

<b>INTRODUCCIÓN AL CAPÍTULO</b>	<b>3</b>
<b>8. INTRODUCCIÓN A RSLOGIX 500</b>	<b>4</b>
8.1. Descripción general del software.....	4
8.2. Instalación del software.....	6
8.3. Configuración del autómata y las comunicaciones .....	7
8.4. Edición de un programa Ladder.....	8
8.5. Descarga del programa .....	11
8.6. Menú ayuda.....	16
8.7. Requisitos mínimos del sistema .....	16
8.8. Bibliografía del capítulo .....	17





## Introducción al capítulo

En este capítulo se dan las directrices básicas para la utilización del RSLogix 500. Este programa permite crear los programas de control en lenguaje Ladder del autómatas MicroLogix 1500.

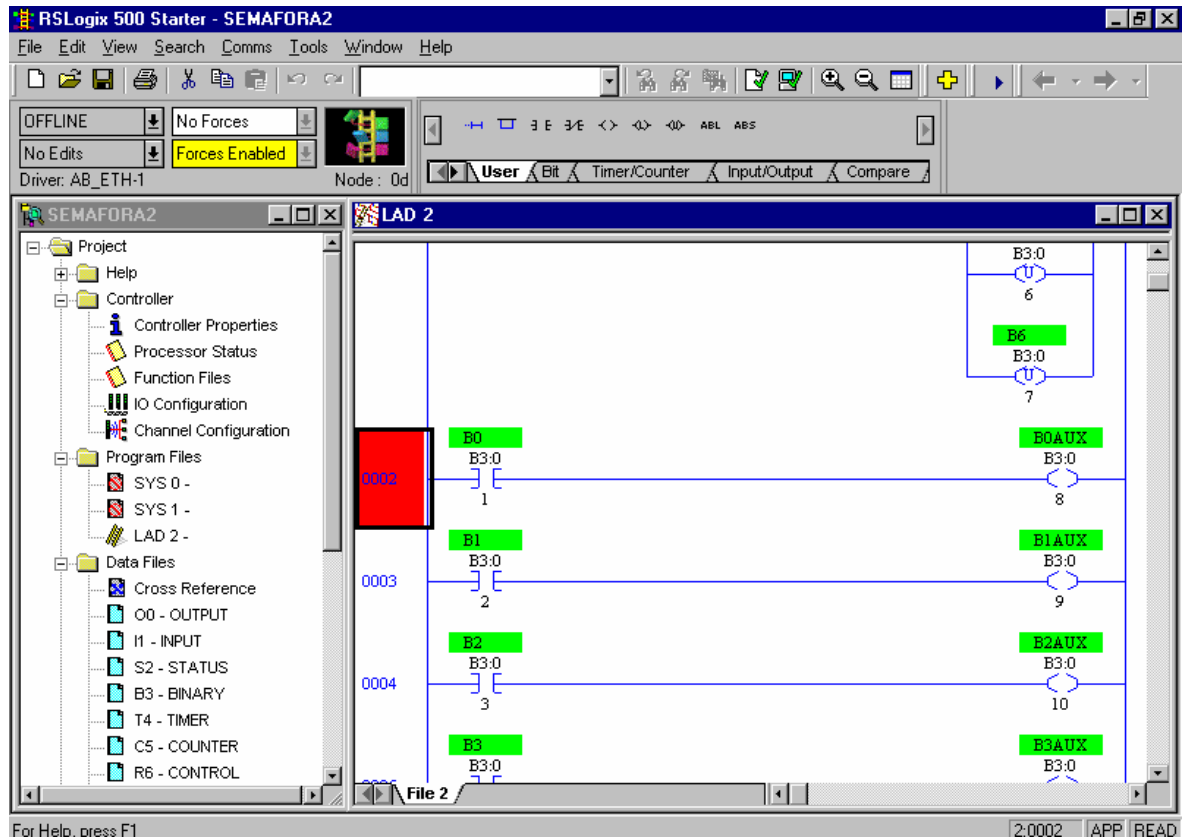


Figura 8.1: Pantalla principal del RSLogix 500



## 8. Introducción a RSLogix 500

### 8.1. Descripción general del software

RSLogix 500 es el software destinado a la creación de los programas del autómatas en lenguaje de esquema de contactos o también llamado lógica de escalera (*Ladder*). Incluye editor de *Ladder* y verificador de proyectos (creación de una lista de errores) entre otras opciones. Este producto se ha desarrollado para funcionar en los sistemas operativos Windows®. [1]

Existen diferentes menús de trabajo (figura 8.2) en el entorno de RSLogix 500, a continuación se hace una pequeña explicación de los mismos:

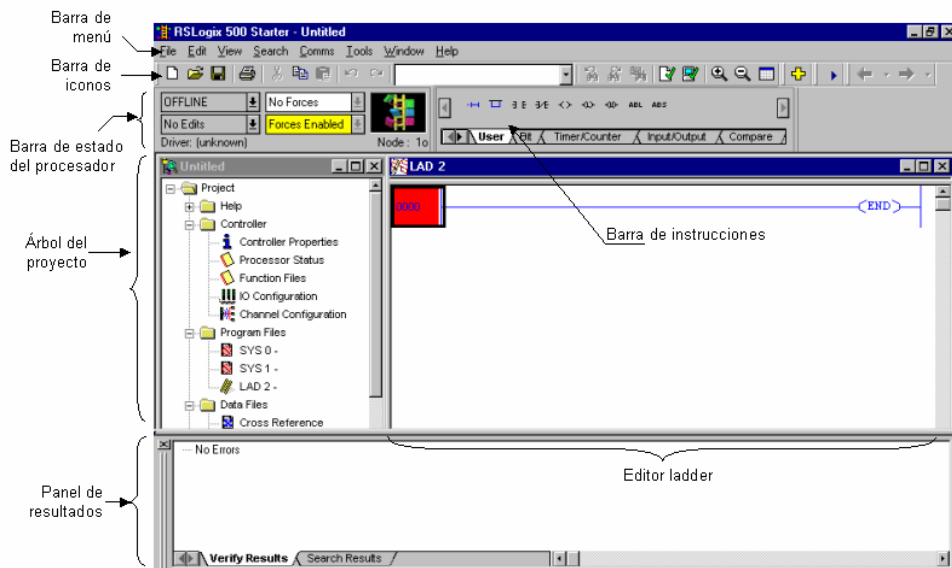


Figura 8.2: Vista principal de RSLogix 500

**Barra de menú:** permite realizar diferentes funciones como recuperar o guardar programas, opciones de ayuda, etc. Es decir, las funciones elementales de cualquier software actual.

**Barra de iconos:** engloba las funciones de uso más repetido en el desarrollo de los programas.

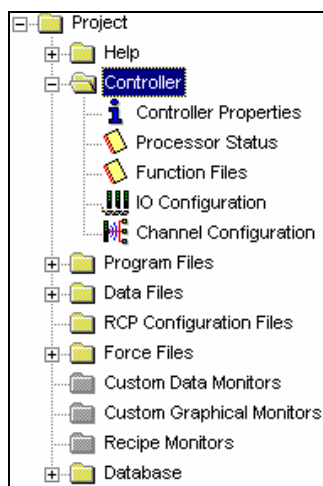
**Barra de estado del procesador:** Nos permite visualizar y modificar el modo de trabajo del procesador (*online*, *offline*, *program*, *remote*), cargar y/o descargar programas (upload/download program), así como visualizar el controlador utilizado (*Ethernet drive* en el caso actual).



Los modos de trabajo más usuales son:

- **Offline:** Consiste en realizar el programa sobre un ordenador, sin necesidad alguna de acceder al PLC para posteriormente una vez acabado y verificado el programa descargarlo en el procesador. Este hecho dota al programador de gran independencia a la hora de realizar el trabajo.
- **Online:** La programación se realiza directamente sobre la memoria del PLC, de manera que cualquier cambio que se realice sobre el programa afectará directamente al procesador, y con ello a la planta que controla. Este método es de gran utilidad para el programador experto y el personal de mantenimiento ya que permite realizar modificaciones en tiempo real y sin necesidad de parar la producción.

**Árbol del proyecto:** Contiene todas las carpetas y archivos generados en el proyecto, estos se organizan en carpetas. [1] Las más interesantes para el tipo de prácticas que se realizará son:

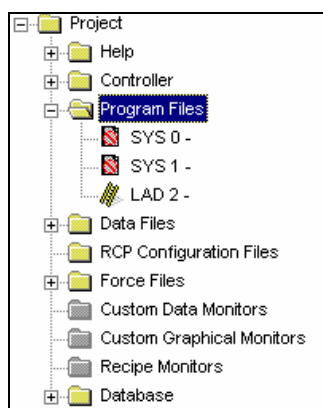


**Controller properties:** contiene las prestaciones del procesador que se está utilizando, las opciones de seguridad que se quieren establecer para el proyecto y las comunicaciones.

**Processor Status:** se accede al archivo de estado del procesador

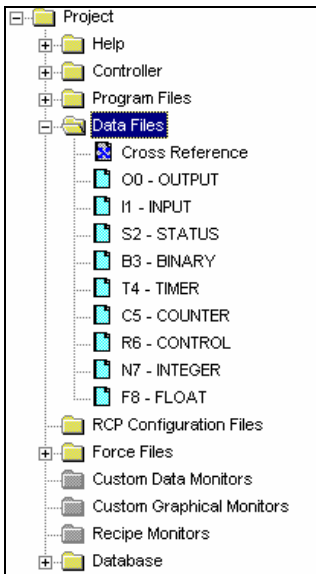
**IO Configuration:** Se podrán establecer y/o leer las tarjetas que conforman el sistema.

**Channel Configuration:** Permite configurar los canales de comunicación del procesador



Contiene las distintas rutinas *Ladder* creadas para el proyecto.





Da acceso a los datos de programa que se van a utilizar así como a las referencias cruzadas (*cross references*). Podemos configurar y consultar salidas (*output*), entradas (*input*), variables binarias (*binary*), temporizadores (*timer*), contadores (*counter*), ...

Si seleccionamos alguna de las opciones se despliegan diálogos similares al siguiente, en el que se pueden configurar diferentes parámetros según el tipo de elemento.

