ejecutable que se utilizará- y **logo32d.exe** para el proyecto *LOGO32D.IDE*).

Si se abre el compilador de MSWLogo aparecerá la siguiente ventana:

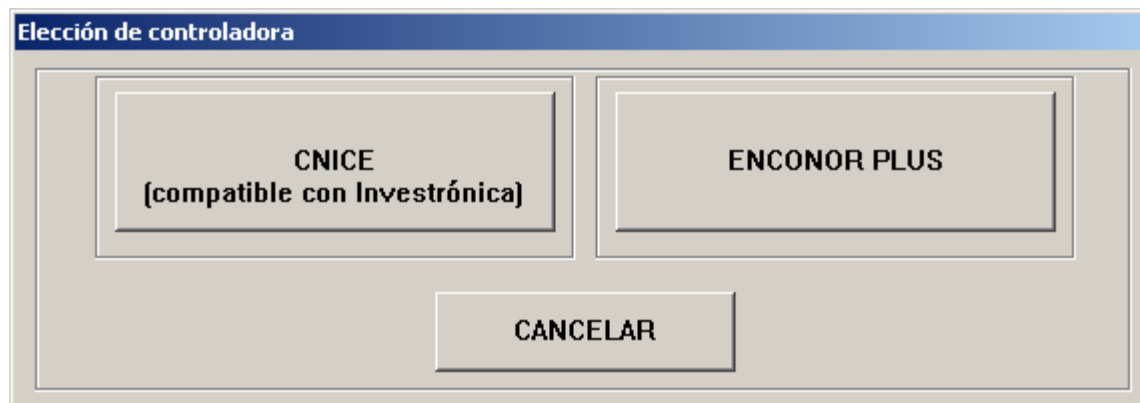


En ella se puede elegir entre tres opciones distintas:

1. **Usar una controladora**: permite trabajar en un entorno de trabajo para una controladora determinada.
2. **Simuladores**: permite trabajar en un entorno de trabajo para simuladores.
3. **Programar**: permite utilizar el compilador de MSWLogo sin haber cargado ningún entorno de trabajo especial.

Centrémonos en la primera opción, que es la que nos interesa en estos momentos.

Al elegir la opción de **Usar una controladora** se mostrará la ventana para elegir el tipo de controladora a manejar.



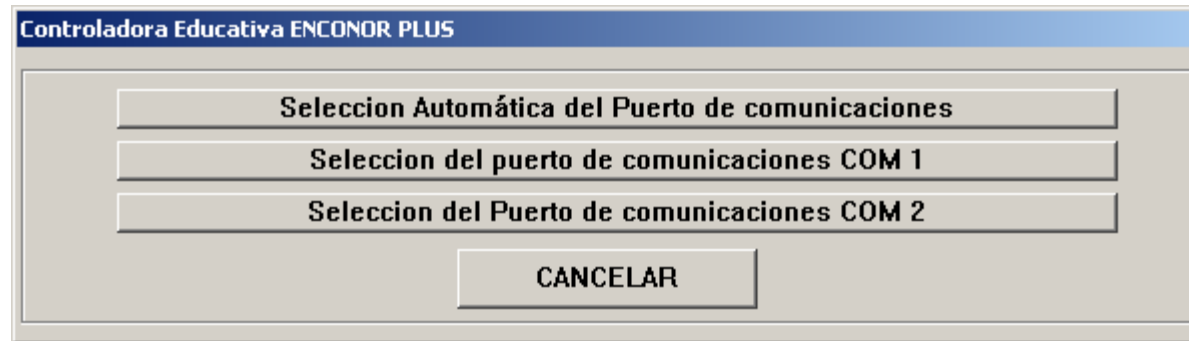
Como ya se ha visto hasta ahora, se ha creado un entorno de trabajo para poder trabajar con las controladoras ENCONOR y CNICE. Este entorno proporciona una serie de procedimientos que permiten controlar las entradas y salidas, tanto digitales como analógicas. Se ha tratado, en la medida de lo posible, de que dichos procedimientos tengan los mismos nombres para ambas controladoras, para facilitar así su utilización. Además también se proporciona un entorno gráfico para poder manejar las controladoras. En el momento en el que se elige el tipo de controladora con la que se desea trabajar los procedimientos de control son cargados y se puede empezar a trabajar con la controladora.

## 2 ENTORNO DE TRABAJO PARA LA CONTROLADORA ENCONOR

A continuación se va a explicar el entorno de trabajo creado para trabajar con la controladora ENCONOR. El entorno de trabajo creado permite manejar la controladora a través de un interfaz gráfico o a través de una serie de procedimientos o primitivas que actuarán directamente sobre las entradas y salidas analógicas y digitales de la controladora. La ventana que permite seleccionar el tipo de entorno con el que se desea trabajar es la que se muestra a continuación:



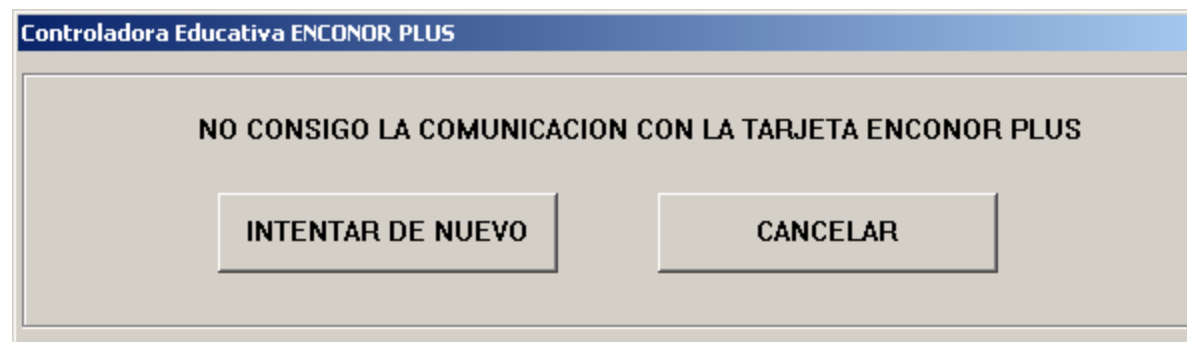
Tanto si se ha elegido trabajar con el entorno gráfico o la programación manual, lo siguiente es establecer el canal o puerto de comunicación. La ventana que permite al usuario elegir el puerto serie por el que establecer la comunicación entre el ordenador y la controladora ENCONOR es la siguiente:



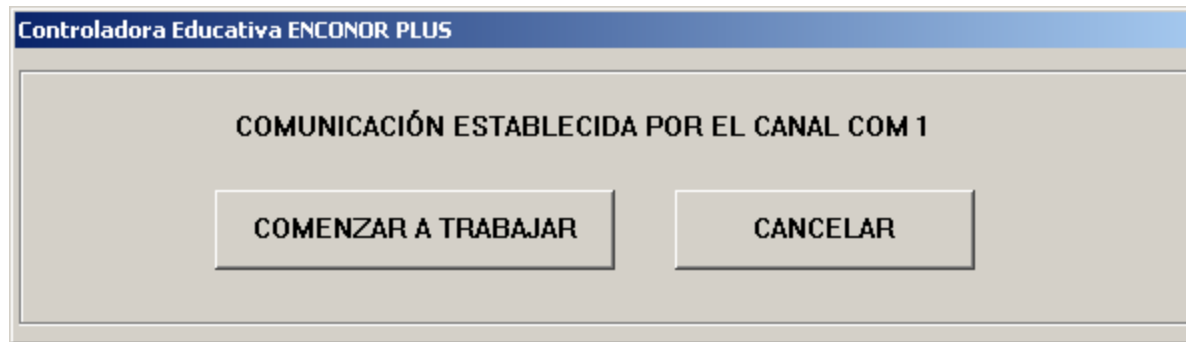
El significado de cada una de las opciones del menú es:

1.- Detección automática del canal de comunicaciones: El programa intenta determinar de forma autónoma el canal serie operativo para comunicarse con la tarjeta controladora. En la mayor parte de los casos esta opción es la más adecuada, y por tanto la primera a elegir. Sin embargo, puede haber algún ordenador que muestre alguna anomalía con esta opción; como por ejemplo que deje de funcionar el ratón o se bloquee el programa. En estos casos puntuales, hay que salir del programa, reiniciar el ordenador, entrar de nuevo y utilizar las otras dos opciones.

2.- Canal de comunicaciones COM 1: El programa intenta ponerse en comunicación con la tarjeta controladora a través del puerto COM 1. Si no consigue establecer la comunicación, aparecerá el correspondiente mensaje de aviso. Se pulsará el botón **INTENTAR DE NUEVO** y se probará con la opción 3 (Selección del Puerto de comunicaciones COM 2) en la ventana anterior.



Si, por el contrario, consigue establecer la comunicación mostrará un mensaje de aviso "Comunicación establecida por el canal 1". Ese será el canal a utilizar en el futuro con ese ordenador en concreto.



3.- Canal de comunicaciones COM 2: El programa intenta ponerse en comunicación con la tarjeta controladora a través del puerto COM 2. Si lo consigue mostrará un mensaje de aviso "Comunicación establecida por el canal 2". Ese será el canal a utilizar en el futuro con ese ordenador en concreto. Si no consigue establecer la comunicación, aparecerá el correspondiente mensaje de aviso.

En este caso, si no funciona ninguna de las tres opciones, puede que haya algún problema en la tarjeta controladora, cable de conexión o ambos. Comprobar que la placa tenga el diodo led amarillo parpadeando, las soldaduras sean correctas, los componentes estén bien colocados y el cable correctamente conectado. Después de asegurarse y tener en cuenta los apartados anteriores repetir el proceso de inicialización.

Si se optó por trabajar con el entorno gráfico y se ha conseguido establecer la comunicación entre la controladora y el ordenador, se debe pulsar el botón **COMENZAR A TRABAJAR**. Aparecerá un entorno gráfico que nos permitirá manipular las salidas digitales una a una o en bloque a golpe de ratón. Además en todo momento podremos ordenar que se visualice el estado de las entradas tanto analógicas como digitales. El valor de tensión de las salidas analógicas también se puede modificar.

Entradas y salidas analógicas

ENTRADAS

54321

SALIDAS

- Salida Analog. 1 +

◀◻▶

- Salida Analog. 2 +

◀◻▶

- Salida Analog. 3 +

◀◻▶

- Salida Analog. 4 +

◀◻▶

Cerrar ventana

Entradas y salidas digitales

SALIDAS

87654321

◻◻◻◻◻◻◻◻

Todas

Cerrar ventana

ENTRADAS

87654321

Leer Entradas

Comunicación Serie

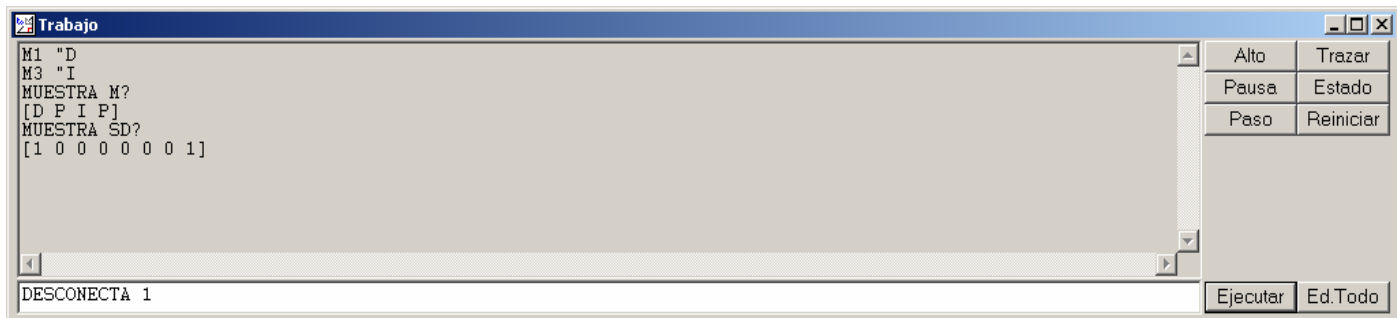
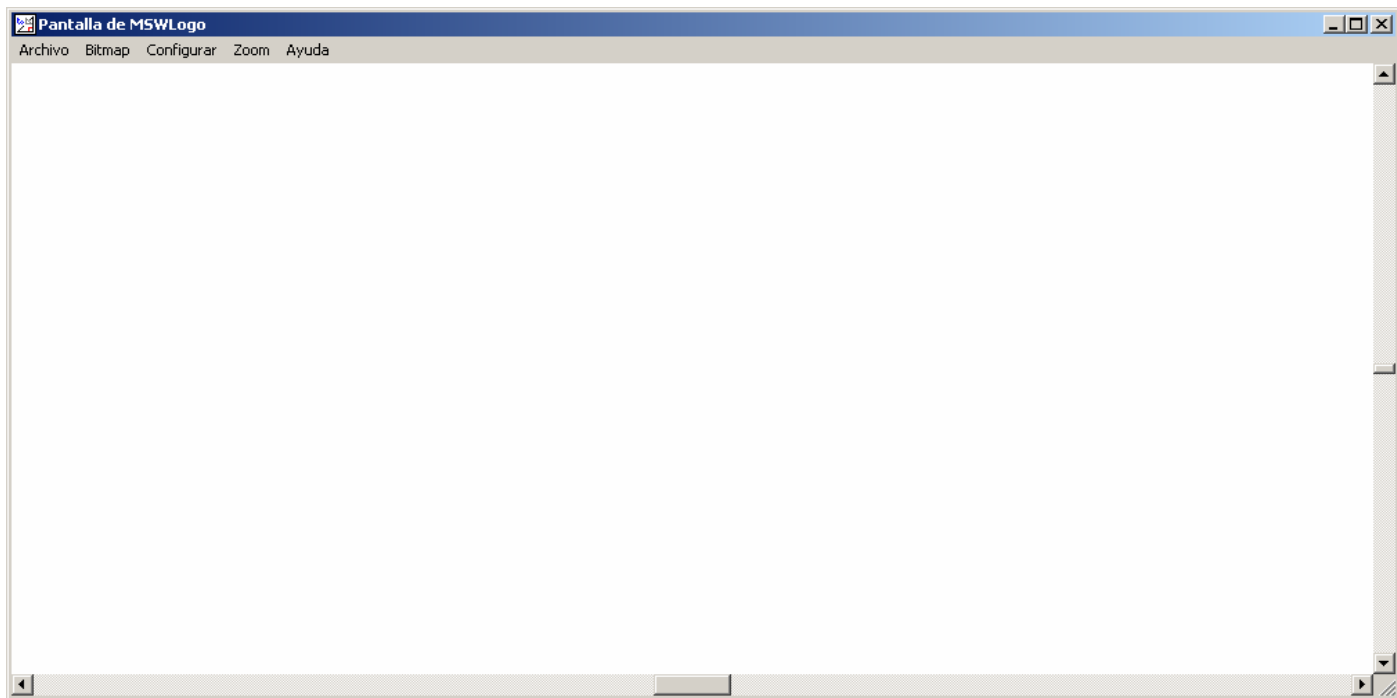
PUERTO: COM 1