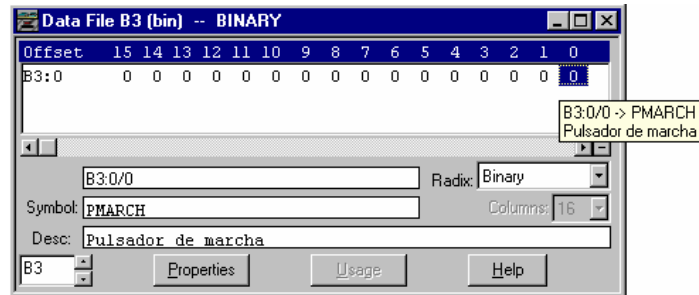



Figura 8.3: Árbol de proyecto

**Panel de resultados:** aparecen los errores de programación que surgen al verificar la corrección del programa realizado  (situados en la barra de iconos). Efectuando doble clic sobre el error, automáticamente el cursor se situará sobre la ventana de programa Ladder en la posición donde se ha producido tal error.

También es posible validar el archivo mediante *Edit > Verify File* o el proyecto completo *Edit > Verify Project*.

**Barra de instrucciones:** Esta barra le permitirá, a través de pestañas y botones, acceder de forma rápida a las instrucciones más habituales del lenguaje Ladder. Presionando sobre cada instrucción, ésta se introducirá en el programa Ladder.

**Ventana del programa Ladder:** Contiene todos los programas y subrutinas Ladder relacionados con el proyecto que se esté realizando. Se puede interaccionar sobre esta ventana escribiendo el programa directamente desde el teclado o ayudándose con el ratón (ya sea arrastrando objetos procedentes de otras ventanas ó seleccionando opciones con el botón derecho del ratón).

## 8.2. Instalación del software

Una vez introducido el CD-ROM de RSLogix 500 el proceso de instalación comenzará automáticamente. Escogemos *Install RSLogix 500* y se siguen las instrucciones, se introduce



el código serie y, cuando se pida, se introduce el disquete llave que viene en el paquete de software. Este activará la aplicación y estará lista para su funcionamiento.

Es recomendable guardar en lugar seguro tanto la llave como el CD de instalación.

### 8.3. Configuración del autómatas y las comunicaciones

Para empezar se ha de configurar el autómatas que se usará, en nuestro caso se trata de un MicroLogix 1500 LSP serie C. Para hacerlo nos dirigimos al menú *File>New* y en el diálogo que aparece seleccionamos el procesador adecuado.

En el mismo diálogo se tiene la posibilidad de seleccionar la red a la que estará conectado. Si hemos efectuado correctamente la configuración de la red anteriormente (con el RSLinx) ya aparecerá el controlador correspondiente, en la esquina inferior izquierda de la figura 8.4 en el desplegable *Driver*. Sino, podemos usar el pulsador que aparece (*Who Active*) que permite acceder a un diálogo similar a *RSWho* y seleccionar la red definida. Seleccionamos el autómatas MicroLogix 1500 que aparece.

Para que aparezca el autómatas en la red se debe estar conectado a Internet y tener activado el RSLinx.

Una vez aceptado (*OK*) aparecerá la ventana del proyecto y la ventana del programa Ladder.

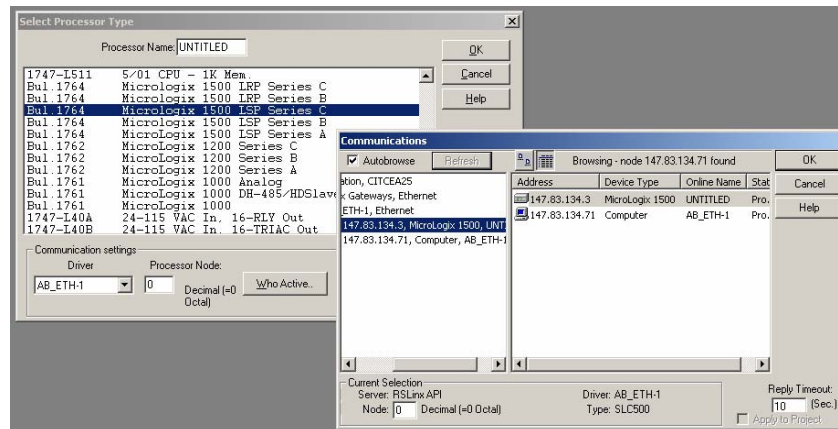


Figura 8.4: Selección del procesador

La configuración de la red se puede modificar en cualquier momento accediendo desde el árbol de proyecto > *Controller>Controller Communications*.



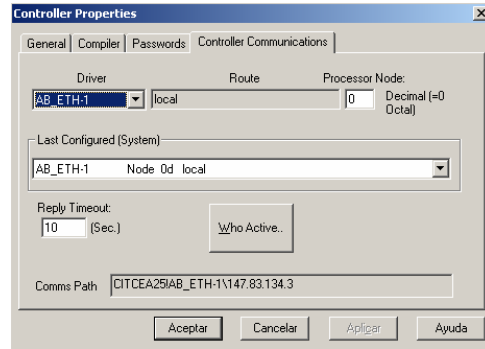


Figura 8.5: Configuración de las comunicaciones del autómat

### 8.4. Edición de un programa Ladder

Las diferentes instrucciones del lenguaje Ladder se encuentran en la barra de instrucciones citada anteriormente (figura 8.1). Al presionar sobre alguno de los elementos de esta barra estos se introducirán directamente en la rama sobre la que nos encontremos.

A continuación se hará una explicación de las instrucciones usadas para la resolución de las prácticas de este curso [2]:



Añadir una nueva rama al programa



Crear una rama en paralelo a la que ya está creada



**Contacto normalmente abierto (XIC - *Examine If Closed*):** examina si la variable binaria está activa (valor=1), y si lo está permite al paso de la señal al siguiente elemento de la rama. La variable binaria puede ser tanto una variable interna de memoria, una entrada binaria, una salida binaria, la variable de un temporizador,...

En este ejemplo si la variable B3:0/0 es igual a 1 se activará la salida O:0/0.



**Contacto normalmente cerrado (XIO - *Examine If Open*):** examina si la variable binaria está inactiva (valor=0), y si lo está permite al paso de la señal al siguiente elemento de la rama.





En este ejemplo si la variable B3:0/0 es igual a 0 se activará la salida O:0/0.



**Activación de la variable (OTE - *Output Energize*):** si las condiciones previas de la rama son ciertas, se activa la variable. Si dejan de ser ciertas las condiciones o en una rama posterior se vuelve a utilizar la instrucción y la condición es falsa, la variable se desactiva.

Para ciertos casos es más seguro utilizar las dos instrucciones siguientes, que son instrucciones retentivas.

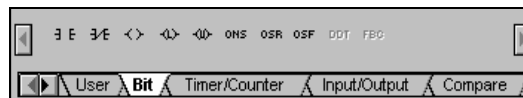


**Activación de la variable de manera retentiva (OTL - *Output Latch*):** si las condiciones previas de la rama son ciertas, se activa la variable y continúa activada aunque las condiciones dejen de ser ciertas.

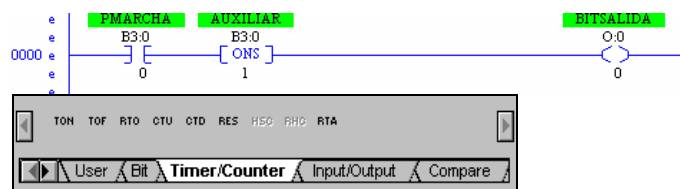
Una vez establecida esta instrucción solo se desactivará la variable usando la instrucción complementaria que aparece a continuación.



**Desactivación de la variable (OTU - *Output Unlatch*):** normalmente esta instrucción se utiliza para anular el efecto de la anterior. Si las condiciones previas de la rama son ciertas, se desactiva la variable y continúa desactivada aunque las condiciones dejen de ser ciertas.



**Flanco ascendente (ONS - *One Shot*):** esta instrucción combinada con el contacto normalmente abierto hace que se active la variable de salida únicamente cuando la variable del contacto haga la transición de 0 a 1 (flanco ascendente). De esta manera se puede simular el comportamiento de un pulsador.

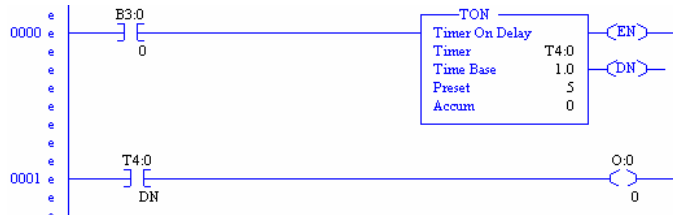


**Temporizador (TON - *Timer On-Delay*):** La instrucción sirve para retardar una salida, empieza a contar intervalos de tiempo cuando las condiciones del renglón se hacen verdaderas. Siempre que las condiciones del renglón permanezcan verdaderas, el temporizador incrementa su acumulador



hasta llegar al valor preseleccionado. El acumulador se restablece (0) cuando las condiciones del renglón se hacen falsas.

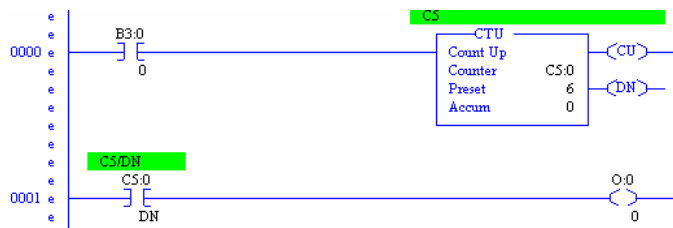
Es decir, una vez el contacto (B3:0/0) se activa el temporizador empieza a contar el valor seleccionado (Preset=5) en la base de tiempo especificada (1.0 s.). La base de tiempo puede ser de 0.001 s., 0.01 s. y 1.00 s. Una vez el valor acumulado se iguala al preseleccionado se activa el bit llamado T4:0/DN (temporizador efectuado). Este lo podemos utilizar como condición en la rama siguiente.



**CTU**

**Contador (CTU - Count Up):** se usa para incrementar un contador en cada transición de renglón de falso a verdadero.

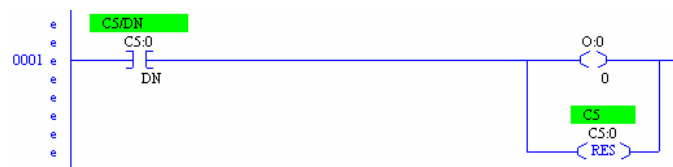
Por ejemplo, esta instrucción cuenta todas las transiciones de 0 a 1 de las variable colocada en el contacto normalmente abierto. Cuando ese número se iguale al preseleccionado (6 en este caso) el bit C5:0/DN se activa. Este bit se puede usar posteriormente como condición en otro renglón del programa.