Comunicación establecida

Cerrar ventana

Si, por el contrario, se optó por trabajar en el *entorno de Programación manual* no aparecerá el interfaz gráfico. La pantalla de MSWLogo se divide en dos ventanas. La superior será la ventana de mensajes y la inferior la de trabajo, es decir, donde escribiremos las órdenes o primitivas.





Destacar que se puede pasar de un entorno de trabajo a otro, es decir, si estamos en *Programación Manual* podemos pasar al *Entorno gráfico* (con la primitiva **graficos**) y viceversa (escribiendo las primitivas que se deseen en la ventana de trabajo del MSWLogo).

Los procedimientos que se cargan en memoria para poder trabajar en uno de los dos entornos de trabajo para esta controladora se clasifican en tres tipos: los procedimientos necesarios para establecer el puerto de comunicación entre el ordenador y la controladora, las primitivas de control que son los procedimientos que servirán para manejar la controladora y los procedimientos necesarios para crear el entorno gráfico para manejar la controladora. El código de dichos procedimientos se encuentra en el **ANEXO 1** de esta documentación.

Destacar que tiene especial importancia para la mayor parte de los procedimientos la **variable :canal** que almacena el puerto por el que se establece la comunicación entre el ordenador y la controladora (COM1 o COM2). Inicialmente no hay establecido ningún canal de comunicaciones, por ello se inicializa la variable al valor -1 (**HAZ "canal -1**).

También es importante el **procedimiento EntornoENCONOR** que es el cuerpo principal del programa y donde se inicializan las variables necesarias para los procedimientos.

## **2.1 Procedimientos para establecer el canal de comunicación:**

La controladora ENCONOR se comunica con el ordenador a través del puerto RS232 (COM1 o COM2). El entorno de trabajo creado permite elegir el canal de comunicación. Para ello se han definido los procedimientos *BuscaPuerto*, *Arranca*, *Arrancar?*, *Arrancar1*, *Arrancar2* y *BuscaPuerto1*.

PROCEDIMIENTO	DESCRIPCIÓN
<b><i>Buscapuerto</i></b>	Crea una ventana para que el usuario elija el canal de comunicación. Se puede elegir directamente el puerto serie por el que se quiere establecer la comunicación (COM1 o COM2) o dejar que dicha elección se haga de forma automática.
<b><i>Arranca</i></b>	Intenta comunicarse con la placa tres veces antes de decidir que no hay posibilidad de comunicación con la tarjeta controladora.

<b><i>Arrancar?</i></b>	Detecta de forma automática si el canal de comunicaciones es el COM1 o el COM2.
<b><i>Arrancar1</i></b>	Abre y configura el puerto de comunicaciones COM1.
<b><i>Arrancar2</i></b>	Abre y configura el puerto de comunicaciones COM2.
<b><i>BuscaPuerto1</i></b>	Comunica al usuario con una ventana si se ha podido o no establecer la comunicación entre el ordenador y la controladora. Si ésta no ha sido posible, permite volver a intentarlo de nuevo.

## 2.2 Primitivas de control:

A continuación se presentan las distintas primitivas de control disponibles para manejar la controladora ENCONOR, separadas según el uso que tengan, junto con los procedimientos auxiliares necesarios para las mismas:

### 2.2.1 Conjunto de primitivas de control diseñadas para trabajar con los actuadores conectados a las salidas digitales:

PRIMITIVA	DESCRIPCIÓN	PROCEDIMIENTOS AUXILIARES	
<b><i>M1 “D</i></b>	Hace girar en un determinado sentido un motor conectado a las salidas digitales S1-S5 (para conseguirlo se activan las dos salidas). Si el actuador es una bombilla, relé o electroimán, simplemente lo activa.	Procedimiento	Descripción
		<b><i>ACTIVA_SALIDA :numerosalida</i></b>	Activa la salida digital que se le indique en <i>:numerosalida</i> , dejando el resto de salidas como estén

<b><i>M1 “I</i></b>	Hace girar en sentido contrario a la primitiva anterior un motor conectado a las salidas digitales S1-S5 (para conseguirlo se activa la salida S1 y se desactiva la salida S5). No se diferencia de la primitiva anterior si están conectados otros actuadores (bombilla, relé, etc.).		como estén.
<b><i>M1 “P</i></b>	Desactiva el actuador que esté conectado a S1-S5 (para conseguirlo se desactivan ambas salidas).	<b><i>DESACTIVA_SALIDA :numerosalida</i></b>	Desactiva la salida digital que se le indique en :numerosalida, dejando el resto de salidas como estén.
<b><i>M2 “D, M2 “I, M2 “P M3 “D, M3 “I, M3 “P M4 “D, M4 “I, M2 “P</i></b>	Lo mismo que las primitivas anteriores, aunque los actuadores se conectan a los restantes pares de salidas digitales (S2-S6, S3-S7 y S4-S8).		
<b><i>M?</i></b>	Devuelve una lista con el estado de los cuatro motores. Ej.: MUESTRA M? [P D D I]		
<b><i>M :L</i></b>	Permite activar/desactivar todos los motores simultáneamente. El parámetro :L tiene que ser una lista. Ej.: M [P P I D]→ Se desactivan los motores M1 y M2, M3 se hace girar a la Izquierda y M4 se hace girar a la derecha.		
<b><i>CONECTADO?</i></b>	Devuelve una lista de 4 elementos (uno por cada motor) cuyos valores son 0 ó 1 según estén activos o no.		
<b><i>SALIDA :NUM</i></b>	Controla individualmente o en conjunto cada una de las ocho salidas digitales. Envía a las salidas digitales el número binario equivalente al decimal que pongamos en :NUM. Para activar, por ejemplo, las salidas 1 y 8 se escribirá el comando:	<b>Procedimiento</b>	<b>Descripción</b>
		<b><i>SALIDA1 :dato</i></b>	Se encarga de escribir en el puerto los datos necesarios para poder controlar las salidas digitales.