**RES**

**Resetear (RES - Reset):** La instrucción RES restablece temporizadores, contadores y elementos de control.

En el ejemplo presentado a continuación una vez aplicado el reset, el contador se pone a cero y cuando la condición del renglón del contador vuelca a ser cierta, empezará a contar de cero.



Para introducir el nombre de las variables se puede hacer mediante el teclado o a partir del Árbol del proyecto >Data Files y seleccionar el elemento necesario (salida, entrada,



variable..). Una vez seleccionado el elemento se abre una ventana y se puede arrastrar con el ratón la variable como se muestra en la figura 8.5, y colocar el nombre de la variable (0:0.0/0) encima de la casilla verde de la instrucción (indicada con el círculo azul).

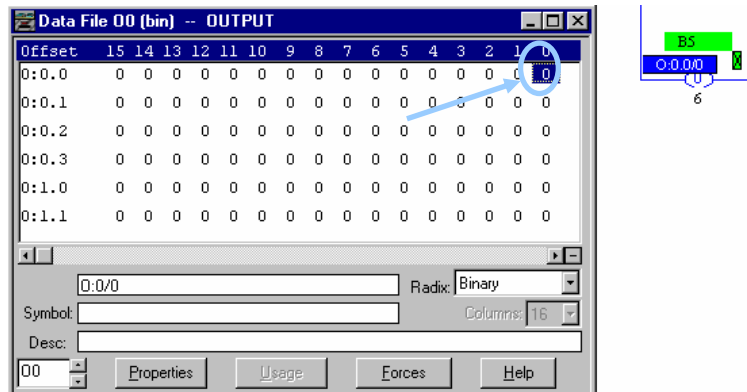


Figura 8.6: Adición de variables

Para más información sobre las instrucciones usadas en el RSLogix 500 se puede acceder al menú de ayuda: *Help>SLC Instruction Help* y se encuentra un explicación muy detallada de su funcionamiento.

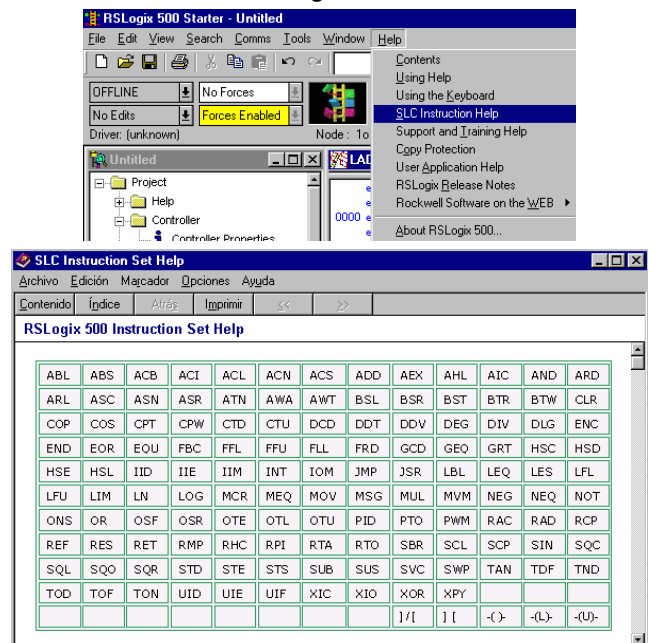


Figura 8.7: Ayuda de las instrucciones

## 8.5. Descarga del programa

Una vez se ha realizado el programa y se ha verificado que no exista ningún error se procede a descargar el programa al procesador del autómatas (*download*).



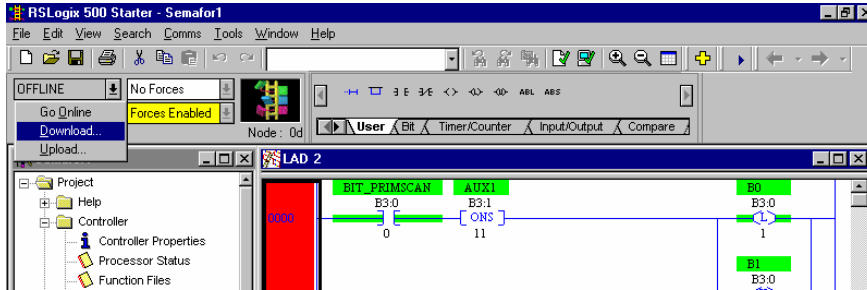


Figura 8.8: Descarga del programa al autómat

A continuación aparece diversas ventanas de diálogo que se deben ir aceptando sucesivamente:

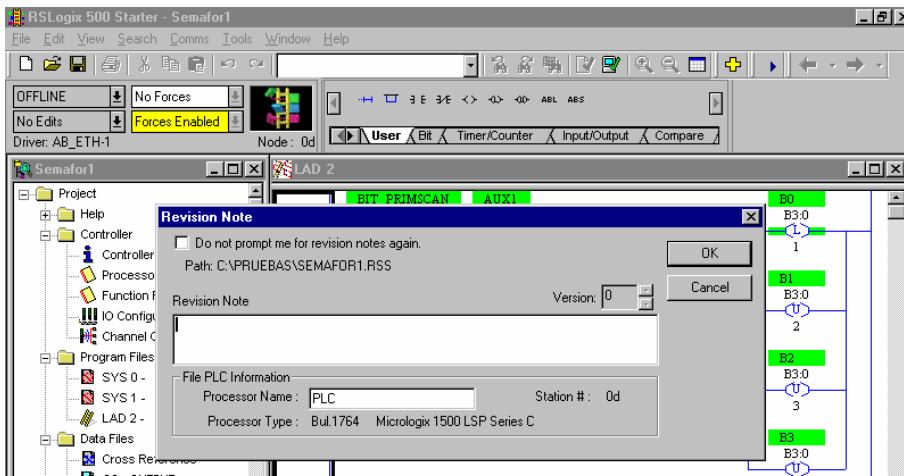


Figura 8.9: Salvar el programa

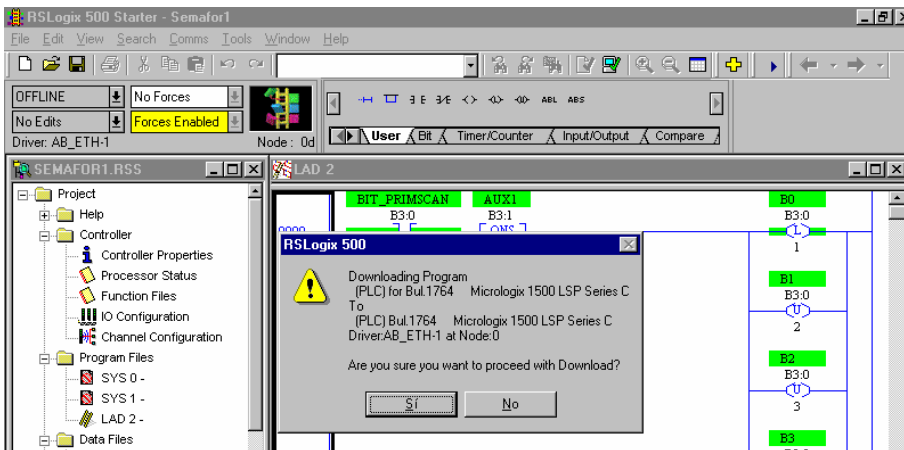


Figura 8.10: Aceptación de la descarga



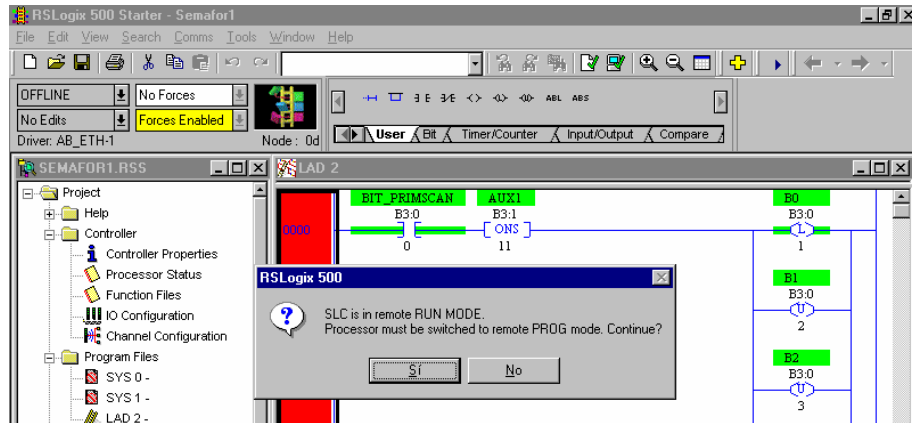


Figura 8.11: Paso a modo *Remote Program*

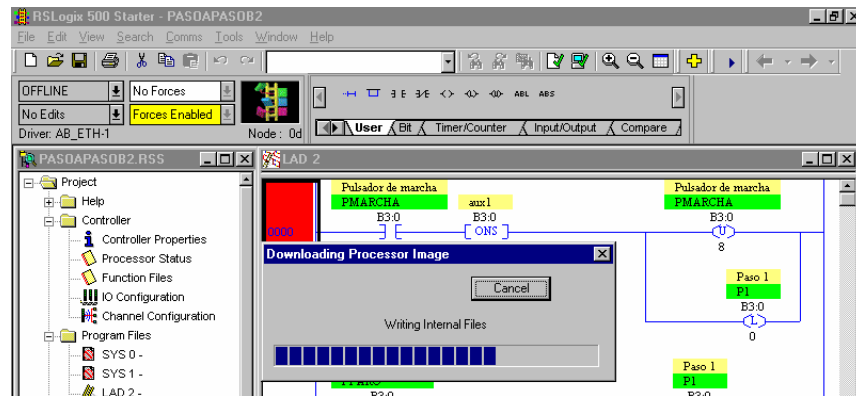


Figura 8.12: Transfiriendo datos del programa

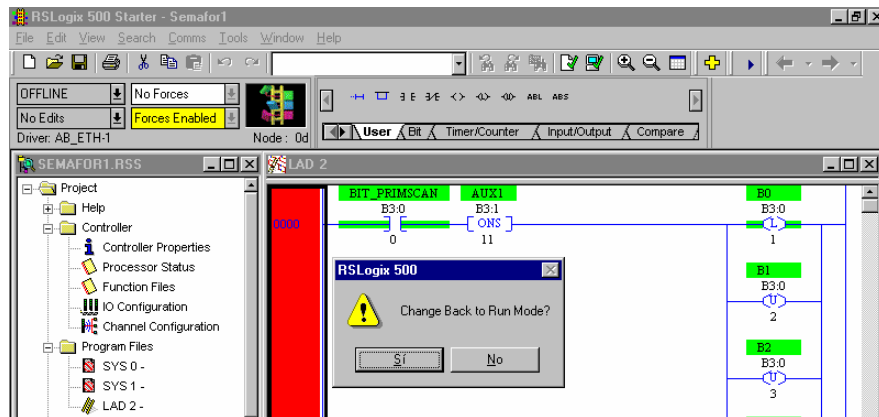


Figura 8.13: Paso a modo *Run* (el programa está en funcionamiento)