	SALIDA 129 Si alguna otra salida digital distinta a las salidas 1 y 8 estaba activada, se desactivará.					
<i>EnviaOcteto :dato</i>	Procedimiento para la compatibilidad con las primitivas proporcionadas por ENCONOR (tiene la misma función que la primitiva <i>SALIDA</i> ). Controla individualmente o en conjunto cada una de las ocho salidas digitales. Envía a las salidas digitales el número binario equivalente al decimal que pongamos en <i>:dato</i> .					
<i>APAGA :n</i>	Procedimiento para la compatibilidad con las primitivas proporcionadas por ENCONOR (tiene la misma función que el procedimiento <i>DEACTIVA_SALIDA</i> ). Desconecta la salida digital que se le indique en <i>:n</i> (1, 2, ... u 8), dejando el resto de salidas digitales como estén.	<table><tr><th>Procedimiento</th><th>Descripción</th></tr><tr><td><i>DEACTIVA_SALIDA :numerosalida</i></td><td>Desactiva la salida digital que se le indique en <i>:numerosalida</i>, dejando el resto de salidas como estén.</td></tr></table>	Procedimiento	Descripción	<i>DEACTIVA_SALIDA :numerosalida</i>	Desactiva la salida digital que se le indique en <i>:numerosalida</i> , dejando el resto de salidas como estén.
Procedimiento	Descripción					
<i>DEACTIVA_SALIDA :numerosalida</i>	Desactiva la salida digital que se le indique en <i>:numerosalida</i> , dejando el resto de salidas como estén.					
<i>SALIDA?</i>	Devuelve una lista de ocho elementos con el estado de las ocho salidas digitales (1 si la salida está activada o 0 si está desactivada).					
<i>CONECTA :NUM</i>	Activa la salida digital indicada en <i>:NUM</i> , siendo <i>:NUM</i> 1, 2, 3, 4, 5, 6, 7 u 8. El resto de salidas digitales se quedan como estén.					
<i>DESCONECTA :NUM</i>	Desactiva la salida digital indicada en <i>:NUM</i> siendo <i>:NUM</i> 1, 2, 3, 4, 5, 6, 7 u 8. El resto de salidas digitales se quedan como estén.					
<i>CONECTAR</i>	Conecta las 8 salidas digitales (si no lo están ya).					
<i>DESCONECTAR</i>	Desconecta las 8 salidas digitales (si es que están)					

	conectadas).
<b>VS?</b>	Devuelve un número decimal que indica qué salidas digitales están activadas. Por ejemplo, si devuelve 3 indica que las salidas digitales 1 y 3 están activadas.
<b>segundos :tiempo</b>	Produce un retardo o espera de tantos segundos como le indiquemos en :tiempo. Se pueden utilizar fracciones de segundo (Ej.: 0.2).

Destacar que en una variable llamada **salidas** es donde se almacena el valor/estado de las ocho salidas digitales, mediante un número decimal que indica el valor en binario de dichas salidas. Por ejemplo, si **salidas** toma el valor 255 quiere decir que todas las salidas digitales están activadas.

### 2.2.2 Conjunto de primitivas de control diseñadas para trabajar con las entradas digitales:

PRIMITIVA	DESCRIPCIÓN	PROCEDIMIENTOS AUXILIARES	
<b>entrada :n</b>	Procedimiento para la compatibilidad con las primitivas proporcionadas por ENCONOR. Devuelve VERDADERO si la entrada digital indicada en :n (1, 2, ... u 8) está conectado o FALSO en caso contrario.	Procedimiento	Descripción
<b>VE?</b>	Devuelve un número decimal que indica qué entradas digitales están activadas. Por ejemplo, si ejecutamos la primitiva y esta devuelve 129, las entradas digitales 1 y 8 tienen valor 1.	<b>Recibir</b>	Intenta leer del puerto por 3 veces los datos que proporcionan las entradas digitales, utilizando para ello el procedimiento <i>Recibir1</i> .

<b><i>SD :entrada</i></b>	Devuelve el valor del sensor digital expresado en <i>:entrada</i> (1, 2, 3, ... u 8), 1 si la entrada está activa o 0 en caso contrario.	<b><i>Recibir1</i></b>	Se encarga de escribir y leer del puerto los datos necesarios para obtener el valor de las entradas digitales.
<b><i>SD?</i></b>	Devuelve el estado de los ocho sensores digitales en forma de lista (0s ó 1s).		
<b><i>EsperaOn :entrada</i></b>	Deja el programa parado a la espera de que se active la entrada digital indicada en <i>:entrada</i> .		
<b><i>EsperaOff :entrada</i></b>	Deja el programa parado a la espera de que se desactive la entrada digital indicada en <i>:entrada</i> .		

Destacar que en una variable llamada ***entradas*** es donde se almacena el valor/estado de las ocho entradas digitales, mediante un número decimal que indica el valor en binario de dichas entradas. Por ejemplo, si *entradas* toma el valor 255 quiere decir que todas las entradas digitales están activadas.

### 2.2.3 Conjunto de primitivas de control diseñadas para trabajar con las entradas analógicas:

PRIMITIVA	DESCRIPCIÓN	PROCEDIMIENTOS AUXILIARES	
<b><i>leeanalógica :dato</i></b>	Procedimiento para la compatibilidad con las primitivas proporcionadas por ENCONOR. Intenta leer del puerto por 3 veces los datos que proporcionan la entrada analógica indicada en <i>:dato</i> (que puede ser 1, 2, 3, 4 ó 5). Si consigue leerlas, devuelve su valor. Para obtener el valor en voltios correspondiente al valor decimal devuelto por la primitiva se debe aplicar la siguiente fórmula: <b>Nº decimal= (Voltaje Entrada mV) / (19'5 mV)</b>	<b>Procedimiento</b>	<b>Descripción</b>

<b><i>SAV</i></b>	Devuelve directamente el valor en voltios del sensor analógico conectado a la entrada analógica 1.	<b><i>leeanalógico1 :dato</i></b>	Se encarga de escribir y leer del puerto los datos necesarios para obtener el valor de la entrada analógica indicada en <i>:dato</i> .
<b><i>SAW</i></b>	Devuelve directamente el valor en voltios del sensor analógico conectado a la entrada analógica 2.		
<b><i>SAX</i></b>	Devuelve directamente el valor en voltios del sensor analógico conectado a la entrada analógica 3.		
<b><i>SAY</i></b>	Devuelve directamente el valor en voltios del sensor analógico conectado a la entrada analógica 4.		
<b><i>SAZ</i></b>	Devuelve directamente el valor en voltios del sensor analógico conectado a la entrada analógica 5.		
<b><i>SA?</i></b>	Devuelve una lista con el valor en voltios de los cinco sensores analógicos.		

Destacar que las entradas analógicas que queden al aire deben conectarse a MASA para no obtener valores de lectura erróneos, es decir, las entradas analógicas de las que no se quieran recibir o leer datos deben conectarse a MASA.

También es importante saber que en una variable llamada ***analógico*** es donde se almacena el valor de la entrada analógica que se esté leyendo en cada momento. El valor que puede tomar es un número entre 0 y 255 que corresponde al valor en tensión existente en la entrada analógica leída.



## 2.2.4 Conjunto de primitivas de control diseñadas para trabajar con las salidas analógicas:

PRIMITIVA	DESCRIPCIÓN	PROCEDIMIENTOS AUXILIARES	
<i>salidaanalogica :NUM :VALOR</i>	Intenta escribir en el puerto por 3 veces los datos necesarios para poner en la salida analógica que se le indica en :NUM (que puede ser 1, 2, 3 ó 4) el valor en tensión indicado en :VALOR (varía entre 0 y 255). Para conocer el valor en tensión en la salida se utiliza la siguiente fórmula: <b>Voltaje Salida (Voltios) = :VALOR / 23.8</b>	<b>Procedimiento</b>	<b>Descripción</b>
<i>VOLTAJE :NUM :VALOR</i>	Fija en la salida analógica indicada en :NUM el valor en tensión especificado en :VALOR (varía entre 0 y 10.5).	<i>potenciometro1 :num :valor</i>	Se encarga de escribir en el puerto los datos necesarios para poner en la salida analógica indicada en :num el valor en voltios indicado en :valor.
<i>VOLTAJE?</i>	Devuelve una lista con el valor en voltios de todas las salidas analógicas.		

Destacar que en se utilizan unas variables llamadas **salanalogica1**, **salanalogica2**, **salanalogica3** y **salanalogica4** para almacenar el valor que presentan las distintas salidas analógicas. Para conocer, por ejemplo, el valor en tensión en la salida digital 1 utiliza la siguiente fórmula:

**Voltaje Salida1 (Voltios) = :salanalogica1 / 23.8**

### 2.3 Procedimientos para crear el entorno gráfico:

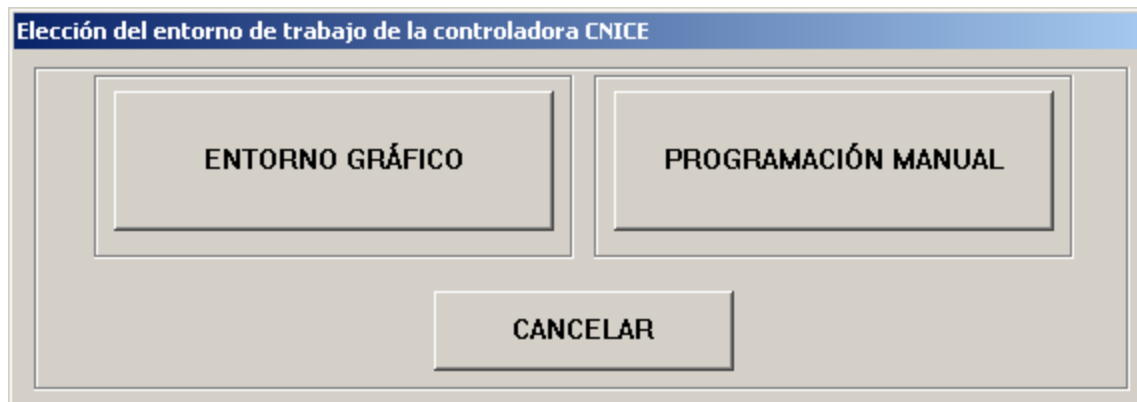
A continuación se presentan las distintas primitivas disponibles para crear el entorno gráfico para manejar la controladora ENCONOR con los procedimientos auxiliares necesarios para las mismas:

PRIMITIVA	DESCRIPCIÓN	PROCEDIMIENTOS AUXILIARES	
<i>graficos</i>	Crea las ventanas necesarias para manejar las entradas y salidas analógicas (primitiva <i>graficosan</i> ) y digitales y la ventana para informar del estado de la comunicación entre el ordenador y la controladora (primitiva <i>informa</i> ).	Procedimiento	Descripción
		<i>pintasalidas</i>	Actualiza el valor de los indicadores de las salidas digitales del entorno gráfico.
		<i>pintaentradas</i>	Actualiza el valor de los indicadores de las entradas digitales del entorno gráfico.
		<i>Recibir</i>	Intenta leer del puerto por 3 veces los datos que proporcionan las entradas digitales, utilizando para ello el procedimiento <i>Recibir1</i> .
		<i>Recibir1</i>	Se encarga de escribir y leer del puerto los datos necesarios para obtener el valor de las entradas digitales.
<i>graficosan</i>	Crea una ventana para controlar las entradas y salidas analógicas.		
<i>informa</i>	Crea una ventana para informar de si se ha podido establecer la comunicación por el puerto serie (en este caso indica por qué puerto) o no.		

### 3 ENTORNO DE TRABAJO PARA LA CONTROLADORA CNICE

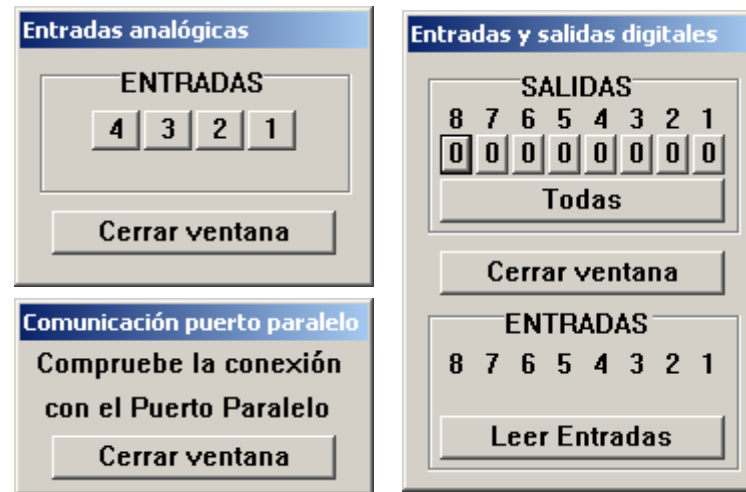
A continuación se va a explicar el entorno de trabajo creado para trabajar con la controladora CNICE. **Hay que destacar que este entorno de trabajo permite trabajar tanto con la controladora CNICE como con la controladora Investrónica.**

El entorno de trabajo creado permite manejar la controladora a través de un interfaz gráfico o a través de una serie de procedimientos o primitivas que actuarán directamente sobre las entradas y salidas analógicas y digitales de la controladora. La ventana que permite seleccionar el tipo de entorno con el que se desea trabajar es la que se muestra a continuación:

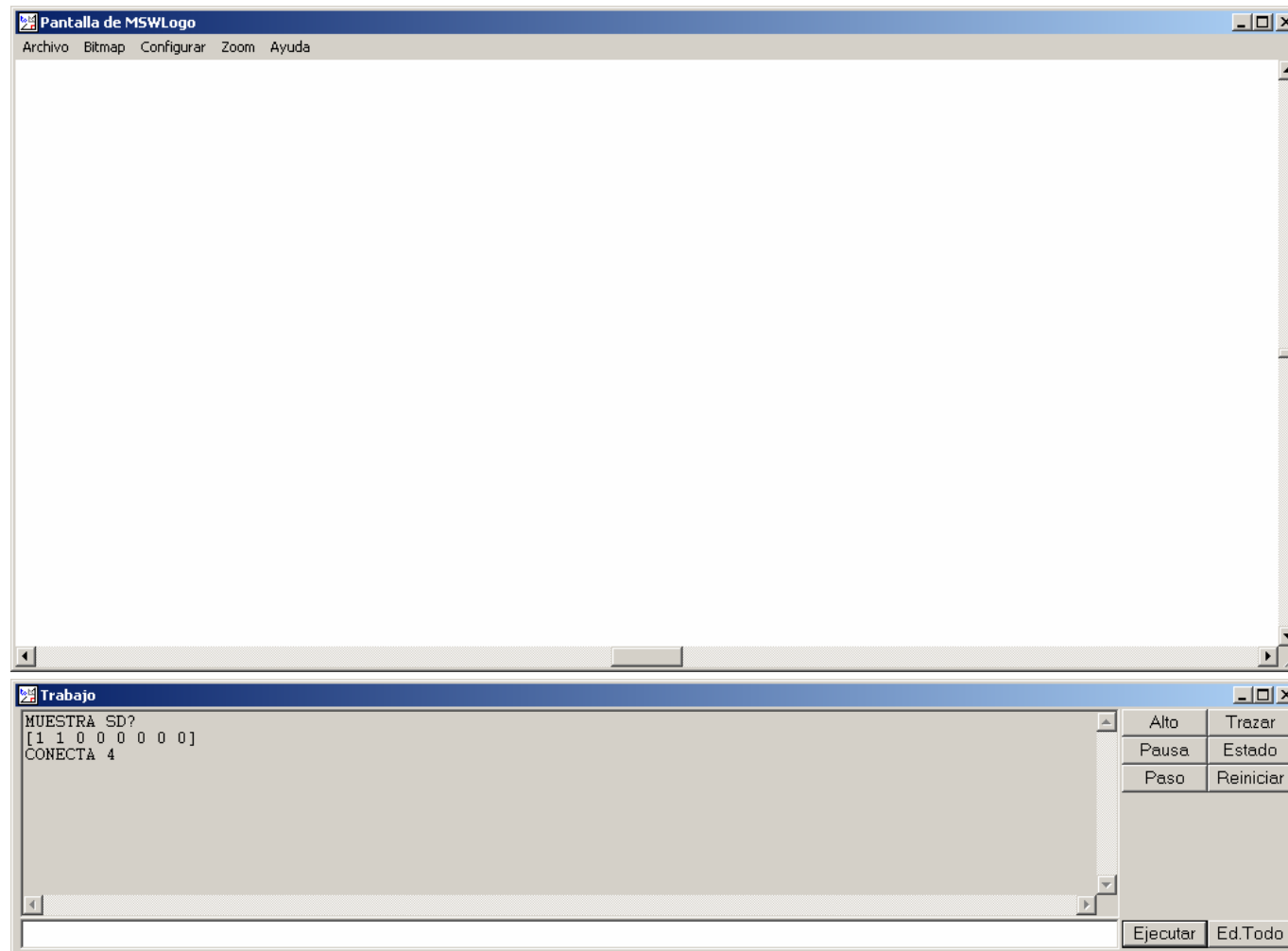


Tanto si se opta por uno u otro entorno de trabajo, se carga la **librería io.dll** para que a través de sus funciones **"PortOut"** y **"PortIn"**, se pueda tener acceso a los puertos del PC cuando el sistema operativo es Windows 2000 o XP. Pero antes de nada hay que asegurarse de que esta librería, se encuentre en el sistema, y mas concretamente, en el directorio **C:\WINNT\System**.

Si se opta por trabajar con el entorno gráfico se debe pulsar el botón **ENTORNO GRÁFICO**. Aparecerá un entorno gráfico que nos permitirá manipular las salidas digitales una a una o en bloque a golpe de ratón. Además en todo momento podremos ordenar que se visualice el estado de las entradas tanto analógicas como digitales.



Si, por el contrario, se optó por trabajar en el entorno de *Programación manual* no aparecerá el interfaz gráfico. La pantalla de MSWLogo se divide en dos ventanas. La superior será la ventana de mensajes y la inferior la de trabajo, es decir, donde escribiremos las órdenes o primitivas.



Destacar que se puede pasar de un entorno de trabajo a otro, es decir, si estamos en *Programación Manual* podemos pasar al *Entorno gráfico* (con la primitiva **graficos**) y viceversa (escribiendo las primitivas que se deseen en la ventana de trabajo del MSWLogo).

Los procedimientos que se han implementado permiten establecer la comunicación entre la controladora y el PC a través del puerto paralelo LPT1.

Los procedimientos que se cargan en memoria para poder trabajar en uno de los dos entornos de trabajo para esta controladora se clasifican en dos tipos: las primitivas de control que son los procedimientos que servirán para manejar la controladora y los procedimientos necesarios para crear el entorno gráfico para manejar la controladora. El código de dichos procedimientos se encuentra en el **ANEXO 2** de esta documentación.

Destacar que tiene especial importancia el **procedimiento EntornoCNICE** que es el cuerpo principal del programa y donde se inicializan las variables necesarias para los procedimientos.

### **3.1 Primitivas de control**

A continuación se presentan las distintas primitivas de control disponibles para manejar la controladora CNICE, separadas según el uso que tengan, junto con los procedimientos auxiliares necesarios para las mismas:

#### **3.1.1 Conjunto de primitivas de control diseñadas para trabajar con los actuadores conectados a las salidas digitales:**

PRIMITIVA	DESCRIPCIÓN	PROCEDIMIENTOS AUXILIARES	
<b>MI “D</b>	Hace girar en un determinado sentido un motor conectado a las salidas digitales S0-S1 (para conseguirlo se activa la salida S1 y se desactiva S0). Si el actuador es una bombilla, relé o electroimán, simplemente lo activa.	Procedimiento	Descripción
		<b>ACTIVAR_SALIDA_X (*)</b>  X indica el número de salida que se quiere activar (1 2 8)	Activa la salida digital que se le indique en X, dejando el resto de salidas como estén.

<b><i>M1 “I</i></b>	Hace girar en sentido contrario a la primitiva anterior un motor conectado a las salidas digitales S0-S1 (para conseguirlo se activa la salida S0 y se desactiva la salida S1). No se diferencia de la primitiva anterior si están conectados otros actuadores (bombilla, relé, etc.)
<b><i>M1 “P</i></b>	Desactiva el actuador que esté conectado a S0-S1 (para conseguirlo se desactivan ambas salidas).
<b><i>M2 “D, M2 “I, M2 “P M3 “D, M3 “I, M3 “P M4 “D, M4 “I, M2 “P</i></b>	Lo mismo que las primitivas anteriores, aunque los actuadores se conectan a los restantes pares de salidas digitales (S2-S3, S4-S5 y S6-S7).
<b><i>M?</i></b>	Devuelve una lista con el estado de los cuatro motores. Ej.: MUESTRA M? <i>[P D D I]</i>
<b><i>M :L</i></b>	Permite activar/desactivar todos los motores simultáneamente. El parámetro :L tiene que ser una lista. Ej.: M [P P I D]→ Se desactivan los motores M1 y M2, M3 se hace girar a la izquierda y M4 se hace girar a la derecha.
<b><i>CONECTADO?</i></b>	Devuelve una lista de 4 elementos (uno por cada motor) cuyos valores son 0 ó 1 según estén activos o no.

quiere activar (1, 2, ..., 8).	
<b><i>DESACTIVAR_SALIDA_X (*)</i></b>  X indica el número de salida que se quiere desactivar (1, 2, ..., 8).	Desactiva la salida digital que se le indique en X, dejando el resto de salidas como estén.
<b><i>SALIDA_DIGITAL :VALOR</i></b>	Pone en el puerto de control la palabra necesaria para activar/desactivar las salidas digitales.
<p>(*) <b><u>Nota:</u></b> en el nombre del procedimiento se debe sustituir la X por el número de salida correspondiente, por ejemplo, si se quiere hacer girar a la izquierda el motor 1 (conectado a las salidas digitales S0-S1) se usará la primitiva <i>M1 “I</i>, la cual llamará a los procedimientos:</p> <p style="text-align: center;"><i>ACTIVAR_SALIDA_1</i> <i>DESACTIVAR_SALIDA_2</i></p>	

<b><i>SALIDA :NUM</i></b>	Controla individualmente o en conjunto cada una de las ocho salidas digitales. Envía a las salidas digitales el número binario equivalente al decimal que pongamos en <i>:NUM</i> . Para activar, por ejemplo, las salidas 1 y 8 se escribirá el comando: <i>SALIDA 129</i> Si alguna otra salida digital distinta a las salidas 1 y 8 estaba activada, se desactivará.	<table><tr><th>Procedimiento</th><th>Descripción</th></tr><tr><td><b><i>ACTIVAR_SALIDA_X</i></b> X indica el número de salida que se quiere activar (1, 2, ..., 8).</td><td>Activa la salida digital que se le indique en <i>:numerosalida</i>, dejando el resto de salidas como estén.</td></tr><tr><td><b><i>DESACTIVAR_SALIDA_X</i></b>  X indica el número de salida que se quiere desactivar (1, 2, ..., 8).</td><td>Desactiva la salida digital que se le indique en <i>X</i>, dejando el resto de salidas como estén.</td></tr><tr><td><b><i>SALIDA_DIGITAL :VALOR</i></b></td><td>Pone en el puerto de control la palabra necesaria para activar/desactivar las salidas digitales.</td></tr></table>	Procedimiento	Descripción	<b><i>ACTIVAR_SALIDA_X</i></b> X indica el número de salida que se quiere activar (1, 2, ..., 8).	Activa la salida digital que se le indique en <i>:numerosalida</i> , dejando el resto de salidas como estén.	<b><i>DESACTIVAR_SALIDA_X</i></b>  X indica el número de salida que se quiere desactivar (1, 2, ..., 8).	Desactiva la salida digital que se le indique en <i>X</i> , dejando el resto de salidas como estén.	<b><i>SALIDA_DIGITAL :VALOR</i></b>	Pone en el puerto de control la palabra necesaria para activar/desactivar las salidas digitales.
Procedimiento	Descripción									
<b><i>ACTIVAR_SALIDA_X</i></b> X indica el número de salida que se quiere activar (1, 2, ..., 8).	Activa la salida digital que se le indique en <i>:numerosalida</i> , dejando el resto de salidas como estén.									
<b><i>DESACTIVAR_SALIDA_X</i></b>  X indica el número de salida que se quiere desactivar (1, 2, ..., 8).	Desactiva la salida digital que se le indique en <i>X</i> , dejando el resto de salidas como estén.									
<b><i>SALIDA_DIGITAL :VALOR</i></b>	Pone en el puerto de control la palabra necesaria para activar/desactivar las salidas digitales.									
<b><i>EnviaOcteto :dato</i></b>	Procedimiento para la compatibilidad con las primitivas proporcionadas por ENCONOR (tiene la misma función que la primitiva <i>SALIDA</i> ). Controla individualmente o en conjunto cada una de las ocho salidas digitales. Envía a las salidas digitales el número binario equivalente al decimal que pongamos en <i>:dato</i> .									
<b><i>APAGA :n</i></b>	Procedimiento para la compatibilidad con las primitivas proporcionadas por ENCONOR (tiene la misma función que el procedimiento <i>DESACTIVAR_SALIDA_X</i> ). Desconecta la salida digital que se le indique en <i>:n</i> (1, 2, ... u 8), dejando el resto de salidas digitales como estén.	<table><tr><th>Procedimiento</th><th>Descripción</th></tr><tr><td><b><i>DESACTIVAR_SALIDA_X</i></b>  X indica el número de salida que se quiere desactivar (1, 2, ..., 8).</td><td>Desactiva la salida digital que se le indique en <i>X</i>, dejando el resto de salidas como estén.</td></tr></table>	Procedimiento	Descripción	<b><i>DESACTIVAR_SALIDA_X</i></b>  X indica el número de salida que se quiere desactivar (1, 2, ..., 8).	Desactiva la salida digital que se le indique en <i>X</i> , dejando el resto de salidas como estén.				
Procedimiento	Descripción									
<b><i>DESACTIVAR_SALIDA_X</i></b>  X indica el número de salida que se quiere desactivar (1, 2, ..., 8).	Desactiva la salida digital que se le indique en <i>X</i> , dejando el resto de salidas como estén.									



		<b><i>SALIDA_DIGITAL :VALOR</i></b>	Pone en el puerto de control la palabra necesaria para activar/desactivar las salidas digitales (valor entre 0 y 255).
<b><i>SALIDA?</i></b>	Devuelve una lista de ocho elementos con el estado de las ocho salidas digitales (1 si la salida está activada o 0 si está desactivada).		
<b><i>CONECTA :NUM</i></b>	Activa la salida digital indicada en <i>:NUM</i> , siendo <i>:NUM</i> 1, 2, 3, 4, 5, 6, 7 u 8. El resto de salidas digitales se quedan como estén		
<b><i>DESCONECTA :NUM</i></b>	Desactiva la salida digital indicada en <i>:NUM</i> siendo <i>:NUM</i> 1, 2, 3, 4, 5, 6, 7 u 8. El resto de salidas digitales se quedan como estén.		
<b><i>CONECTAR</i></b>	Conecta las 8 salidas digitales (si no lo están ya).		
<b><i>DESCONECTAR</i></b>	Desconecta las 8 salidas digitales (si es que están conectadas).		
<b><i>VS?</i></b>	Devuelve un número decimal que indica qué salidas digitales están activadas. Por ejemplo, si devuelve 3 indica que las salidas digitales 1 y 3 están activadas.		
<b><i>segundos :tiempo</i></b>	Produce un retardo o espera de tantos segundos como le indiquemos en <i>:tiempo</i> . Se pueden utilizar fracciones de segundo (Ej.: 0.2).		

Destacar que en una variable llamada ***ESTADO\_SALIDAS*** es donde se almacena el valor/estado de las ocho salidas digitales, mediante un número decimal que indica el valor en binario de dichas salidas. Por ejemplo, si ***ESTADO\_SALIDAS*** toma el valor 255 quiere decir que todas las salidas digitales están activadas.

También hay una variable por cada salida llamada **ESTADO\_SALIDA\_X**, donde X indica el número de salida (1, 2, ..., 8). Estas variables tomarán valor 1 ó 0 dependiendo si la salida correspondiente está activada o no.

### 3.1.2 Conjunto de primitivas de control diseñadas para trabajar con las entradas digitales:

PRIMITIVA	DESCRIPCIÓN	PROCEDIMIENTOS AUXILIARES	
<i>entrada :n</i>	Procedimiento para la compatibilidad con las primitivas proporcionadas por ENCONOR. Devuelve VERDADERO si la entrada digital indicada en :n está conectado o FALSO en caso contrario.	Procedimiento	Descripción
		<b>LEER_ENTRADA_DIGITAL_BAJA</b>	Lee y escribe en el puerto los datos necesarios para obtener el valor en decimal de las entradas digitales bajas (1, 2, 3 y 4).
<i>VE?</i>	Devuelve un número decimal que indica qué entradas digitales están activadas. Por ejemplo, si ejecutamos la primitiva y esta devuelve 129, las entradas digitales 1 y 8 tienen valor 1.	<b>LEER_ENTRADA_DIGITAL_ALTA</b>	Lee y escribe en el puerto los datos necesarios para obtener el valor en decimal de las entradas digitales altas (5, 6, 7 y 8).
<i>SD :entrada</i>	Devuelve el valor del sensor digital expresado en :entrada (1, 2, 3, ... , 8), 1 si la entrada está activa ó 0 en caso contrario.		
<i>SD?</i>	Devuelve el estado de los ocho sensores digitales en forma de lista (0s ó 1s).	Procedimiento	Descripción
		<b>SDB?</b>	Devuelve una lista con el estado de la parte baja de las entradas digitales.
		<b>SDA?</b>	Devuelve una lista con el estado de la parte alta de las entradas digitales.
<i>EsperaOn :entrada</i>	Deja el programa parado a la espera de que se active la entrada digital indicada en :entrada.		

<b><i>EsperaOff:entrada</i></b>	Deja el programa parado a la espera de que se desactive la entrada digital indicada en <i>:entrada</i> .
---------------------------------	--

Hay que recordar que en esta controladora se utiliza lógica negativa para las entradas digitales, es decir, cuando las entradas están al aire o se les mete un 1, en ellas se lee un 0, por el contrario, si se le mete un 0 en la entrada se leerá un 1.

También debemos saber que la lectura de las entradas, tanto las analógicas como las digitales, se realizan en dos partes, primero leyendo la parte alta y después la parte baja del dato de entrada.

### 3.1.3 Conjunto de primitivas de control diseñadas para trabajar con las entradas analógicas:

PRIMITIVA	DESCRIPCIÓN	PROCEDIMIENTOS AUXILIARES	
<b><i>leeanalogica :dato</i></b>	Procedimiento para la compatibilidad con las primitivas proporcionadas por ENCONOR. Lee y escribe en el puerto los datos necesarios para obtener el valor decimal (valor entre 0 y 255) que proporciona la entrada analógica indicada en <i>:dato</i> (que puede ser 1, 2, 3 ó 4. Si consigue leerlas, devuelve su valor. Para obtener el valor en voltios correspondiente al valor decimal devuelto por la primitiva se debe aplicar la siguiente fórmula: <b><math>\text{Voltaje Entrada (V)} = (\text{N}^\circ \text{ decimal} * 5) / 256</math></b>	<b>Procedimiento</b>	<b>Descripción</b>
<b><i>SAW</i></b>	Devuelve directamente el valor en voltios del sensor analógico conectado a la entrada analógica IN0.	<b><i>TENSION_DE_ENTRADA_ANALOGICA :CANAL</i></b>	Convierte el valor decimal leído en la entrada analógica indicada en <i>:CANAL</i> en un valor de tensión.
<b><i>SAX</i></b>	Devuelve directamente el valor en voltios del sensor analógico conectado a la entrada analógica IN1.		

<b><i>SAY</i></b>	Devuelve directamente el valor en voltios del sensor analógico conectado a la entrada analógica IN2.	<b><i>LEER_ENTRADA_ANALOGICA_num</i></b> <b><i>:SELECC</i></b>	Se encarga de escribir y leer del puerto los datos necesarios para obtener el valor de la entrada analógica indicada en <i>:SELECC</i> .
<b><i>SAZ</i></b>	Devuelve directamente el valor en voltios del sensor analógico conectado a la entrada analógica IN3.		
<b><i>SA?</i></b>	Devuelve una lista con el valor en voltios de los cinco sensores analógicos.		

### 3.2 Procedimientos para crear el entorno gráfico:

A continuación se presentan las distintas primitivas disponibles para crear el entorno gráfico para manejar la controladora CNICE con los procedimientos auxiliares necesarios para las mismas:

PRIMITIVA	DESCRIPCIÓN	PROCEDIMIENTOS AUXILIARES	
<i>graficos</i>	Crea las ventanas necesarias para manejar las entradas y salidas analógicas (primitiva <i>graficosan</i> ) y digitales y la ventana para informar de que se debe controlar el estado de la comunicación entre el ordenador y la controladora (primitiva <i>informa</i> ) si no se obtienen los resultados esperados.	Procedimiento	Descripción
		<i>pintasalidas</i>	Actualiza el valor de los indicadores de las salidas digitales del entorno gráfico.
		<i>pintaentradas</i>	Actualiza el valor de los indicadores de las entradas digitales del entorno gráfico.
		<i>ACTIVAR_SALIDA_X (*)</i>  X indica el número de salida que se quiere activar (1, 2, ..., 8).	Activa la salida digital que se le indique en X, dejando el resto de salidas como estén.
		<i>DESACTIVAR_SALIDA_X (*)</i>  X indica el número de salida que se quiere desactivar (1, 2, ..., 8).	Desactiva la salida digital que se le indique en X, dejando el resto de salidas como estén.
		<i>SALIDA_DIGITAL :VALOR</i>	Pone en el puerto de control la palabra necesaria para activar/desactivar las salidas digitales.
<i>graficosan</i>	Crea una ventana para controlar las entradas		

	<i>analógicas.</i>
<b><i>informa</i></b>	Crea una ventana para informar de que se debe controlar el estado de la comunicación entre el ordenador y la controladora (primitiva <i>informa</i> ) si no se obtienen los resultados esperados.