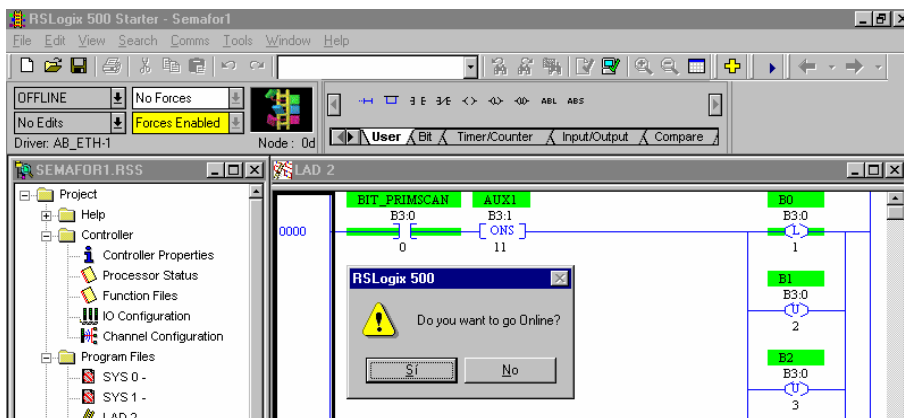


Figura 8.14: Paso a modo on-line (conectado)

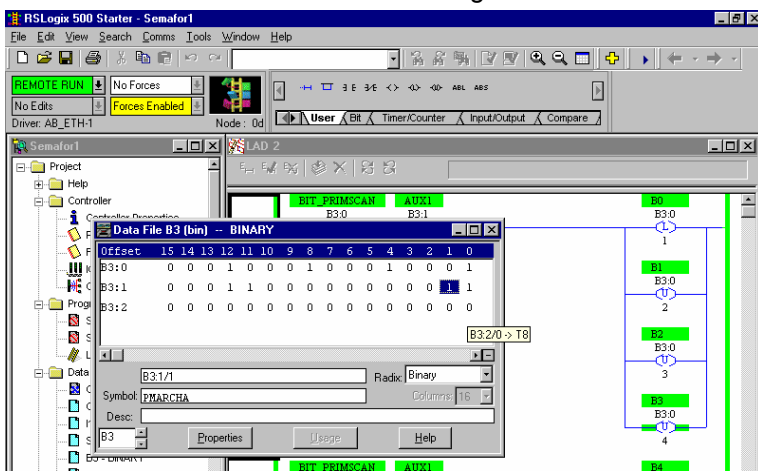


Figura 8.15: Programa on-line y forzado de entrada

Para desconectar el enlace entre el ordenador personal y el autómata se deben seguir los siguientes pasos, siempre teniendo en cuenta que una vez desconectado el autómata este sigue funcionando con el programa descargado. Es importante dejar el programa en un estado segura (pulsador de paro).

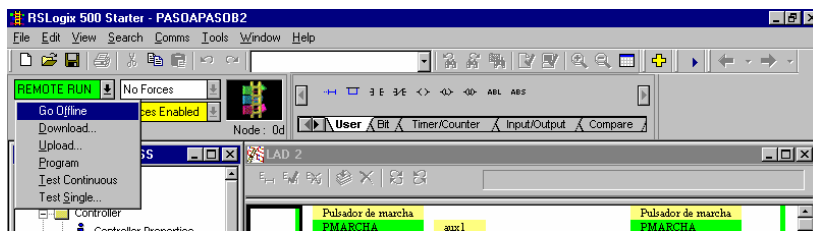


Figura 8.16: Paso a modo off-line (desconectado)

A continuación aparece un diálogo para salvar el programa realizado, de esta manera se puede salvar todos los archivos de datos (tablas de variables, salidas, temporizadores,...).



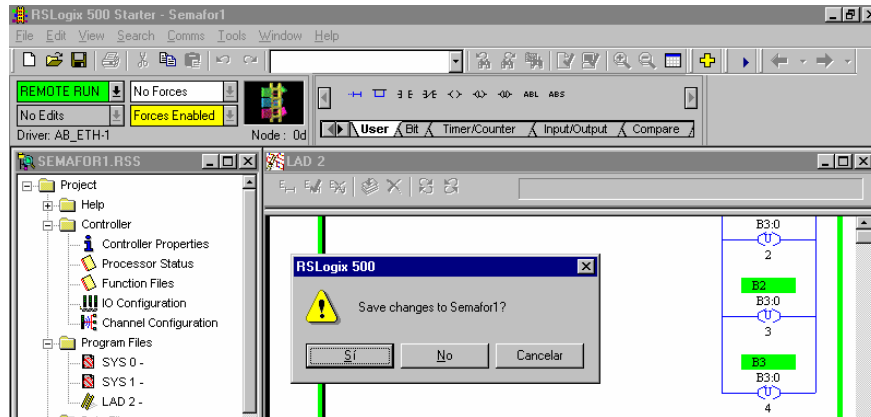


Figura 8.17: Salvar los resultados

Pueden surgir algunos problemas durante la descarga del programa, el más común es que existan problemas con la conexión a Internet. Entonces al descargar el programa surgirá un diálogo en el que se muestra que el camino de la conexión no está funcionando. (figura 8.18). Para solucionar el problema se debe comprobar si la configuración del *drive* en el RSLinx es correcta y si la conexión a Internet del usuario está funcionando de manera normal (figura 8.19).

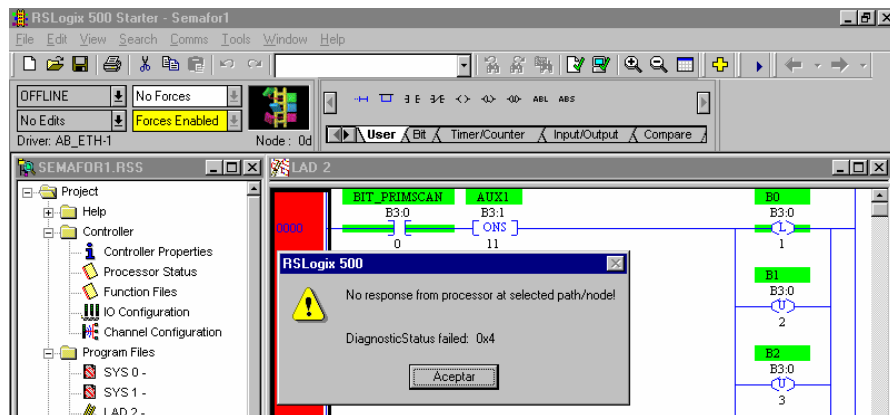


Figura 8.18: Conexión sin funcionar

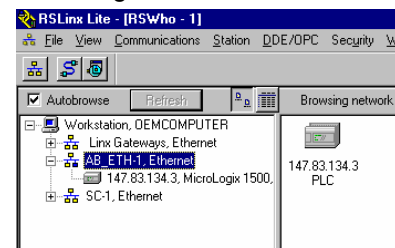
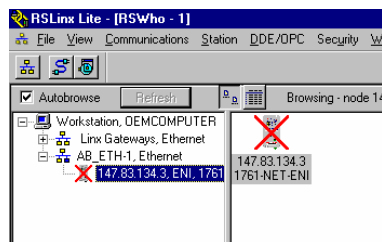


Figura 8.19: Buena y mala conexión a Ethernet



### 8.6. Menú ayuda

Para cualquier duda que se pueda presentar en el uso del programa, se puede utilizar la ayuda que es bastante completa. Esta permite buscar según palabras clave o por agrupaciones de contenido.

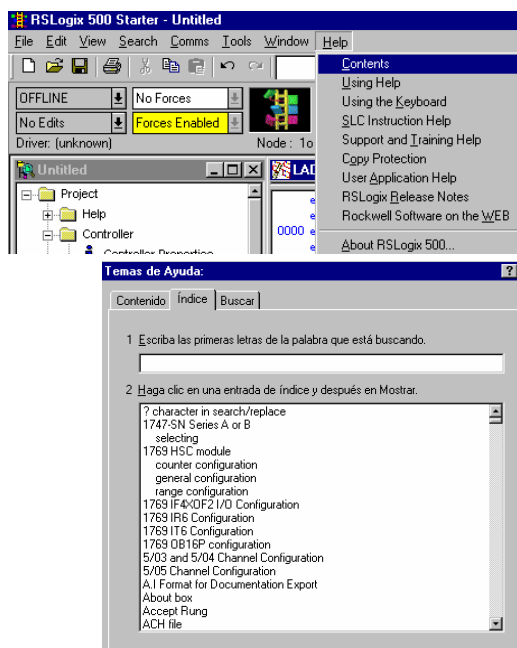


Figura 8.20: Ayuda del RSLogix 500

### 8.7. Requisitos mínimos del sistema

Para poder utilizar este software sin problemas se requiere tener un sistema con las siguientes características como mínimo:[1]

- Intel Pentium II® o superior
- 128 MB de RAM para Windows NT, Windows 2000, o Windows XP (64 MB para Windows 98®)
- 45 MB de espacio de disco duro disponible
- Monitor y adaptador gráfico SVGA 256-color con resolución 800x600
- CD-ROM drive
- Disquetera de 3.5 pulgadas (solo para la activación del programa mediante la llave)
- Cualquier dispositivo de señalamiento compatible con Windows
- RSLinx™ (software de comunicación) versión 2.31.00 o posterior.





## 8.8. Bibliografía del capítulo

- [1] RSLogix 500, Programming for the SLC 500 and MicroLogix Families, Getting Results Guide, Doc ID LG500-GR001A-EN-P [<http://www.ab.com/micrologix>, 27 de octubre de 2004]
- [2] Controladores Programables MicroLogix 1200 y MicroLogix 1500, Manual de referencia del conjunto de instrucciones, Publicación 1762-RM001D-ES-P <http://www.ab.com/micrologix>:

