

# Guia básica de uso

## Automation Studio 3.x



Perfection in Automation  
[www.br-automation.com](http://www.br-automation.com)



## I Listado de Contenidos

1. CONTROL	4
1.1 Creación del proyecto en Automation Studio (Hardware automático)	4
1.2 Configuración de adaptador de Ethernet	6
1.3 Creación del proyecto (Hardware manual)	9
1.4 Creación de la Compact Flash	12
1.5 Añadir tareas - Automation Runtime Multitasking	14
1.6 Definición de variables y constantes	18
1.7 Definición de tipos	22
1.8 Mapeado y configuración de E/S	24
1.9 Insertar funciones de librerías	26
1.10 Herramientas de Debug	28
1.11 Cambio de sistema operativo	32
1.12 Trabajar con distintas configuraciones	33
1.13 Simulación del Runtime - AR000	36
1.14 La ayuda de Automation Studio	38

2. VISUALIZACIÓN	40
2.1 Añadir una visualización	40
2.2 Añadir un servidor VNC	42
2.3 Funcionamiento básico de visual components	44
2.4 Interacción con variables de programa	45
2.5 Insertar Textos, idiomas, fuentes	48
2.6 Grupos de texto, textos dinámicos	50
2.7 Cambio del lenguaje y página de inicio o por programa	52
2.8 Bitmaps	53
2.9 Botones	55
2.10 Estilos	58
2.11 Capas	58
2.12 StatusDatapoint, mostrar o esconder objetos	60
2.13 Passwords	61
2.14 Fecha y hora	63
2.15 Gestión de alarmas	65
2.16 Uso de unidades	67
2.17 Gráficas de tendencias o Trends	70
3. GLOSARIO	73

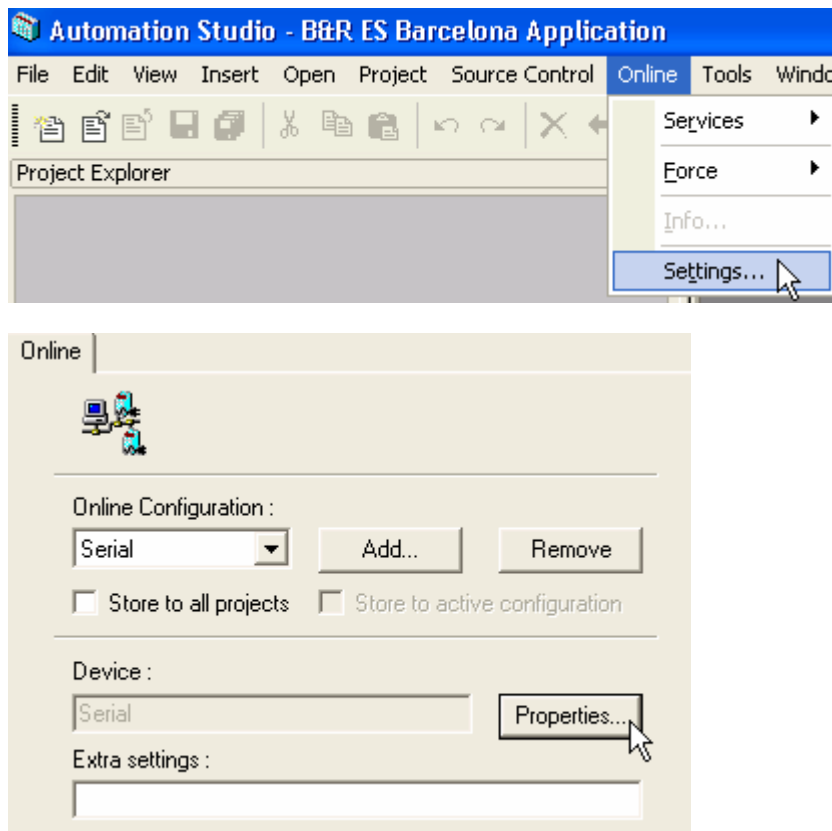
B&R se reserva el derecho a cambiar los contenidos de este manual sin previo aviso. Bernecker + Rainer Automatización industrial S.L.U. no da garantía, expresada o implícita, referente a la presente documentación. Bernecker + Rainer Automatización industrial S.L.U. no se hace responsable en el caso de accidentes o daños producidos por el uso de estos productos. Los nombres de software, hardware y marcas registradas citadas en este documento están registradas por las respectivas compañías.

## 1. CONTROL

### 1.1 Creación del proyecto en Automation Studio (Hardware automático)

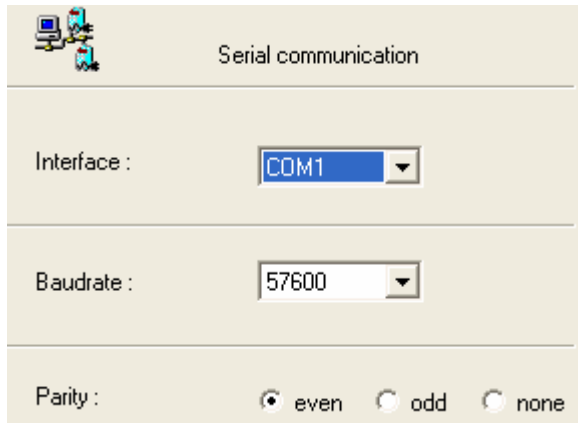
Antes de crear el proyecto se configura la conexión para que al crear el nuevo proyecto se pueda descargar automáticamente el hardware del sistema.

Para configurar el puerto serie, hay que ir a "Online - Settings":



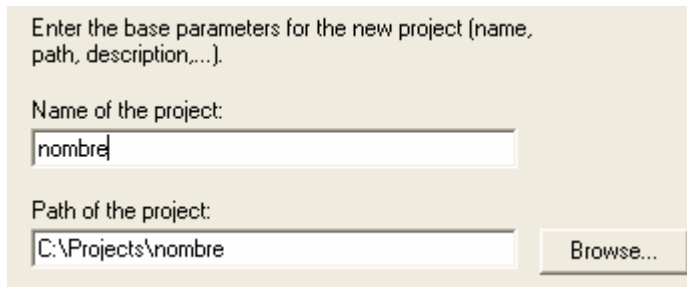
Aquí se selecciona conexión tipo serie y se entra a la ventana de "Properties".

Una vez en las propiedades del puerto serie, hay que seleccionar cual es el puerto que se ha conectado y se comprueba que la configuración sea 57600 bps, paridad "even".



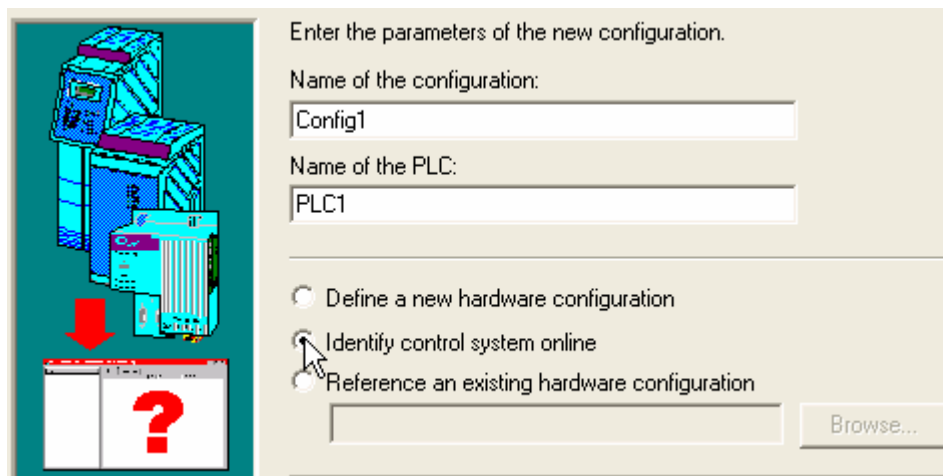
Ahora se crea el nuevo proyecto en el menú file - New Project...

En el dialogo que aparece en la pantalla se introduce un nombre para el proyecto, la ruta y una descripción si se quiere.



En el siguiente diálogo puede introducir un nombre para la configuración de hardware (más adelante se podrán añadir distintas configuraciones) y se selecciona "Upload hardware from target" para que detecte el hardware automáticamente.

Si el sistema está bien configurado se creará un nuevo proyecto de AS (Automation Studio) con nuestro hardware en el árbol de hardware.



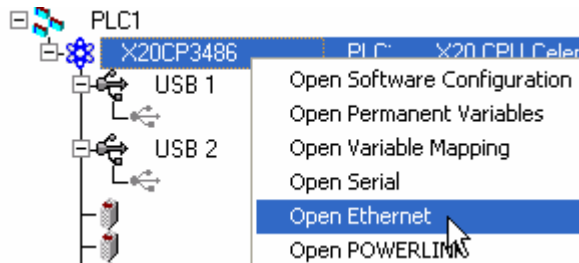
## 1.2 Configuración de adaptador de Ethernet

Una vez ya se haya conectado on-line y se tiene el proyecto con el hardware, se configura la conexión ethernet, que será la más usada para acceder a los autómatas o pantallas.

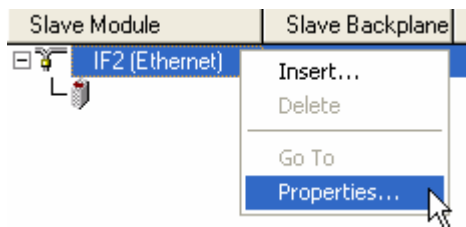
Primero se configura la IP que se va a asignar al autómata. En las pestañas de la parte izquierda hay que seleccionar la de "Physical View" (es el sitio donde se configura el hardware del proyecto).



Una vez en el árbol de hardware se pulsa botón derecho sobre la CPU escogiendo "Open Ethernet".



En la ventana de ethernet de la CPU se pulsa botón derecho en el IF de ethernet y "Properties".



Primero hay que activar el interface Ethernet cambiando el desplegable de "Activate interface" a ON.



Allí se introduce la dirección IP deseada para el autómata y su máscara de subred (normalmente 255.255.255.0).

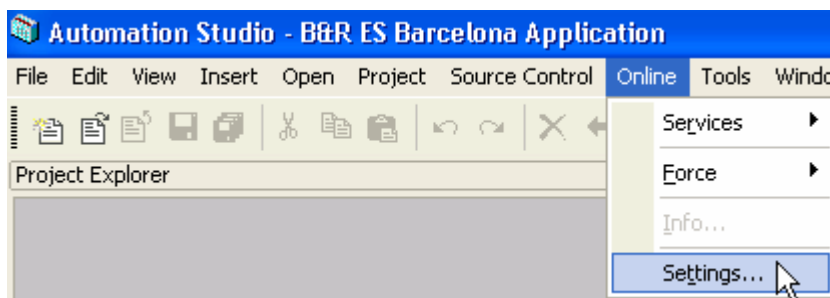
IF2	
Activate interface	on
Device parameters	
Baud rate	auto
Mode	enter IP address manually
Internet address	192.168.0.10
Subnet Mask	255.255.255.0
INA parameters	
Activate online communication	on
Port number	11159
Modbus parameters	
Activate Modbus communication	off

La IP configurada en el ordenador debe ser distinta de la del autómatas y debe estar en el mismo rango: si la dirección IP del autómatas es la 192.168.0.10, el PC debe estar en cualquier dirección 192.168.0.X.

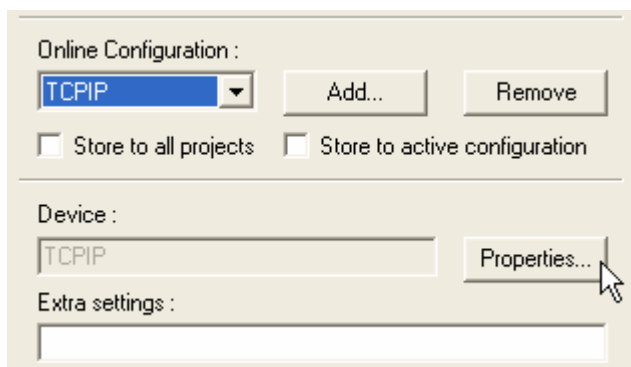
Ahora se compila y transfiere el programa para que el autómatas tenga la IP configurada en su sistema operativo. Para esa operación AS pedirá hacer un "Warmrestart" (reinicio) del autómatas.

Una vez reiniciado el autómatas se cambia la conexión on-line del puerto serie a la red ethernet.

En "Online - Settings..."

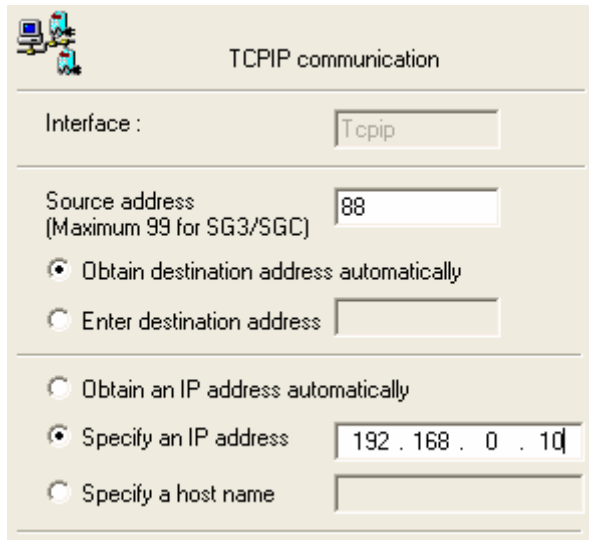


Ahora se selecciona conexión TCP y se accede a las propiedades de la conexión.



En "Properties", se introduce la dirección IP del autómatas configurado antes y se verifica que la casilla de obtener dirección de destino automática esté marcada.

También hay que poner en "Source address" un valor cualquiera que no sea 1.



TCPIP communication

Interface :

Source address  
(Maximum 99 for SG3/SGC)

Obtain destination address automatically

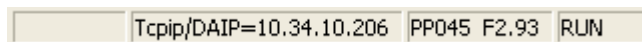
Enter destination address

Obtain an IP address automatically

Specify an IP address

Specify a host name

Ahora al pulsar OK, el ordenador intentará conectar al autómata. Si todo está correcto, aparecerá "RUN" en la barra de estado.

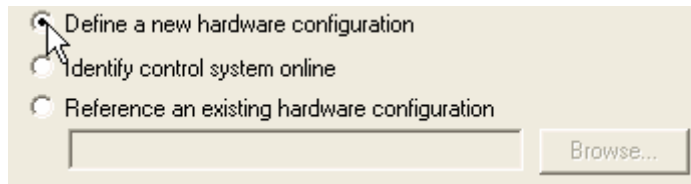




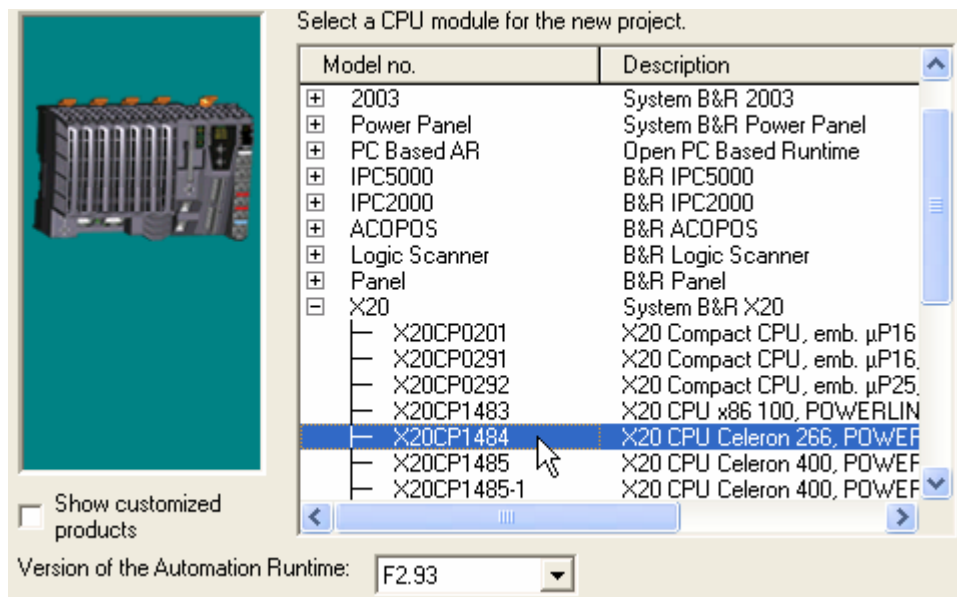
### 1.3 Creación del proyecto (Hardware manual)

Automation Studio permite crear un proyecto sin tener el hardware conectado y generar una tarjeta Compact Flash que contenga el sistema operativo y el programa para un autómeta.

Primero se crea un proyecto sin cargar el hardware del autómeta, escogiendo "Define a new hardware configuration".



Si no se carga el hardware automáticamente AS (Automation Studio) pregunta por el modelo de CPU. En esta caso una CPU X20CP1484.

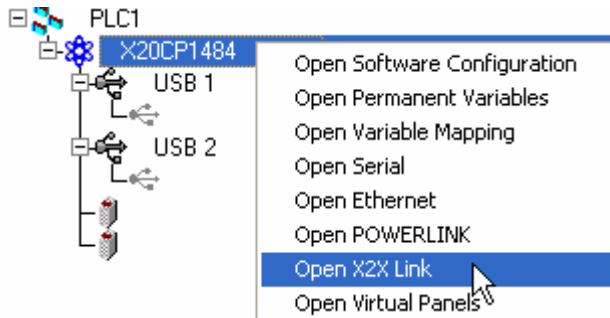


Ahora el proyecto consiste sólo en la CPU, donde hay que añadir las tarjetas de E/S conectadas al interface X2X (el interface X2X es el "Backplane" del autómeta, que es donde se van añadiendo los módulos de entradas o salidas. Aunque sea un Backplane, con el X2X se pueden poner islas de entradas o salidas separadas 100m entre sí).

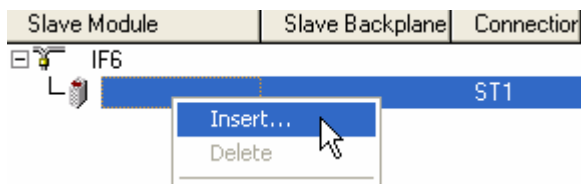
Para añadir las tarjetas o elementos de hardware hay que ir a la pestaña de "Physical View" (es el sitio donde se configura el hardware del proyecto).



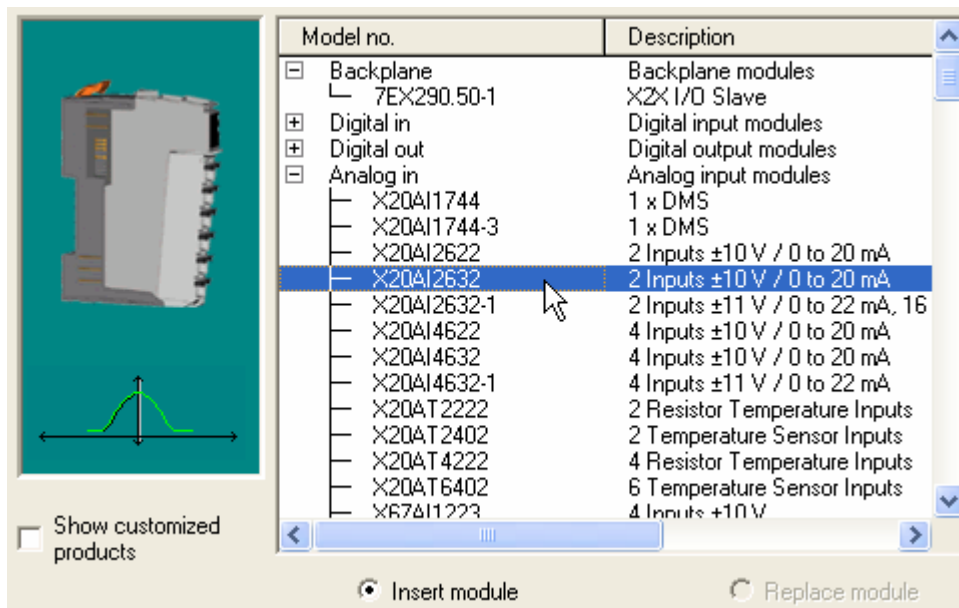
Con el botón derecho en la CPU se escoge "Open X2X Link".



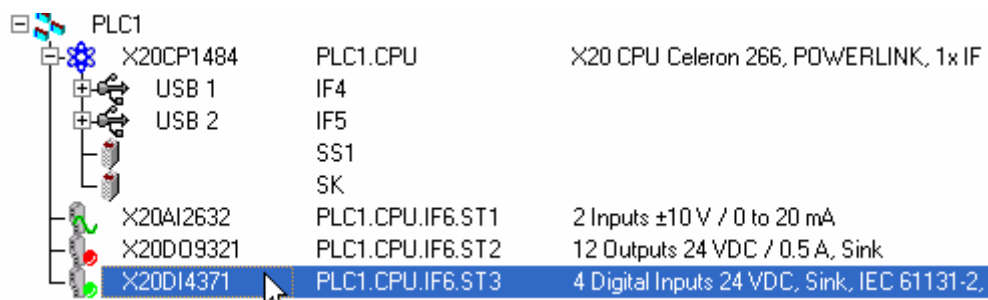
Una vez en la ventana X2X hay que pulsar botón derecho - “insert...”



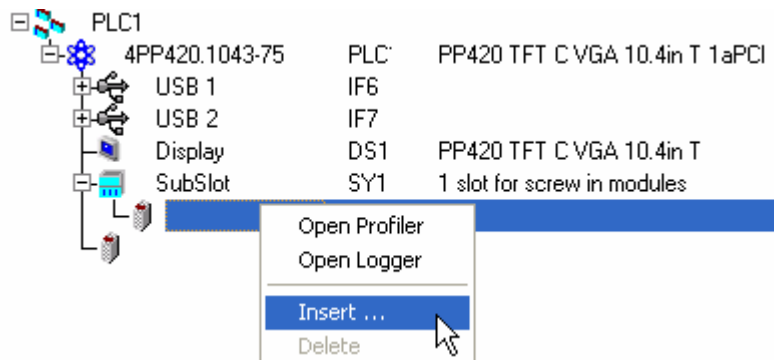
Y se seleccionan los módulos que se quieran añadir.



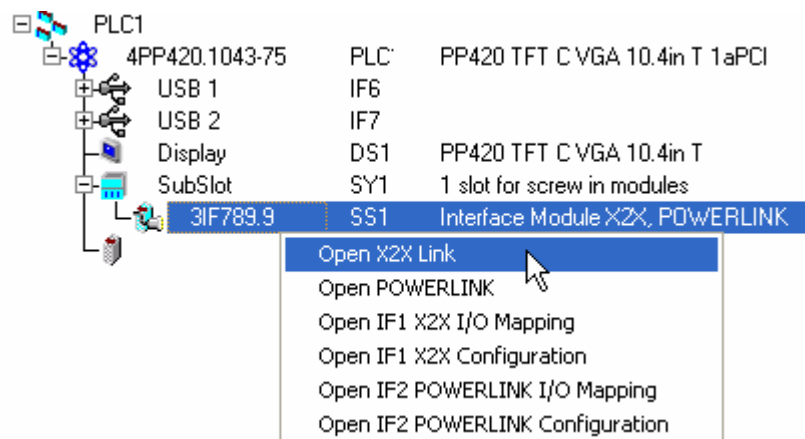
Las tarjetas de E/S configuradas irán apareciendo en el árbol de hardware de la izquierda del AS (Automation Studio).



Si se dispone de un hardware que no lleva el bus X2X integrado (como en el caso de un Power Panel), debe insertarse primero la tarjeta X2X en el "SubSlot".



Pulsando el botón derecho sobre la tarjeta del "SubSlot" se puede abrir la ventana del bus X2X, donde se pueden añadir los módulos de entradas y salidas.



Después de configurar el hardware hay que configurar la IP que se le configurará el autómatas, igual que en el capítulo anterior.

Con todo el hardware y la IP definidos, ya se puede crear una Compact Flash, introducirla en el autómatas y conectarse al autómatas en modo "RUN" si todo ha salido bien.

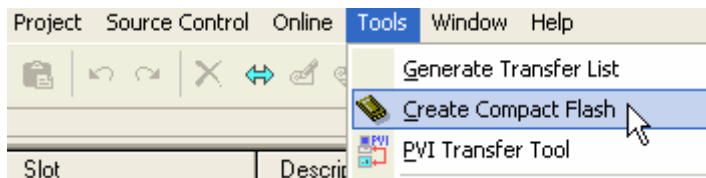
La creación de la Compact Flash se describe en el siguiente capítulo.

## 1.4 Creación de la Compact Flash

Para grabar la Compact Flash se necesita un grabador de Compact Flash: ya sea USB externo o tarjeta PCMCIA en caso de un ordenador portátil.

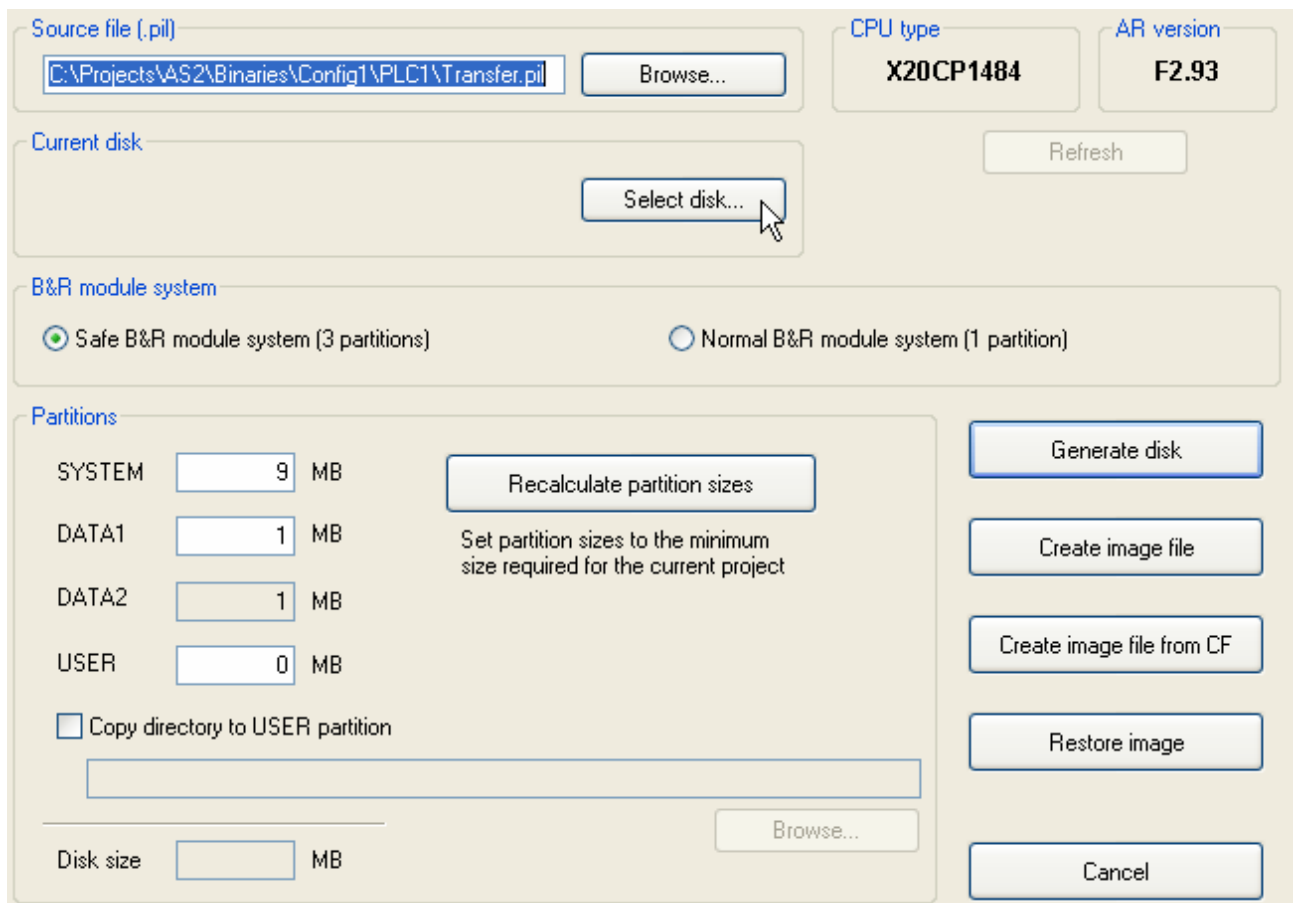
Para empezar se introduce la Compact Flash en el ordenador.

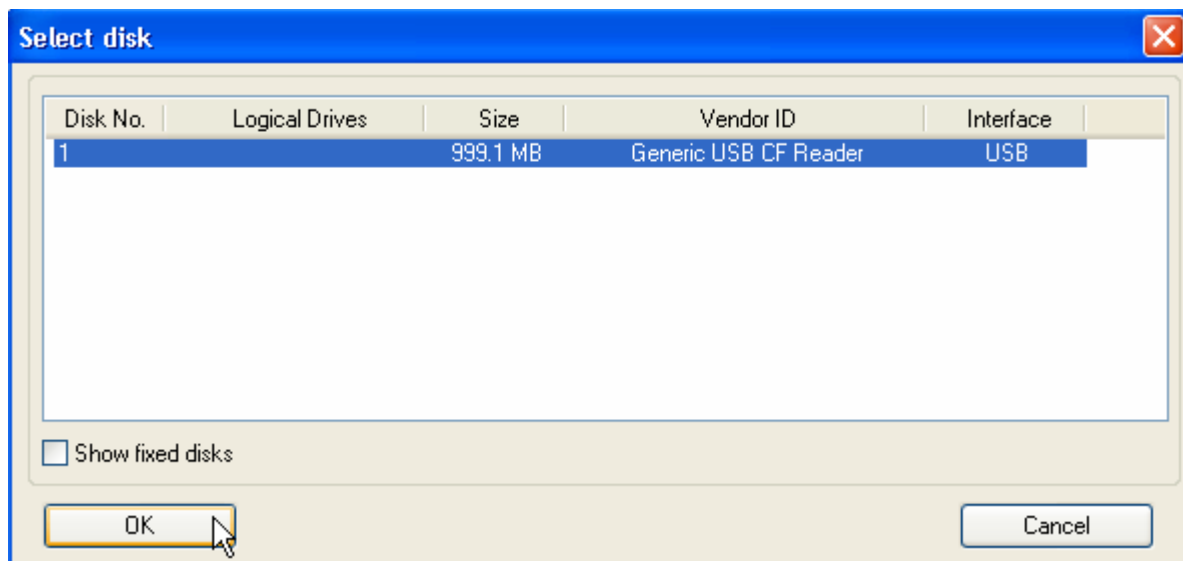
Hay que seleccionar "Tools - Create Compact Flash".



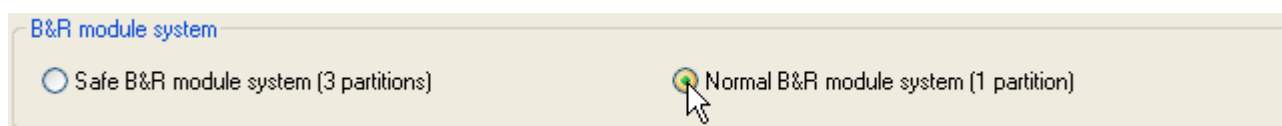
En este momento Automation Studio compilará el proyecto y preparará los ficheros que serán transmitidos a la Compact Flash.

En el dialogo de "Generate Compact Flash" se empieza pulsando en "select disc" para seleccionar el disco que se quiera grabar.





Después de seleccionar el disco, se escoge la opción de "Normal B&R module system" y se puede pulsar el botón de "Generate disk".



Al generar el disco avisa que todos los datos del disco serán borrados, a lo cual hay que responder si.



Una vez generada toda la Compact Flash, se puede insertar al autómata y conectarse al autómata en modo "RUN" si todo ha salido bien.

## 1.5 Añadir tareas - Automation Runtime Multitasking

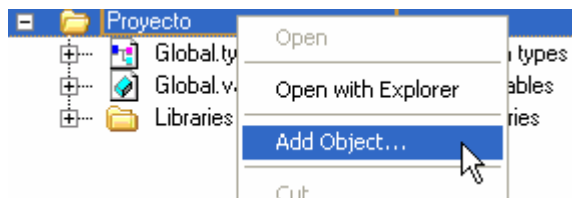
El Automation Runtime (el sistema operativo que se ejecuta en los PLCs de B&R), es un sistema determinístico y multitasking. Eso significa que cada tarea de software tiene asignado un tiempo de ciclo, o sea que cada vez que se cumpla el tiempo de ciclo la tarea debe ser ejecutada.

El Automation Runtime maneja todas las tareas y las organiza en el tiempo para que cada tarea cumpla con su tiempo de ciclo. Si no se pueden cumplir los ciclos de tiempo, el sistema se irá a "Service Mode".

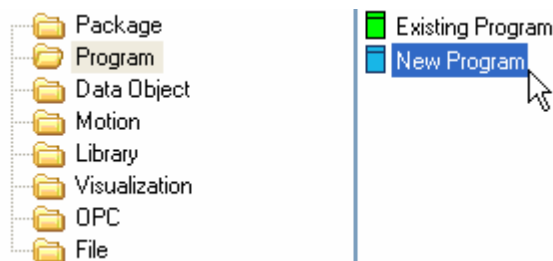
Para insertar una tarea hay que pasar a la pestaña de "Logical View" (que es donde se crean todos los elementos de software).



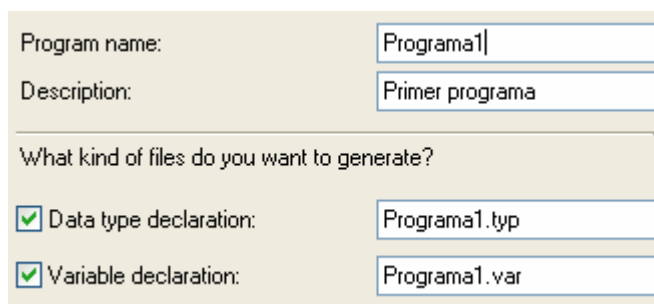
En el árbol de software hay que pulsar botón derecho - "Add Object".



Cuando se inserta un objeto, AS pregunta que tipo de objeto se quiere. En el caso de una tarea de PLC hay que seleccionar "Program - New Program".

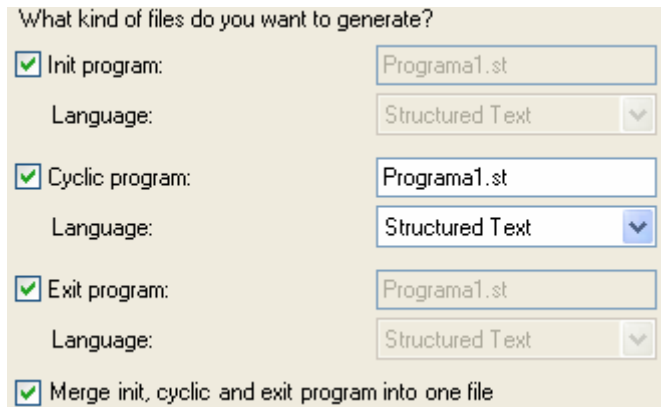


Para la nueva tarea hay que ponerle el nombre y dejar las opciones de abajo marcadas así AS creará una declaración de variables y de tipos.

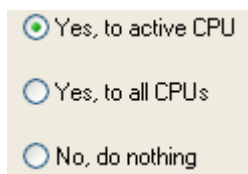


Ahora viene el momento de seleccionar el lenguaje de programación para esta tarea. Automation Studio puede ser programado en 8 lenguajes distintos que se seleccionan en la siguiente pantalla.

También se pueden marcar las opciones de tener parte de programa de Inicio o de Salida, que se juntaran en un mismo fichero si se selecciona la opción de "Merge init, cyclic and exit program into one file".

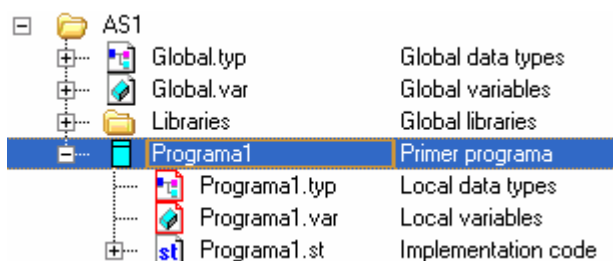


La tarea se puede asignar automáticamente a la configuración actual si se pulsa en la opción "to active CPU" o "to all CPUs". (Las tareas de software son comunes para todas las configuraciones, se pueden asignar o no a cada configuración. Más información en el capítulo de Configuraciones)



La tarea Programa1 se ha creado en el árbol de software, apareciendo el archivo Programa1.st que contiene el código, el Programa1.typ que contiene los tipos de datos locales para esa tarea y el Programa1.var que contiene las variables locales para la tarea.

Para los tipos y variables globales tenemos los archivos Global.typ y Global.var.



Cada tarea se divide en una parte de inicio, una parte cíclica y una parte de salida. La parte de inicio sólo se ejecuta una vez al arrancar el autómata, mientras que la parte cíclica se va ejecutando según el tiempo de ciclo que le hemos asignado y la de salida se ejecuta solo cuando se para la CPU.

Al abrir la tarea con doble clic en el archivo Programa1.st se puede ver como queda separada en las tres partes comentadas antes. El código tiene que escribirse entre PROGRAM \_CYCLIC (por ejemplo) y END\_PROGRAM.

```

PROGRAM _INIT
(* TODO : Add your code here *)
END_PROGRAM
|
PROGRAM _CYCLIC
(* TODO : Add your code here *)
END_PROGRAM
|
PROGRAM _EXIT
(* TODO : Add your code here *)
END_PROGRAM

```

Todas las tareas que se vayan generando en el árbol de software pueden distribuirse en los distintos tiempos de ciclo según la configuración de hardware. La organización de las tareas en ciclos se encuentra en la pestaña de "Physical View",



Botón derecho sobre la CPU - "Open Software configuration".



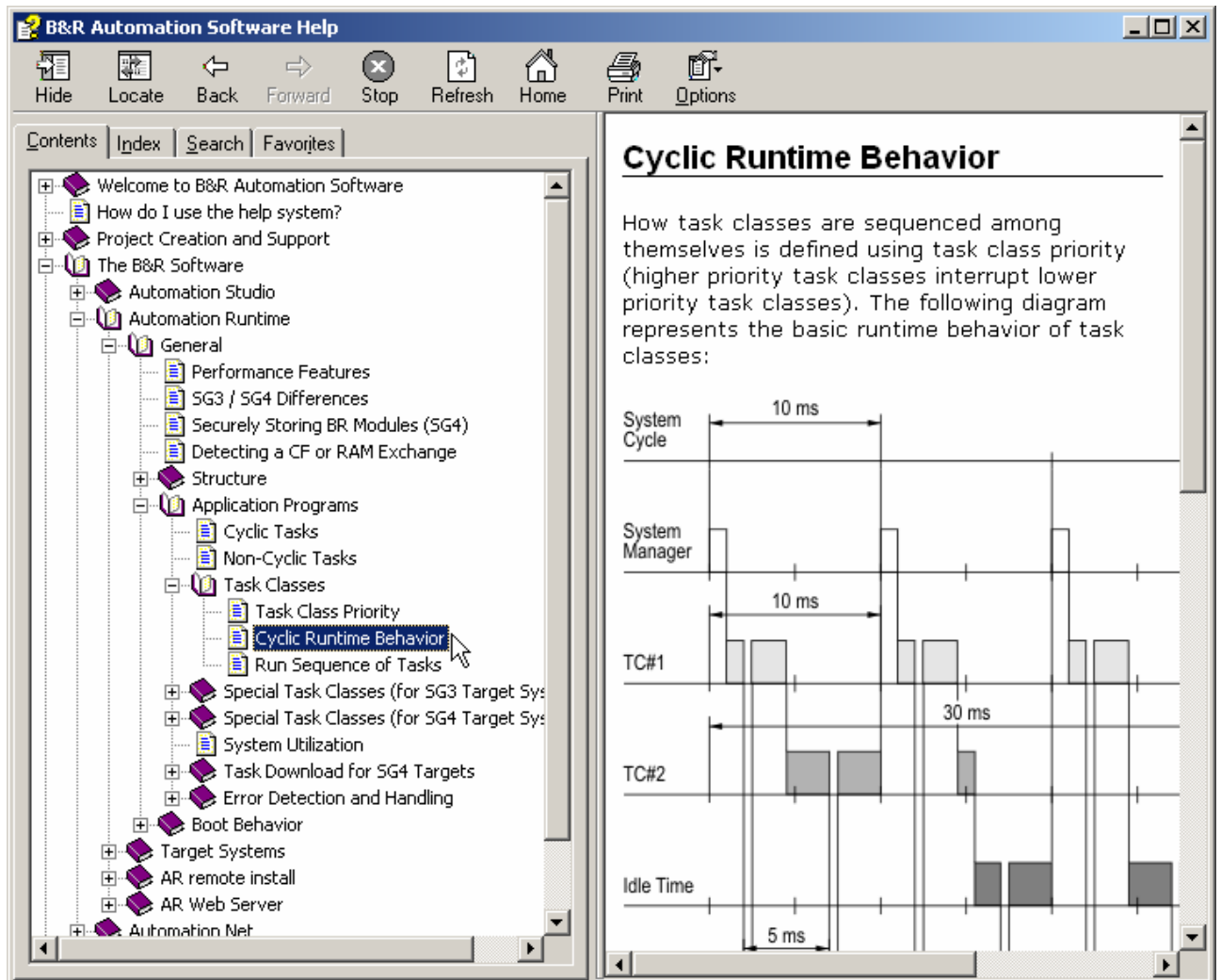
En la ventana de "Software Configuration" se pueden arrastrar las tareas del árbol de software a los distintos tiempos de ciclo que hay en la CPU. También se pueden añadir tareas con el icono "Add Object" (el icono situado a la izquierda)

Object Name	Version	Transfer ...	Size (byt...	Source
[-] CPU				
[-] Cyclic #1 - [10 ms]				
[-] Programa1	1.00.0	UserROM	0	Programa1
[-] Cyclic #2 - [20 ms]				
[-] Cyclic #3 - [50 ms]				

Con los iconos superiores se puede acceder a las distintas partes de código o a las declaraciones de variables o tipos globales y locales.



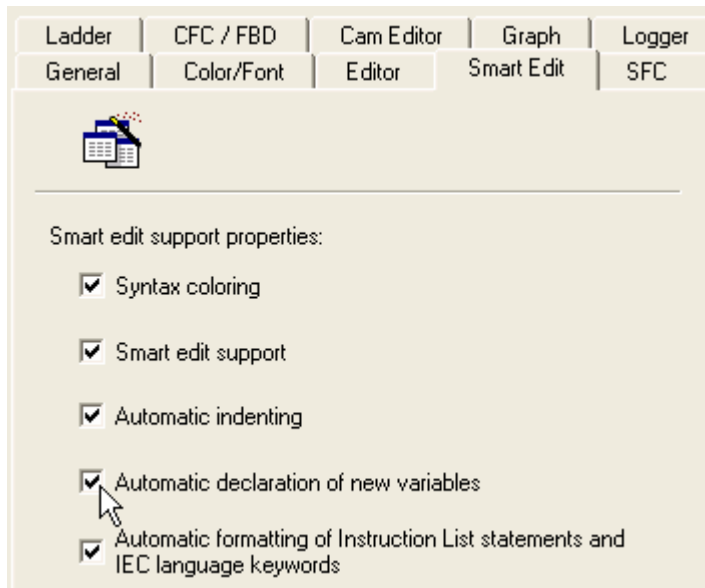
Podemos encontrar más información acerca del funcionamiento del Automation Runtime en la ayuda de Automation Studio:



## 1.6 Definición de variables y constantes

Automation Studio gestiona automáticamente variables, constantes o arrays. Ayuda a declarar las variables, y una vez declaradas se pueden llamar fácilmente desde el editor de programa.

Para que la declaración de variables automática funcione hay que tener la opción de “Automatic declaration of new variables” marcada. La encontramos en el menú “Tools - options - Smart Edit”.

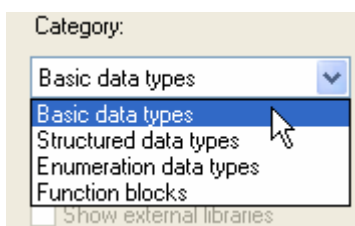


Para declarar una variable se escribe su nombre en el programa y cuando haya un salto de línea aparece directamente el dialogo de nueva variable.

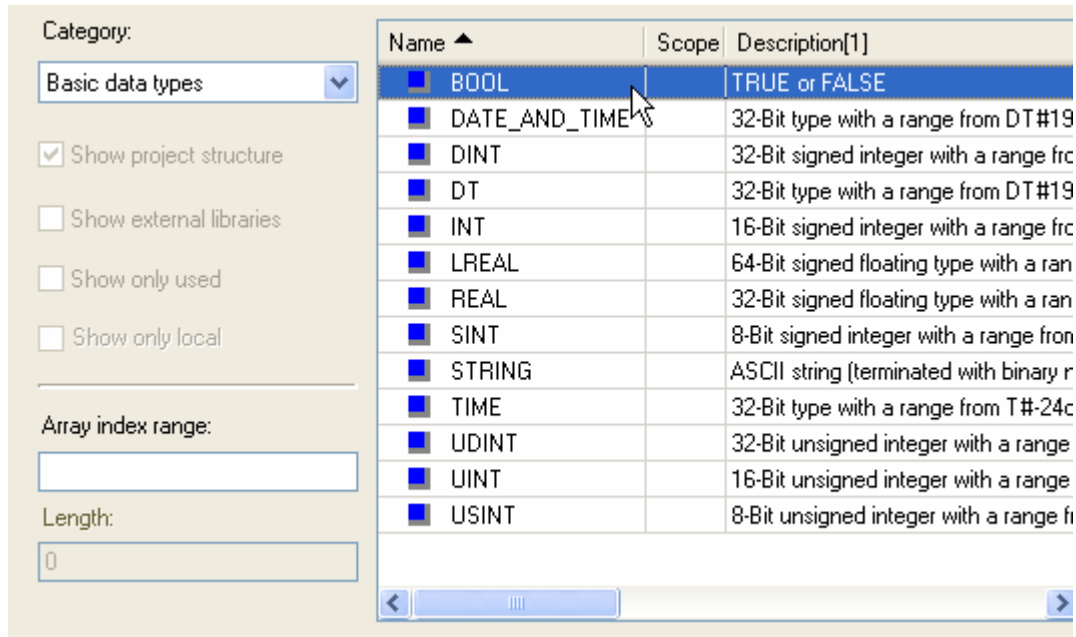
Name	Type	Local	Reference	Constant	Value	Description
variable	BOOL	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		

En este dialogo se puede escoger el tipo de esta variable. Pulsando espacio 2 veces o haciendo doble clic sobre el tipo aparece el dialogo de declaración de variable.

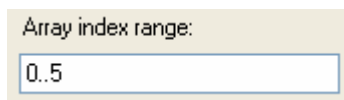
En “Category” se puede escoger los tipos básicos (Basic data types), creados por el usuario (Structured data types), bloques de funciones (Function Blocks) o enumeraciones (Enumeration data types).



En “Basic data types” aparecen los tipos habituales en cualquier aplicación.



En este dialogo se puede escoger el tipo de variable y también definir el rango de un vector en "Array index range" si lo que se quiere es definir un array.



Después del tipo se puede definir la validez de la variable, que puede ser:

- Local: sólo se puede llamar desde la tarea en la que está definida.
- Global: se puede llamar desde todas la tareas.

También se puede definir si es una variable en memoria o una constante. Por defecto las variables son variables en memoria, para definir una constante hay que marcar constante y asignarle un valor.

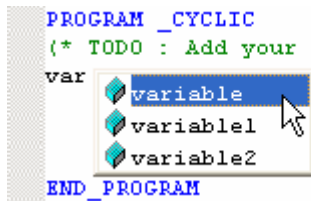
Name	Type	Local	Reference	Constant	Value
variable2	USINT	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	25

Para crear una variable que sea remanente, o sea que se mantenga en memoria cuando se apaga el autómatas, se selecciona "RETAIN" en la casilla "Value".

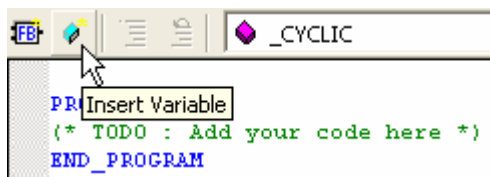
Name	Type	Local	Reference	Constant	Value
variable2	USINT	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	RETAIN

Para insertar una variable ya definida en un programa, no hay que volver a escribirla sino que se pueden escribir las primeras letras y después puede escogerse de las variables ya definidas.

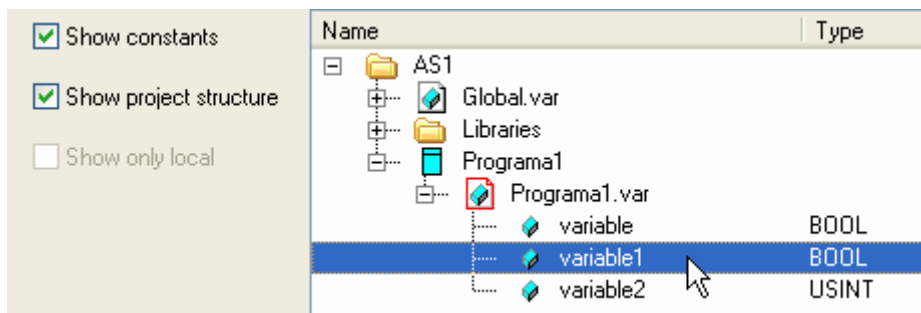
Para introducir una variable en un programa se escribe primero alguna letra de la variable y después se pulsa ctrl. + espacio para que aparezca una pequeña lista.



También se puede clicar en el icono de insertar variable.

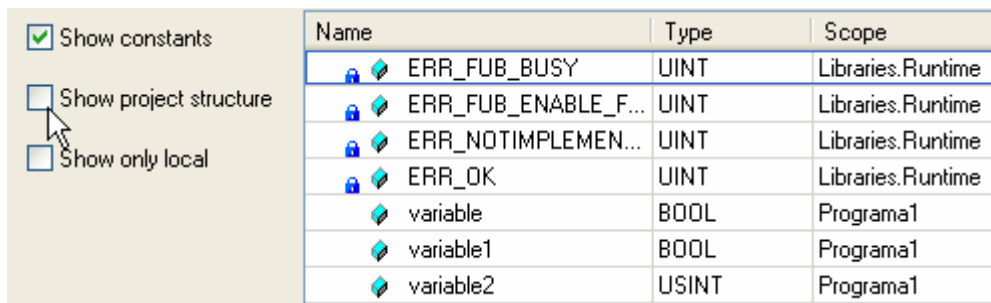


Entonces aparece el dialogo de añadir variable.



Para poder encontrar mejor la variable, se puede filtrar introduciendo las primeras letras en el recuadro de "Filter".

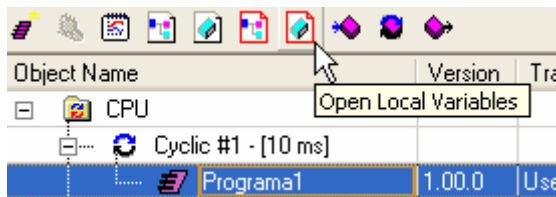
El dialogo aparece con el árbol de software que se puede ir abriendo para encontrar las variables, pero si se quita la opción "Show Project structure" simplemente muestra la lista con todas las variables.



Una vez seleccionada la variable, esta aparece escrita en el código.

Para consultar o cambiar las variables que se usan en una tarea se puede usar el dialogo de declaración de variables. Se puede abrir desde los ficheros

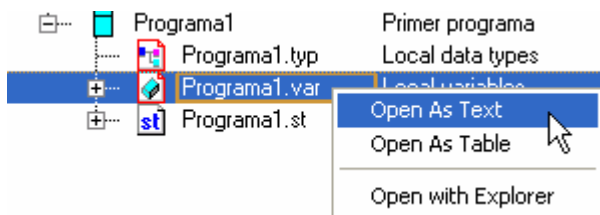
Global.var o Programa1.var o en los iconos en la ventana de "Software Configuration".



En este dialogo se pueden modificar los nombres de variables, validez o modificar los valores de las constantes. Es importante guardar los cambios antes de transferir.

Name	Type	& Reference	Constant	Value
variable1	BOOL	<input type="checkbox"/>	<input type="checkbox"/>	
variable	BOOL	<input type="checkbox"/>	<input type="checkbox"/>	
variable2	USINT	<input type="checkbox"/>	<input type="checkbox"/>	

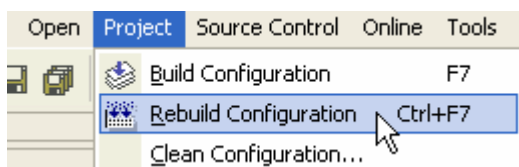
También hay la opción de abrir la definición de variables como un fichero de código estándar pulsando botón derecho sobre el fichero .var - "open as text"



```

VAR
  variable1 : BOOL;
  variable  : BOOL;
  variable2 : USINT;
END_VAR
    
```

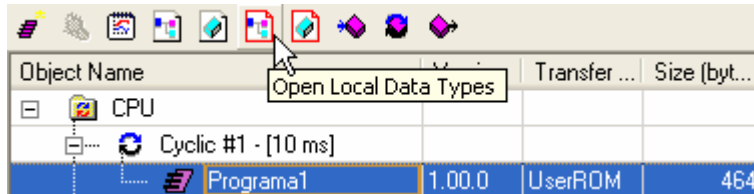
Si se cambia una variable de global a local o si se eliminan variables del programa estas permanecen ocupando memoria en el PLC. Para hacer limpieza de todas las variables, hay que hacer "Rebuild Configuration", que se encuentra en el menú "Project - Rebuild Configuration".



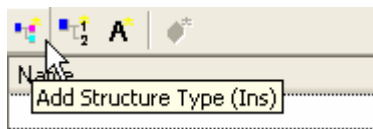
## 1.7 Definición de tipos

Automation Studio permite definir los propios tipos de variables del usuario, que pueden ser estructuras, arrays o combinaciones de estos y que pueden incluir otros tipos creados por el usuario.

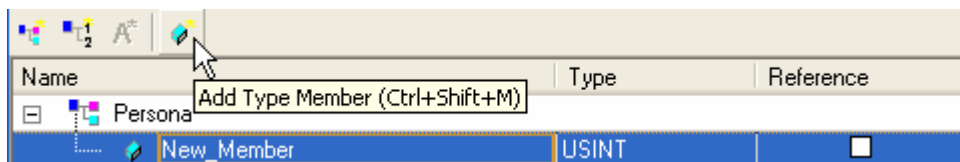
Se puede crear un nuevo tipo abriendo el fichero Programa1.typ o Global.typ o con los iconos de la ventana de "Software Configuration". (También se puede abrir como tabla o como código de texto)



En la ventana de data "types" se crea un nuevo datatype clicando en el botón "Add Structure type".



Se introduce el nombre del tipo, en este ejemplo será un tipo "persona". Cuando se ha creado el tipo, se añaden los ítems deseados con el botón "Add type member".



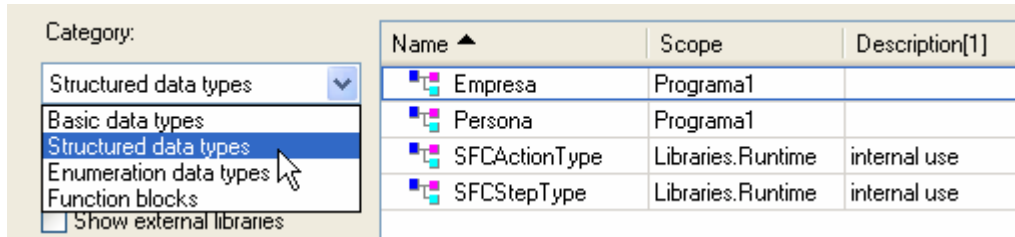
Para cada ítem, se asigna un nombre y un tipo, con el mismo dialogo que se ha visto en la definición de variables.

Ahora se crea un tipo con nombre "Empresa" que va a incluir el tipo anterior "Persona".

En este caso, el tipo empresa tiene un campo String con el nombre de la empresa, un campo con el número de trabajadores y un array del tipo persona.

Name	Type	Reference
Empresa		
Nombre	STRING[80]	<input type="checkbox"/>
NumPersonal	UINT	<input type="checkbox"/>
Personas	Persona[0..50]	<input type="checkbox"/>
Persona		
Altura	REAL	<input type="checkbox"/>
edad	USINT	<input type="checkbox"/>

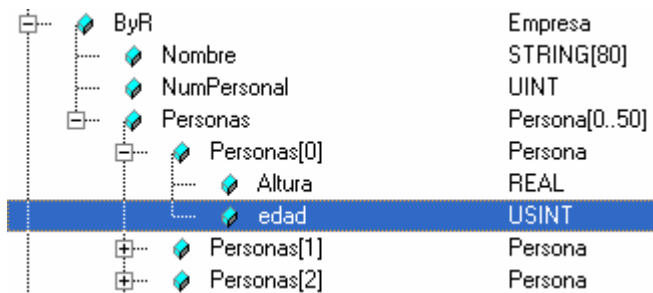
Para usar los nuevos tipos al definir una variable en Automation Studio, se crea una variable como siempre y en el dialogo de nueva variable se selecciona "Structured data types".



La definición de variable queda con ese aspecto.

Name	Type	Local	Reference	Constant
ByR	Empresa	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Ahora, al insertar la variable, se pueden escoger los distintos campos.

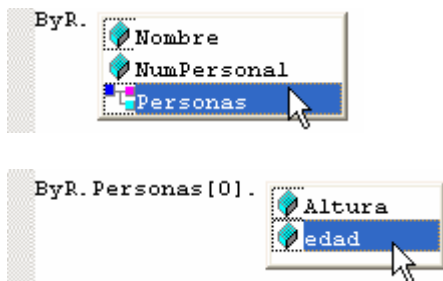


El acceso a sub-estructuras se hace con el punto y a los arrays entre []. Para acceder a la edad de la segunda persona se escribe.

```
ByR.Personas[1].edad := 20;
```

Hay que tener en cuenta que si se crea un array de 8 datos, se va a acceder a los distintos datos con los índices de 0 a 7.

También se pueden ir accediendo a los campos mientras se va escribiendo: cada vez que se escribe un punto, el programa sugiere la lista de posibles opciones.



## 1.8 Mapeado y configuración de E/S

Para poder asignar una variable a un punto de Entrada o Salida usamos el "I/O Mapping".

El mapeado de entradas / salidas lo encontramos en la pestaña de "Physical View".



En el árbol de hardware con doble click en la tarjeta de entradas / salidas o con el botón derecho e I/O Mapping se abre la ventana de mapeado.



Channel Name	Data Type	Task Class	PV or Channel Name
+ ModuleOk	BOOL		
+ DigitalInput01	BOOL		
+ DigitalInput02	BOOL		
+ DigitalInput03	BOOL		
+ DigitalInput04	BOOL		
+ DigitalInput05	BOOL		
+ DigitalInput06	BOOL		
+ DigitalInput07	BOOL		

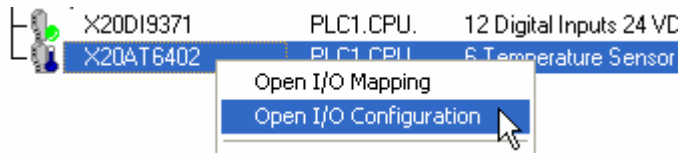
En el "I/O Mapping" se pueden ver todos los elementos mapeables de que dispone la tarjeta, haciendo doble click en "PV Name" se escoge que variable se asignará a la entrada.

Channel Name	Data Type	Task Class	PV or Channel Name
+ ModuleOk	BOOL		
+ Temperature01	INT	Automatic	Programa1.variable2
+ Temperature02	INT		
+ Temperature03	INT		

En este caso se ve una tarjeta de entradas de temperatura, por lo que en "Data Type" aparece INT. Eso significa que sólo dejará asignar variables tipo INT a esta entrada o salida.

También se puede acceder a la configuración de las entradas y salidas con el botón derecho - "Open I/O Configuration".





Desde esta ventana se configura el tipo de entrada, los filtros, la configuración de un sensor de temperatura o el funcionamiento de un encoder.

Como ejemplo, se muestra la configuración de un sensor de temperatura para una tarjeta X20AT6402.

IF6.ST2		X20AT6402	
Function model	internal comp...	Module's operating mode	
<b>General</b>			
Module supervised	on	Service mode if there is no hardware	
Module information	off	Additional module information	
IO cycle counter	off	IO cycle counter	
Input Filter	20 ms	Input Filter	
Sensor type	J	Sensor type	
Disable channel 01	J	Selectively disabling of not used cha	
Disable channel 02	K	Selectively disabling of not used cha	
Disable channel 03	S	Selectively disabling of not used cha	
Disable channel 04	N	Selectively disabling of not used cha	
Disable channel 05	$\pm 32767 \mu V, 1 \mu$	Selectively disabling of not used cha	
Disable channel 06	$\pm 65534 \mu V, 2 \mu$	Selectively disabling of not used cha	
<b>Simulation</b>			
Simulation device		Assigned simulation device	

Hay dos configuraciones importantes para todos los módulos:

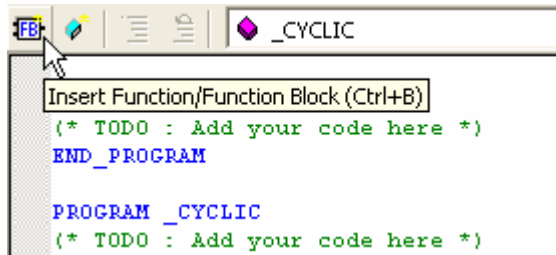
**Module supervised:** si está configurado en "on", cuando el módulo no funciona o se desconecta de la red X2X el autómat se pone en modo diagnóstico. Si está en "off", el autómat manda un mensaje de warning al logger pero sigue funcionando.

**Module information:** si está configurado en "on", en el "I/O Mapping" aparece información adicional asignable a una variable, como el número de serie, la ID del módulo, revisión y firmware.

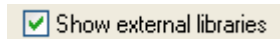
### 1.9 Insertar funciones de librerías

Si usamos los lenguajes IEC 61131 y B&R Automation Basic, Automation Studio añade automáticamente la librería al proyecto así como la declaración de la variable, la sintaxis de los parámetros de entrada y la llamada a la función.

Para añadir un bloque de función, como por ejemplo un temporizador, hay que pulsar sobre el botón de insertar bloque de funciones.



En el dialogo de insertar bloque de funciones se selecciona la opción "show external libraries" para ver las librerías que aún no están en el proyecto.



El bloque de funciones de temporizador se encuentra dentro de la librería "Standard".

Name	Description[1]
RIGHT	extracts a certain amount (L) of rightmost characters fro...
RS	sets or resets the output (reset has priority)
SEMA	blocks access within the same task class (simple sema...
SR	sets or resets the output (set has priority)
TOF	implements a switch off delay
TOF_10ms	implements a switch off delay based on 10 ms steps
TON	implements a switch on delay
TON_10ms	implements a switch on delay based on 10 ms steps
TP	implements a pulse generator
TP_10ms	implements a pulse using 10 ms steps
+	Sys_lib The SYS_LIB library contains functions f
+	VCLIB
+	VcScrSht
+	Visapi

Después de seleccionar la función hay que dar un nombre a la variable asociada a la instancia de esa función. Evidentemente el tipo de esta variable será de tipo que corresponda a la función llamada.

Name	Type	Local	Reference	Constant	Value
TON_0	TON	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Automation Studio creará entonces el código necesario para la función. La sintaxis dependerá de cada lenguaje de programación, pero siempre se estructura en tres bloques:

- Definición de los parámetros de entrada.
- Llamada a la función.
- Lectura de los parámetros de salida.

```
(* cyclic program *)
(* definición de entradas *)
TON_0.IN := Inicia;
TON_0.PT := T#5s200ms;
(*Llamada a la función*)
TON_0();
(*Lectura de salidas*)
Fin := TON_0.Q;
```

Ahora se puede ver el estado del Temporizador en el Watch. El Watch es una herramienta de diagnóstico que se describe en el siguiente capítulo.

Name	Type	Scope	Force	Value
Fin	BOOL	local		TRUE
Inicia	BOOL	local		TRUE
TON_0	TON	local		
IN	BOOL			TRUE
PT	TIME			T#5s_200ms
Q	BOOL			TRUE
ET	TIME			T#5s_200ms
StartTime	TIME			T#1m_14s_900ms
M	BOOL			TRUE
Restart	UDINT			0

La sintaxis en Automation Basic es:

```
(* cyclic program *)
(* definición de entradas *)
TON_0.IN= Inicia
TON_0.PT= T#5s200ms
(*Llamada a la función*)
TON_0 FUB TON()
(*Lectura de salidas*)
Fin = TON_0.Q
```

y en ANSI C:

```
#include <bur/plctypes.h>
#ifdef _DEFAULT_INCLUDES
#include <AsDefault.h>
#endif

void _CYCLIC TimerCYCLIC( void )
{
    /* Definición de entradas*/
    TON_0.IN = Inicia;
    TON_0.PT = 5200; /*5s200ms*/

    /*Llamada a la función*/
    TON(&TON_0);

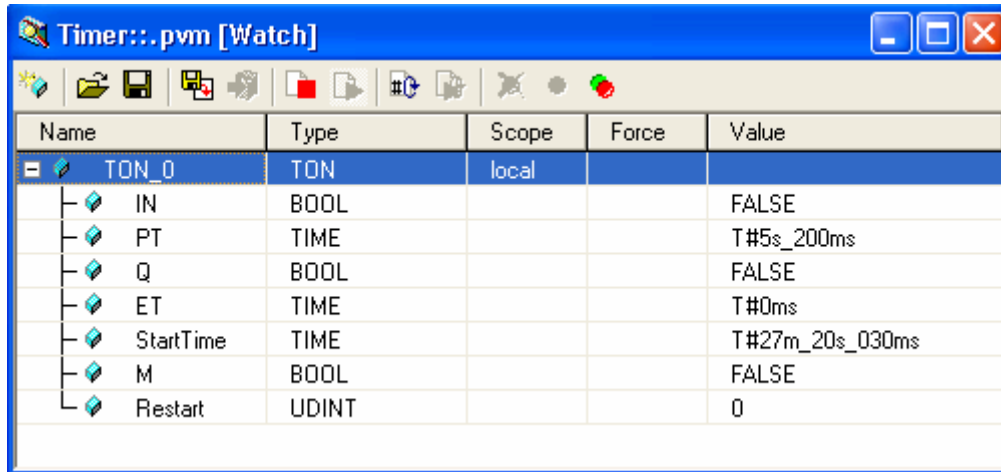
    /*lectura de salidas*/
    Fin = TON_0.Q;
}
```

## 1.10 Herramientas de Debug

Hay muchas posibilidades de debugging en Automation Studio. Aquí se detallan brevemente.

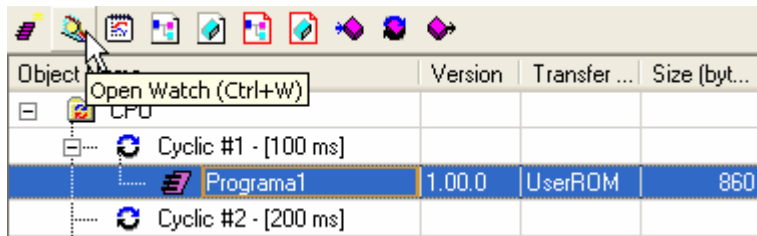
### - Watch:

El "Watch" permite ver y cambiar el valor de las variables de la tarea (o las globales). También sirve para forzar las entradas o salidas.



Name	Type	Scope	Force	Value
TON_0	TON	local		
IN	BOOL			FALSE
PT	TIME			T#5s_200ms
Q	BOOL			FALSE
ET	TIME			T#0ms
StartTime	TIME			T#27m_20s_030ms
M	BOOL			FALSE
Restart	UDINT			0

Para acceder al "Watch" hay que pulsar el icono de "Watch" en la ventana de "Software Configuration".



Object	Version	Transfer ...	Size (byt...
Open Watch (Ctrl+W)			
CPU			
Cyclic #1 - [100 ms]			
Programa1	1.00.0	UserROM	860
Cyclic #2 - [200 ms]			

### - Logger:

El "Logger" muestra todos los warnings o mensajes de error que se han producido en el autómata a nivel de sistema.

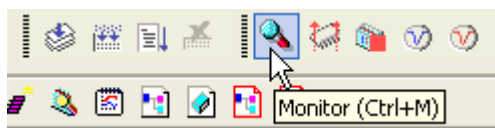
Es imprescindible para ver cual es el último error que ha habido después de un reinicio de la CPU debido a un problema grave.

	Level	Time	Logger M...	Error Description	ASCII Data	Location
57	Warning	2007-01-29 00:05:3...	System	PLCopen MC Library: Error with details in '...	nmalloc error (see info)	Online
58	Warning	2007-01-29 00:04:4...	System	WARNING: PLC Reset: Reset/Halt using...		Online
59	Warning	2007-01-29 00:01:1...	System	PLCopen MC Library: Initialization aborted		Online
60	Warning	2007-01-29 00:01:1...	System	PLCopen MC Library: Error with details in '...	nmalloc error (see info)	Online
61	Error	2007-01-29 00:00:3...	System	AR-PnP: module not plugged (recognized...	SL1:SS1.IF1.ST1.IO	Online
62	Warning	2007-01-29 00:00:1...	System	AR-NvMem: Recognition of a new CF/H...	HDD/CF/RAM was changed	Online
63	Warning	2007-01-29 00:00:1...	System	WARNING: System halted due to power l...	Boot:Powerup	Online
64	Information	2007-01-29 00:00:1...	System		base system log module creat...	Online

Para abrir el "Logger" hay que ir al menú "open - logger".

#### - Monitor Mode:

Pulsando en el icono de la lupa se entra en "Monitor Mode":



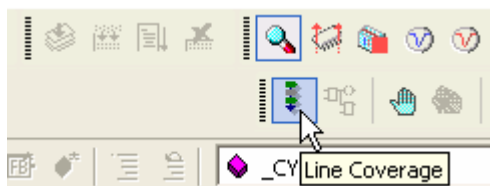
Cuando se entra en "Monitor Mode", la ventana de "Software Configuration" se duplica para mostrar en dos ventanas separadas los objetos que están configurados en el proyecto y los que están en el PLC. Con distintos colores se puede ver si no coinciden las dos ventanas o ha habido modificaciones.

Project			Target		
Cyclic #1 - [100 ms]			Cyclic #1 - [100 ms]		
Programa1	1.00.0		Programa1	1.00.0	Running
Cyclic #2 - [200 ms]			Cyclic #2 - [200 ms]		
Nuevo	1.00.0		Modific	1.00.0	Running
Modific	1.00.0		Cyclic #3 - [500 ms]		
Cyclic #3 - [500 ms]			Cyclic #4 - [1000 ms]		
Cyclic #4 - [1000 ms]			Timer	1.00.0	Running

También permite ver el estado de las variables cuando se abre una tarea, ya sea en ladder, ST o C.

#### - Line Coverage:

Si se está en "Monitor Mode", dentro de las tareas se puede activar el "Line coverage", que muestra de color verde que líneas de la tarea se están ejecutando.



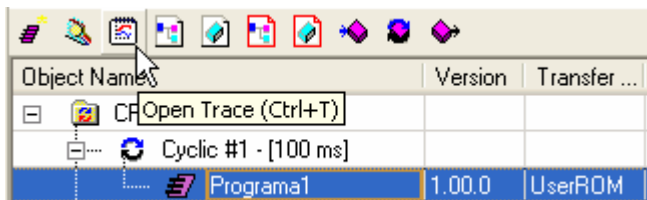
### - Breakpoints:

También dentro de la tarea y en "Monitor Mode", se pueden añadir "Breakpoints", que paran el programa en la línea que se seleccione.

```
PROGRAM _CYCLIC
  b := 2;
  Variable := 5;
  c := 3;
```

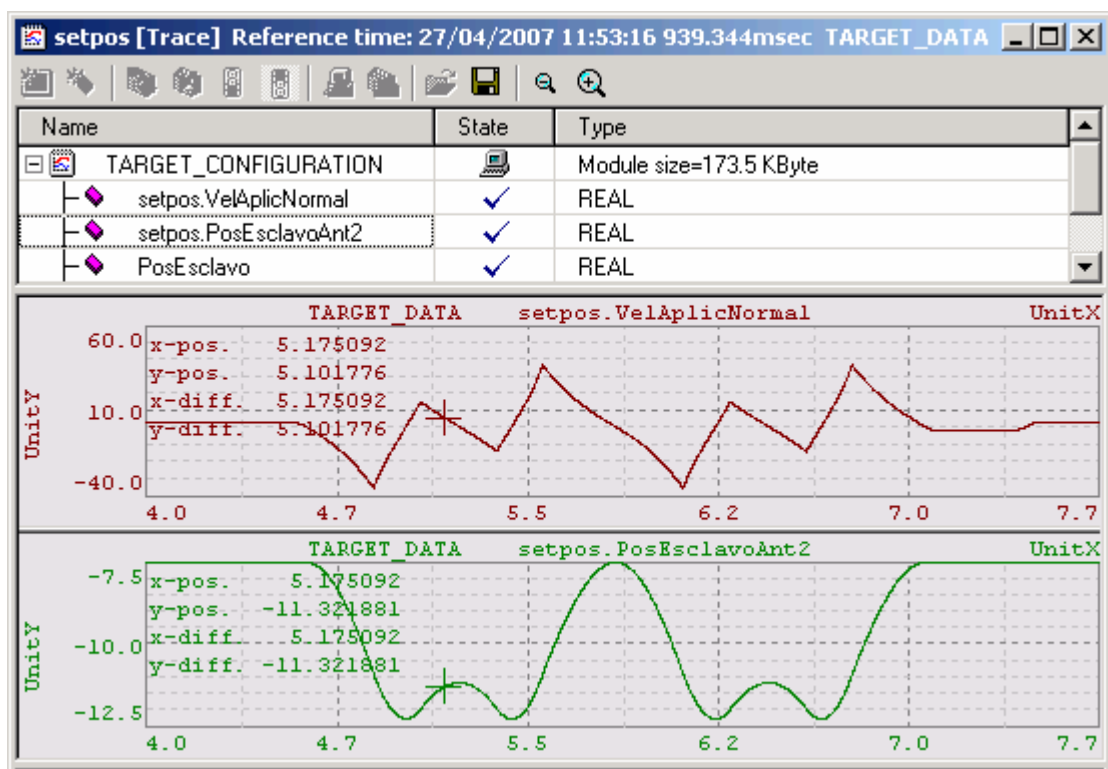
### - Trace:

En "Monitor Mode" se puede abrir el trace con el icono de "Trace" de la ventana de "Software Configuration".



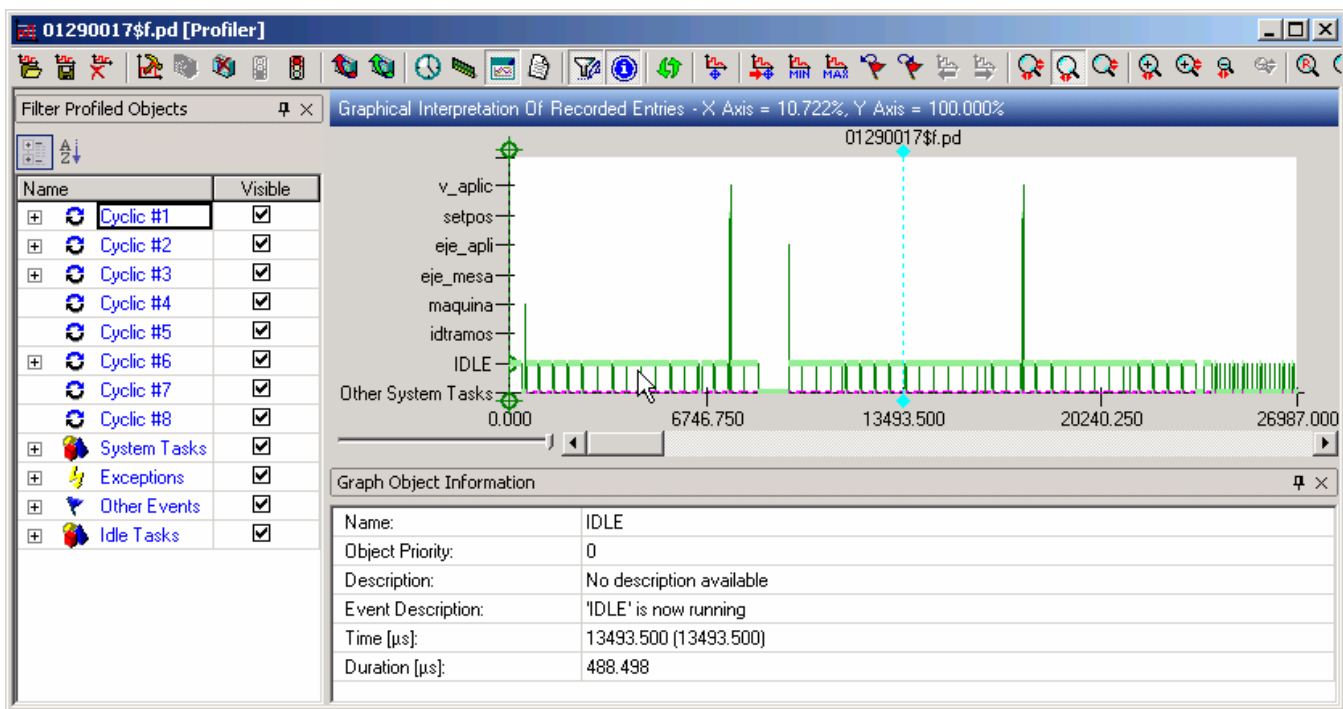
Para hacer el trace de una variable hay que seguir los siguientes pasos:

- Clic en "insert trace configuration" para crear la configuración de trace.
- Clic en "insert a new variable" para ir insertando las variables que se quieren ver.
- Clic en "Install trace" para arrancar el trace en el PLC.
- Parar o arrancar la lectura de datos con el semáforo verde o rojo.
- Clicar en "Show target data" para transferir los datos y visualizar el trace.



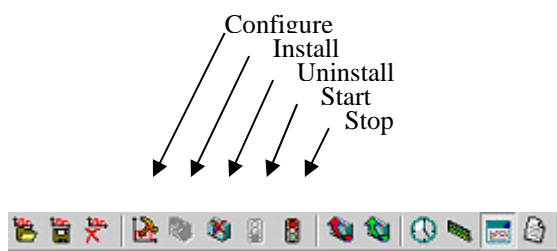
- Profiler:

El Profiler permite ver en que tareas se está ocupando la CPU en todo momento. Se puede abrir desde el menú en "Open - Profiler".



Para configurar el profiler hay que seguir los siguientes pasos:

- Configurar el profiler con el botón "configure"
- Instalar el profiler en el target con el botón "Install"
- Parar o arrancar la lectura de datos con el semáforo verde o rojo.
- Leer datos con "Update Data Object"



### 1.11 Cambio de sistema operativo

Cada versión de AS lleva asociada una o varias versiones de Automation Runtime, que es el sistema operativo que se carga en un PLC o pantalla. Los sistemas operativos se identifican con una letra seguida de un número de versión. Por cada número se le van asignando letras a medida que salen versiones nuevas.

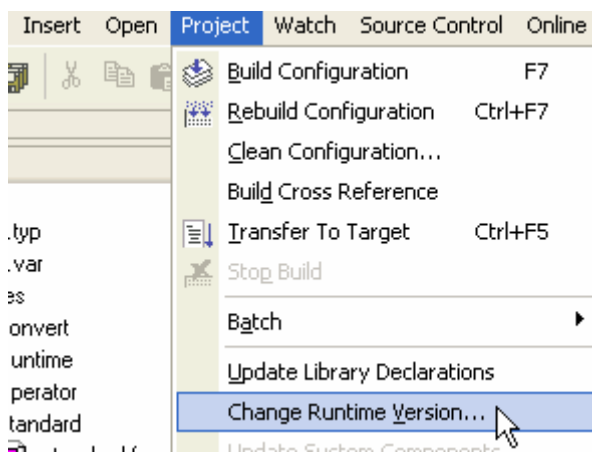
Por ejemplo después de la versión C2.90 vendrá la versión D2.90.

Para poder compilar y transferir un programa a un PLC donde ya se ha instalado un Automation Runtime, hay que tener en el PC el Automation Studio con la versión de Automation Runtime que tenga el PLC.

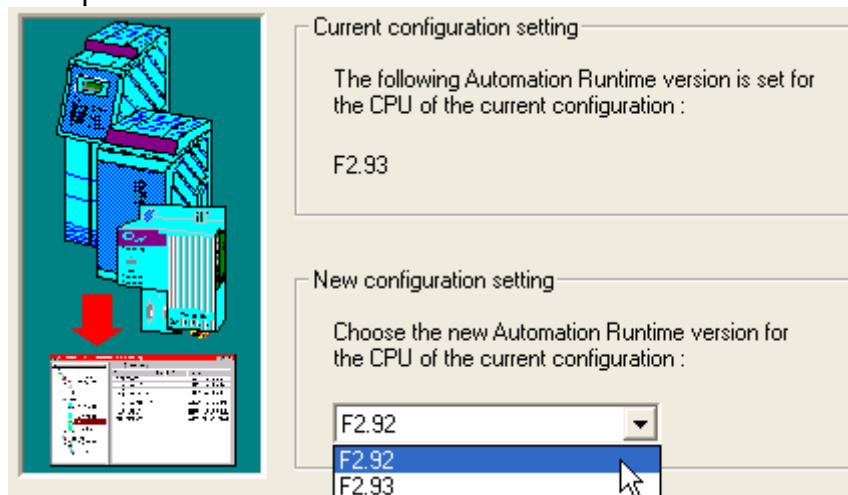
Si no se dispone del sistema operativo, Automation Studio avisará cuando se conecte on-line con el PLC. La versión de sistema operativo se puede leer en la barra de Status.

Si se quiere trabajar con un PLC y no se dispone del sistema operativo, se puede cambiar el sistema operativo del proyecto.

Para hacerlo, hay que clicar en "Project - Change Runtime version".



Entonces, aparece un dialogo que indica el sistema operativo que hay actualmente en el proyecto y donde se puede seleccionar el sistema operativo nuevo que se usará en el PLC.





## 1.12 Trabajar con distintas configuraciones

Como ya se ha comentado en este documento, Automation Studio permite trabajar con distintas configuraciones. Esas distintas configuraciones se basan en que muchas veces hay software o partes de software que funcionan para distintas máquinas, por eso se necesita un software que permita tener distintas tareas de software que sean comunes en todas las máquinas y configuraciones de máquina que pueden tener hardware muy distinto (incluso distintos PLC).

La parte izquierda del Automation Studio tiene 3 pestañas que se han ido viendo en este documento:

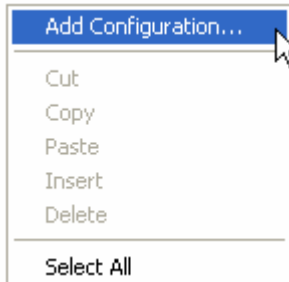


- La pestaña de **Logical View**, contiene los elementos de software comunes en cualquier configuración de máquina. Permite añadir tareas de software ficheros, crear packages (carpetas para agrupar objetos) o definir variables y tipos.
- La pestaña de **Configuration View**, sirve para crear y seleccionar las distintas configuraciones de máquina. Aquí se puede ver cual es la configuración que está activa.
- La pestaña de **Physical View**, permite configurar el hardware para la configuración activa en ese momento. Por cada configuración se puede tener una "Physical View" totalmente distinta: distinta CPU, distintas E/S, distintos servodrives, distinta "Software Configuration"...

En la pestaña de "Configuration View", se muestran las configuraciones del proyecto y la configuración activa queda escrita en negrita. Para cambiar de una configuración a otra, simplemente hay que hacer doble click sobre la configuración.

Configuration	Batch	Description
<ul style="list-style-type: none"> <li>X20_CPU [Active]           <ul style="list-style-type: none"> <li>Hardware.hc</li> <li>PLC1               <ul style="list-style-type: none"> <li>Cpu.sw</li> <li>Cpu.per</li> <li>IoMap.iom</li> <li>PvMap.vvm</li> <li>ArConfig.rtc</li> <li>sysconf.br</li> <li>sysconf.sys</li> </ul> </li> </ul> </li> <li>PowerPanel           <ul style="list-style-type: none"> <li>Hardware.hc</li> <li>PLC1</li> </ul> </li> </ul>	<input type="checkbox"/>	<p>Hardware topology</p> <p>Files belonging to this PLC</p> <p>Software configuration</p> <p>Declaration of permanent variables</p> <p>IO mapping file</p> <p>PV mapping file</p> <p>Runtime configuration file</p> <p>CPU system configuration</p> <p>CPU system configuration</p> <p>Hardware topology</p> <p>Files belonging to this PLC</p>

Si se quiere añadir una nueva configuración, hay que pulsar con el botón derecho - "Add Configuration".

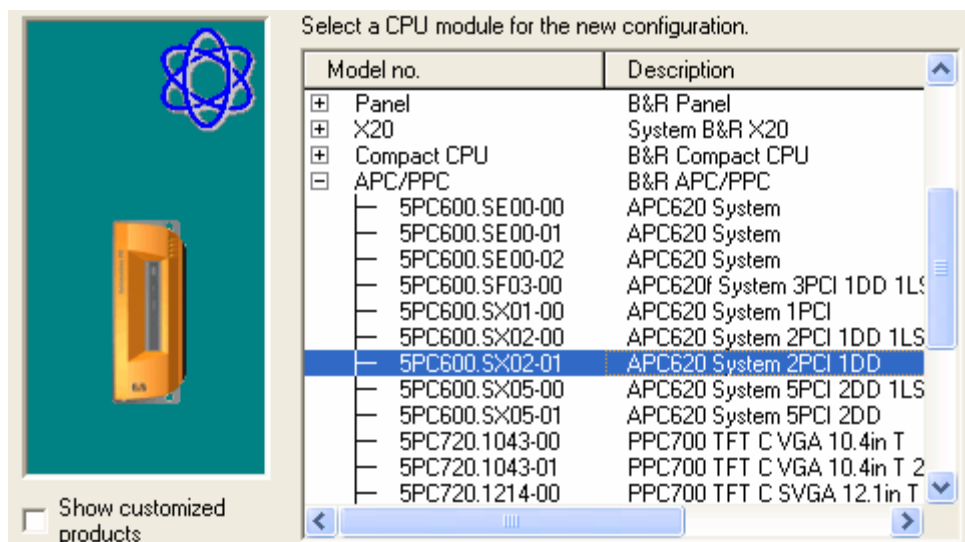


Cuando se añade la configuración hay que asignarle un nombre y decidir como se va a definir la configuración (manualmente o detectando el hardware conectado).

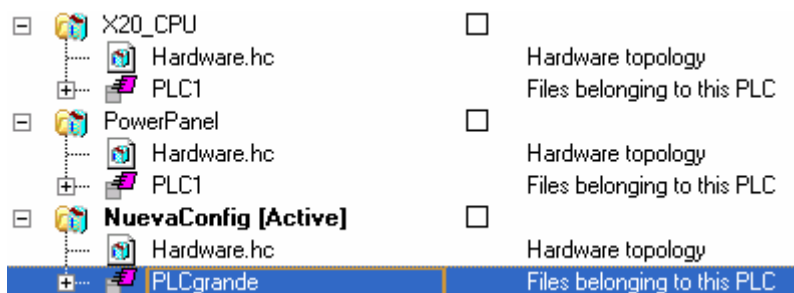
 A dialog box titled 'Enter the parameters of the new configuration.' contains the following fields and options:
 

- 'Name of the configuration:' with a text box containing 'NuevaConfig'.
- 'Name of the PLC:' with a text box containing 'PLCgrande'.
- Three radio buttons:
  - Define a new hardware configuration
  - Identify control system online
  - Reference an existing hardware configuration
- A text box below the radio buttons and a 'Browse...' button.

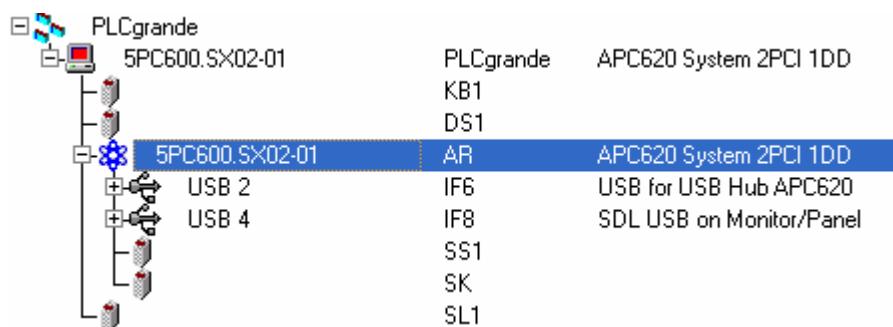
Al crear la configuración nueva hay que escoger un PLC para esta nueva configuración. Hay que hacerlo de la misma forma como si fuera un nuevo proyecto.



Cuando se active la nueva configuración haciendo doble click en la configuración que se ha añadido.



Si ahora se accede al árbol de hardware del “Physical View” aparecerá el hardware correspondiente.



Por cada configuración distinta hay que configurar la “Software Configuration”, que sirve para asignar que tareas del software común se asignan a cada configuración.

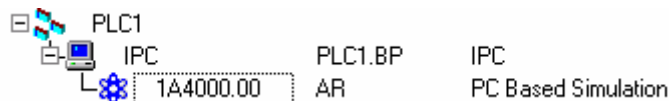
### 1.13 Simulación del Runtime - AR000

Una interesante configuración que se puede añadir al proyecto es la configuración para simular el Automation Runtime en el propio PC.

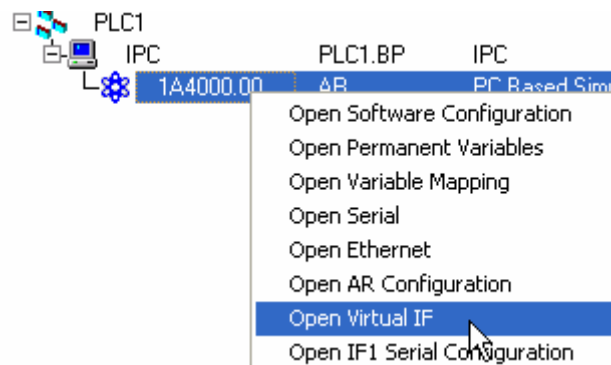
El Runtime simulado para PC se llama AR000, que se puede seleccionar junto con todos los distintos PLCs al generar una nueva configuración (1A4000.00).

+	Power Panel	System B&R Power Panel
-	PC Based AR	Open PC Based Runtime
-	1A4000.00	PC Based Simulation Runtime
-	1A4600.10	PC Based Runtime AR010 - H
-	1A4600.10-2	PC Based Runtime AR010, AF

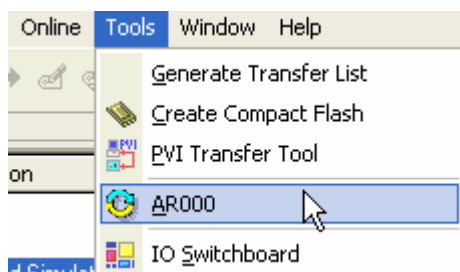
El AR000 crea un árbol de hardware con una "Software Configuration" donde se pueden añadir las mismas tareas de software que se han usado para las otras configuraciones, que se arrastran de la "Logical View".



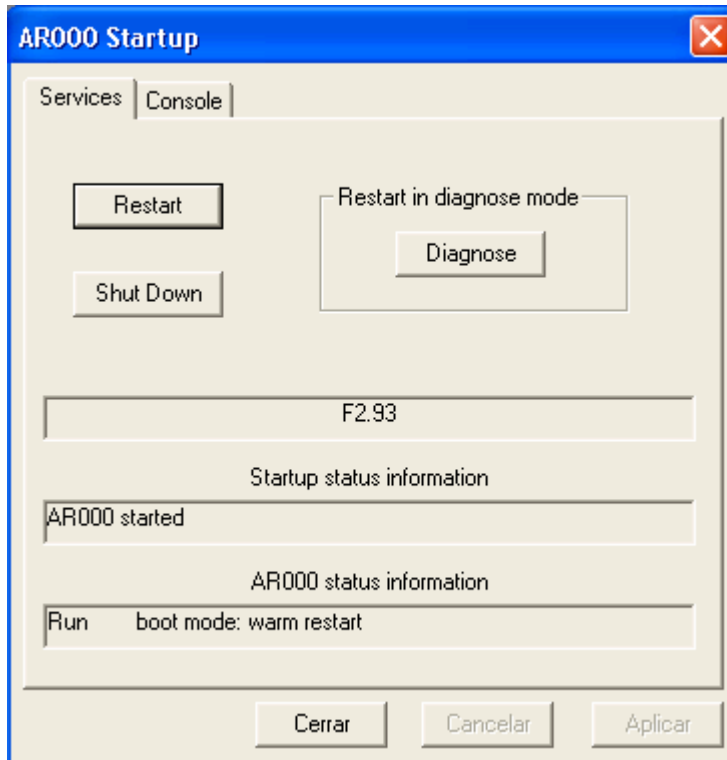
También es importante saber que pulsando sobre "Open Virtual IF" se puede añadir un servidor VNC al AR000 y simular una visualización.



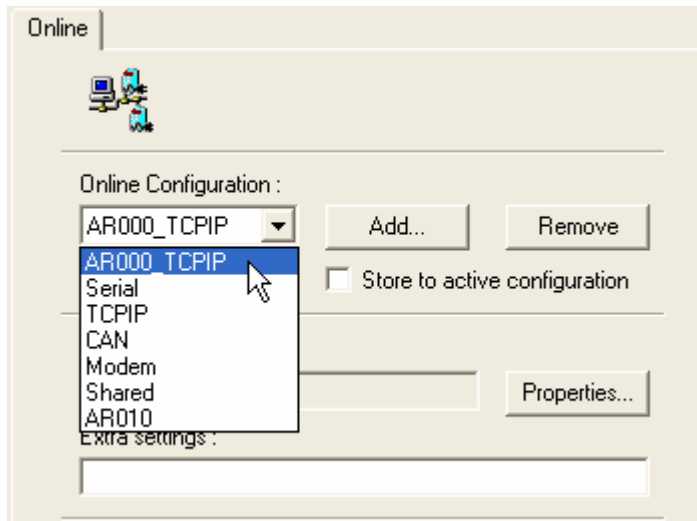
Cuando se ha creado toda la configuración hay que arrancar el AR000 en el menú "Tools - AR000".



En la pantalla del software AR000 se muestra su status y la información del estado en que se encuentra (boot, Run, Serv...)



Para conectar on-line con el AR000 hay que configurar la conexión en "Online - Settings" y escoger la configuración Online AR000\_TCPIP.



Si la configuración es correcta, la barra de status debe indicar que el AR000 está en Run. Para referirse al AR000 desde VNC hay que usar la dirección 127.0.0.1 (o "Localhost").

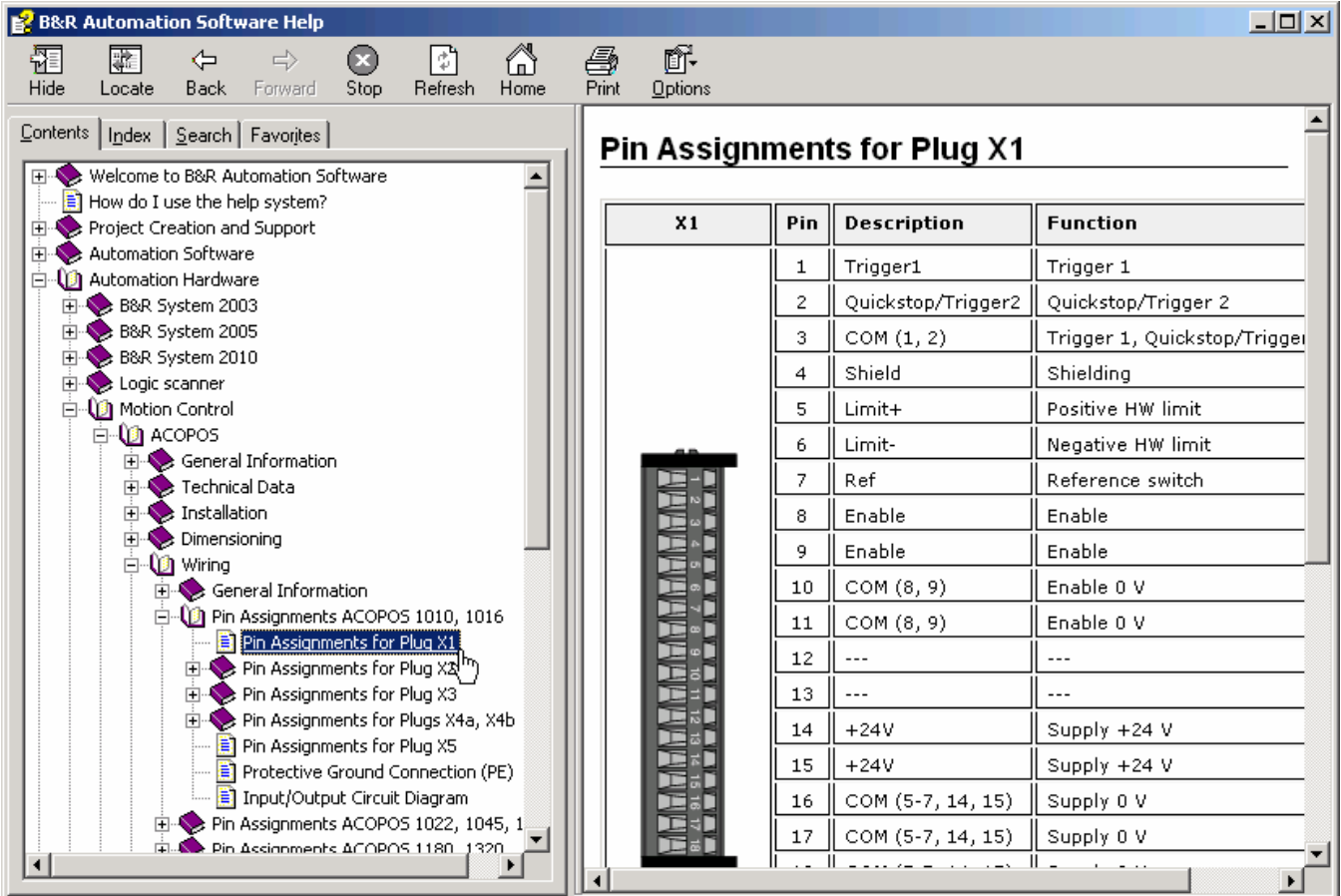


## 1.14 La ayuda de Automation Studio

Todo lo explicado en este documento puede encontrarse en la ayuda de Automation Studio. La ayuda es una potente herramienta que hay que ir consultando a medida que se va programando con Automation Studio.

La ayuda se divide en dos grandes grupos: el de Automation Software y el de Automation Hardware.

El grupo de Automation Hardware, contiene toda la información se encuentra en el manual de un equipo: conexionado, pinouts, características técnicas...



The screenshot shows the 'B&R Automation Software Help' window. The left pane displays a tree view of the help contents, with 'Pin Assignments for Plug X1' selected under the 'Automation Hardware' > 'Wiring' > 'Pin Assignments ACOPOS 1010, 1016' path. The right pane displays the 'Pin Assignments for Plug X1' table, which includes a small image of the plug connector and a table with columns for Pin, Description, and Function.

X1	Pin	Description	Function
	1	Trigger1	Trigger 1
	2	Quickstop/Trigger2	Quickstop/Trigger 2
	3	COM (1, 2)	Trigger 1, Quickstop/Trigger 2
	4	Shield	Shielding
	5	Limit+	Positive HW limit
	6	Limit-	Negative HW limit
	7	Ref	Reference switch
	8	Enable	Enable
	9	Enable	Enable
	10	COM (8, 9)	Enable 0 V
	11	COM (8, 9)	Enable 0 V
	12	---	---
	13	---	---
	14	+24V	Supply +24 V
	15	+24V	Supply +24 V
	16	COM (5-7, 14, 15)	Supply 0 V
	17	COM (5-7, 14, 15)	Supply 0 V

En el apartado de Automation Software, se encuentra la información necesaria para el programador. En la Sub-categoría de Automation Studio, se encuentra la ayuda para las diversas partes de Automation Studio: "Visual Components", "NC Software" (Motion), "Programming Languages"...

Hay que prestar especial atención a la categoría de "Libraries", donde se encuentra la información referente a todos los bloques de funciones de las librerías de Automation Studio, que debe ser consultada a menudo por el programador.

The screenshot shows the B&R Automation Software Help window. The left pane displays a tree view of the help content, with the 'STANDARD' library expanded to show 'Timer Function Blocks' and the 'TON' block selected. The right pane displays the documentation for the 'STANDARD - TON()' block.

### STANDARD - TON()

The TON function block implements a switch on delay.

#### Parameter

I/O	Parameter	Data Type	Description
IN	IN	BOOL	Input signal.
IN	PT	TIME	Delay Time
OUT	Q	BOOL	Output signal, the rising edge of the input signal is delayed by PT.
OUT	ET	TIME	Elapsed time

#### Call Syntax (Automation Basic)

```
TON(IN,PT,Q,ET)
```

#### Time Diagram

Por cada bloque de funciones se encuentra: descripción del bloque, listado de parámetros de entrada o salida, sintaxis del bloque, números de errores y un ejemplo del bloque en distintos lenguajes.

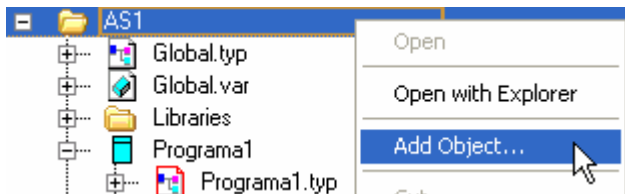
## 2. VISUALIZACIÓN

### 2.1 Añadir una visualización

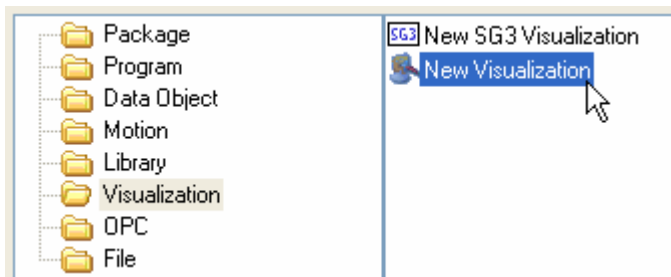
Para añadir una visualización en AS, ya sea para un panel o para un servidor VNC, hay que hacer dos pasos: añadir un objeto de visualización y después asociar ese objeto al panel que se tenga.

Los objetos de visualización, que contienen la edición de las pantallas, se generan en la "Logical View" y se asocian con el hardware en la "Physical View" (cada configuración puede tener distintos hardware de visualización).

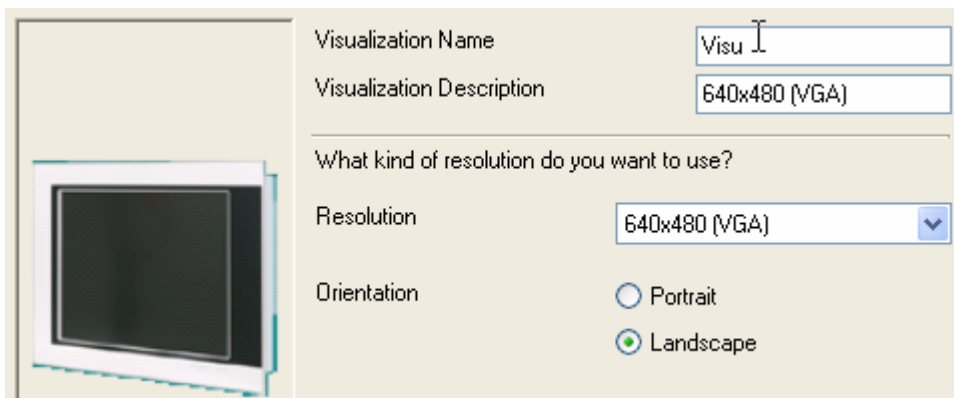
El primer paso es crear el objeto de visualización en "Logical View - Add object":



Hay que seleccionar "Visualization - New Visualization"



Y después hay que escoger el nombre y resolución (para un panel de 10" será 640 x 480).



Automation Studio ha creado ahora un objeto de visualización que se puede ver junto con los otros objetos que hay en el "Logical View". Cuando se quiera



editar solo hay que dar doble clic en el objeto visualización y se abrirá el editor de Visual Components.

Una vez creado el Objeto hay que asignarlo al hardware.

El hardware se asigna en el "Physical View" pulsando el botón derecho sobre "display" y seleccionando "Open VC Mapping" (En el caso que tengamos un Power Panel).



En "VC Object Name" hay que seleccionar el objeto de visualización que se haya creado para esta pantalla.

Module Address	Master	VC Object Name	Rotation Angle
DS1		<input type="text" value="Visu"/>	

Para que el objeto aparezca en el desplegable, tiene que estar añadido en la ventana de "Software Configuration". Hay que añadirlo si no se ha hecho automáticamente al crear el objeto de visualización.

No Data Objects			
Visualisation			
Visu	1.00.0	UserROM	
Binary Objects			

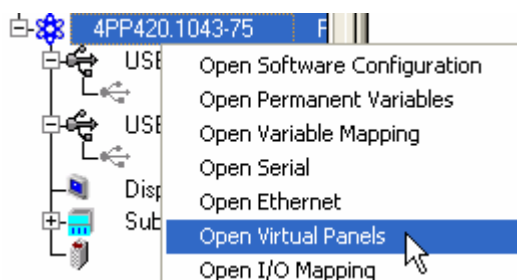
Una vez asignado el hardware y antes de compilar el programa hay que abrir el objeto de visualización de nuevo y guardar los cambios para que reconozca el panel.

## 2.2 Añadir un servidor VNC

El Automation Runtime de B&R permite añadir un servidor de VNC a la CPU, así se puede controlar la visualización desde un PC estándar utilizando un software cliente de VNC (por ejemplo: [www.realvnc.com](http://www.realvnc.com)).

Es importante aclarar que el VNC puede correr en cualquier CPU B&R SG4 (la generación que funciona con Compact Flash), tanto en las que tienen pantalla como en las que no tienen como también en el AR000 que es el Runtime de simulación.

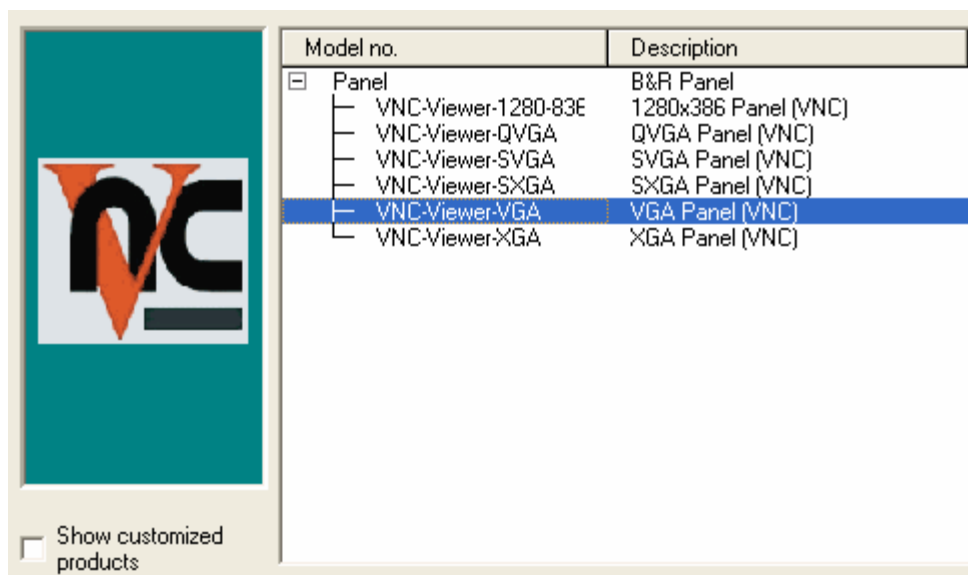
Para añadir el servidor de VNC, hay que seleccionar "Open Virtual Panels" en la "Physical Config".



Después hay que pulsar el botón derecho en la ventana de "Virtual Panels - insert".



Después se selecciona "VNC - Viewer".



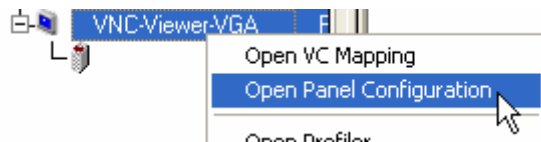
Una vez añadido el VNC hay que asignarle un objeto de visualización pulsando en "Open VC Mapping".



Aquí se puede asignar el objeto Visu que se ha creado antes o, simplemente poner que el VNC tiene como master un panel.

Module Address	Master	VC Object Name	Rotation Angle
IF8.ST1		<div style="border: 1px solid black; padding: 2px;"> <span style="float: right;">▼</span>   <span style="background-color: #0056b3; color: white; padding: 2px;">Visu</span> </div>	

Pulsando "Open Panel Configuration" se accede a la configuración, donde se pueden asignar dos passwords, uno permite solo ver la visualización y el otro permite pulsar botones en la pantalla.

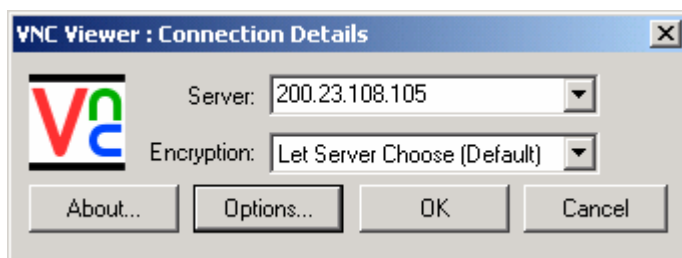


IF8.ST1		VNCVGA
VNC parameters		
Max. connections	1	Maximum allowed number of concurrent connections to th
Refresh rate	1000	[ms] Time between refreshes of the remote display
Passwords		
View only	read	Password for view only access to this VNC server
View and control	rw	Password for view and control access to this VNC server

La configuración anterior ya se puede transferir al panel.

Por la parte del PC hay que instalar el software cliente de VNC, que se puede descargar gratis de Internet.

En el cliente, hay que introducir la ip a la cual queremos conectar e introducir después el password.



En el caso que haya más de un servidor VNC en el mismo PLC, se escribe el puerto después de la IP separado por :.

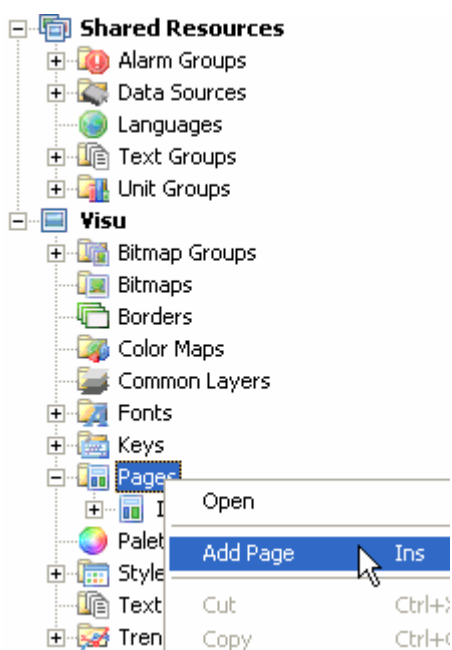
El puerto por defecto es el 5900 y cada nuevo servidor VNC que se añade al proyecto toma un nuevo valor sumando uno al anterior.

### 2.3 Funcionamiento básico de visual components

Visual components se divide en tres partes:

- **Árbol de recursos:** aquí se encuentran todos los recursos que pueden ser usados en nuestro objeto de visualización, como las páginas, bitmaps, grupos de texto...
- **Editor de recursos:** en la parte central, se van abriendo las ventanas de los recursos que vamos editando, especialmente las páginas.
- **Propiedades de objetos:** cuando clicamos sobre un objeto en el editor de páginas, en la parte derecha nos aparecen sus propiedades (tamaño, posición, color, datapoints...)

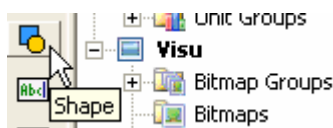
Para añadir una página al objeto de visualización hay que pulsar botón derecho del ratón sobre "Pages" y seleccionar "Add Page".



Al momento de crear la página se puede asignar un nombre y un índice para identificarla. También se puede escoger el estilo que se va a usar en esta página (los estilos se explicarán con más detalle)

Name	NuevaPagina
Index	20
StyleClass	default
<b>Position</b>	
<b>Appearance</b>	
<b>Runtime</b>	
Description	

Para añadir un objeto a la página, por ejemplo un simple cuadrado, hay que pulsar en la barra de objetos que se encuentra a la izquierda y dibujar el objeto en la página.



Una vez dibujado el cuadrado, aparecen sus propiedades.

Name	Shape_1
StyleClass	Shape
<b>Position</b>	
Left	70
Top	62
Width	75
Height	57
<b>Appearance</b>	
ForeColor	0
BackColor	7
ColorMap	<None>
ColorData...	<None>
FillStyle	Opaque
<b>Format</b>	
<b>Runtime</b>	
Description	

En el apartado de "Position" se puede cambiar la ubicación del objeto en la pantalla como también su anchura y altura.

En "Appearance" se puede escoger el color del relleno y del contorno por separado, pero también podemos asignar un "Datapoint" (una variable) que defina el color y puede ir cambiando en el tiempo según el programa de autómatas.

## 2.4 Interacción con variables de programa

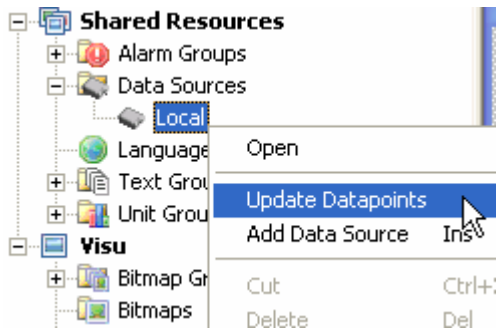
Visual Components puede leer y escribir todo tipo de variables del programa de autómatas, que pueden ser globales o exclusivas de una tarea.

Para ver las variables que interactúan con el programa de autómatas se puede hacer doble clic en "Data Sources - Local".



Desde aquí se puede ver las variables que interactúan con el autómatas y se pueden activar o desactivar variables o cambiar el tipo con que serán tratadas en Visual Components.

Cada vez que se añaden variables al programa de autómatas y el editor de Visual Components está abierto hay que clicar (con los Datapoints ya abiertos) en "Update Datapoints" en "Datasources".

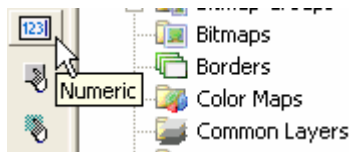


Es importante tener en cuenta que no puede usarse ninguna variable en Visual Components si no está llamada en algún sitio del programa del PLC. (Aunque solo ser Variable := Variable;)

Para conectar los "Datapoints" con elementos de programa se muestra un ejemplo con un cuadro numérico.

Previamente se crea una variable en una Tarea llamada numérico y si Visual Components está abierto clicamos en "Update Datapoints".

Ahora ya se puede crear el cuadro numérico clicando en su icono.

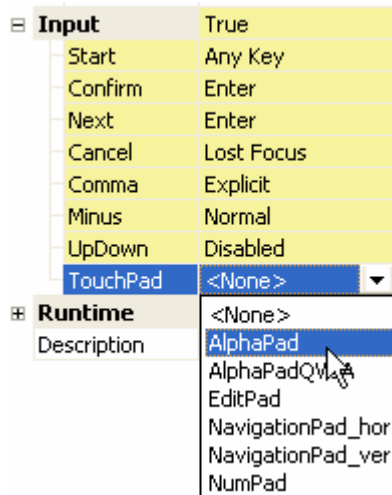


En las propiedades se ve que son distintas que las del cuadrado.

Name	Numeric_1
StyleClass	Output
<b>Position</b>	
<b>Appearance</b>	
<b>Format</b>	
<b>Alignment</b>	
MinInteger...	1
AddFractio...	0
AddFractio...	<None>
<b>UnitText</b>	None
<b>Value</b>	Standard
Datapoint	<None>
MinValue	<None>
MinDatapoint	<None>
MaxValue	<None>
MaxDatap...	<None>
TeachData...	<None>
Simulation...	0
<b>Input</b>	False
<b>Runtime</b>	
Description	

- **AddFractionDigits:** número de decimales que se quieren mostrar.
- **Datapoint:** aquí se puede escoger la variable a la que hará referencia el cuadro numérico. Clicando en el cuadrado de la derecha nos muestra todas las variables disponibles.
- **Min,MaxValue:** valor Máximo y mínimo que se le podrá escribir al cuadro numérico.
- **Min,MaxDatapoint:** variable que nos indica el valor mínimo y máximo y que puede cambiar por programa del PLC
- **Simulation Value:** es el valor que saldrá en el cuadro numérico cuando editemos la página en Visual Components para simular un valor.
- **Input:** si el input está en False, no se puede editar el numero en la pantalla, si se pone a True, se puede editar el cuadro numérico.

Para poder introducir valores en el cuadro numérico hay que poner el "Input" a True, entonces se despliega un submenú del menú de "Input". La opción más importante es "Touchpad", que permite escoger que teclado saldrá en pantalla cuando se pulse en el cuadro numérico.

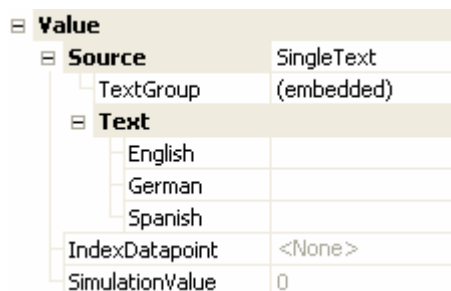


## 2.5 Insertar Textos, idiomas, fuentes

Para insertar un texto estático en AS hay que clicar en el icono de insertar texto.

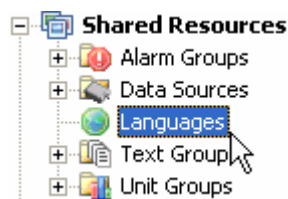


La diferencia básica en el árbol de propiedades se encuentra en "Value".

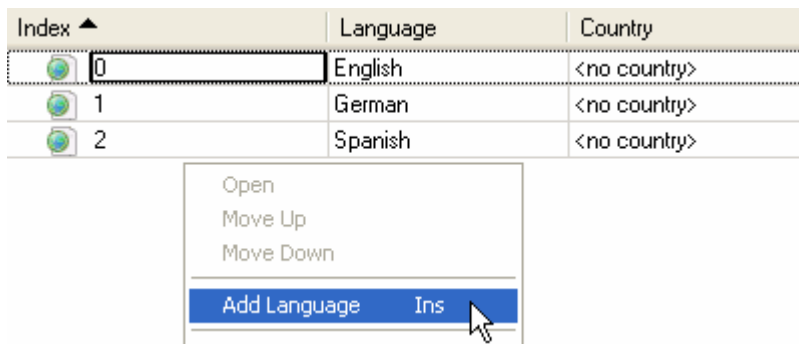


Para escribir un texto estándar hay que seleccionar "SingleText" en "Source" y en "Text" se puede escribir el texto en cualquiera de los idiomas que se definan en el proyecto.

La gestión de los idiomas en Automation Studio se realiza en el árbol de objetos clicando en "Languages".



Allí aparece la lista de lenguajes que se usan en el proyecto. Si se quiere añadir un lenguaje se pulsa "Insert" o con el botón derecho del ratón "Add Language".



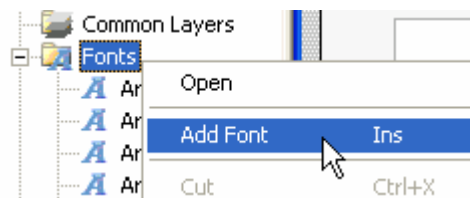


Una vez el lenguaje está añadido, aparecerá siempre en las propiedades de los textos. Si se quiere ver la visualización en un idioma u otro se puede cambiar la vista en la lista desplegable que está situada debajo del menú.



Este cambio de idioma solo es válido para la vista en el editor. En el siguiente capítulo se muestra como definir el idioma de arranque y cambiar de idioma por programa.

Para añadir fuentes al proyecto, hay que pulsar en "Add Font", después de pulsar con el botón derecho sobre "Fonts".



Al añadir una nueva fuente aparece una ventana con sus propiedades.

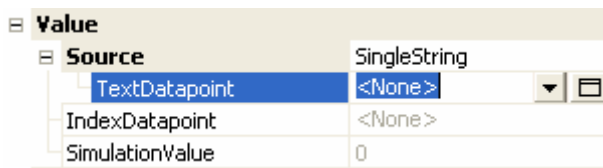
Language ▲	Font Face	Size	Bold	Italic
English	Arial	12	False	False
German	Arial	12	False	False
Spanish	Arial	12	False	False

Aquí se define el tipo de fuente, su tamaño y si es en negrita. Una fuente se define por cada idioma por separado ya que distintos idiomas pueden tener distintas fuentes, por ejemplo algunos idiomas necesitan fuentes unicode (Chino, Árabe...)

## 2.6 Grupos de texto, textos dinámicos

Aparte de introducir textos fijos como en el capítulo anterior, Visual Components permite variedad formas de cambiar textos dinámicamente. La primera forma, y más sencilla, es que un texto se corresponda con una variable tipo String del programa de PLC, así en la pantalla se verá un texto con el valor que tenga la variable.

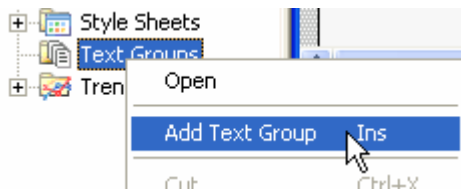
Para hacerlo, seguimos los mismos pasos que con un texto normal, pero en vez de seleccionar "SingleText", se selecciona "SingleString".



Después, simplemente hay que seleccionar una variable de programa de tipo String y se mostrará en la pantalla.

Otra opción para crear textos dinámicos es usar los grupos de texto. Los grupos de texto consisten en una lista de textos con un índice. Al poner un texto en pantalla se puede seleccionar un grupo de texto y asignar una variable como índice del grupo de texto, entonces el texto que se ve en pantalla irá cambiando según cambie la variable del índice.

Antes de nada, hay que crear un grupo de texto pulsando "Add TextGroup" en el árbol de objetos.



Al crear el nuevo grupo de texto, aparece la ventana con el índice y el texto separado por cada idioma. A modo de ejemplo se crea un grupo de texto llamado EstadoMaquina.

Index ▲	English	German	Spanish
0			Maquina Parada
1			Homing Motores
2			Funcionamiento Manual
3			Funcionamiento Automático
4			Parada controlada
5			Paro Emergencia

Ahora cambiando la variable del programa de PLC VarEstadoMaquina se quiere que un texto nos muestre siempre el estado de la máquina.

Para configurarlo se lleva a cabo el procedimiento habitual para añadir un texto, en "Source" se selecciona "MultipleTexts" y aparecerán las nuevas propiedades.

Value	
Source	MultipleTexts
TextGroup	EstadoMaquina
TextIndexOffset	0 : Maquina Parada
IndexDatapoint	Local.VarEstadoMaquina
SimulationValue	0

En "TextGoup" se ha introducido el grupo de texto y se ha escogido en "IndexDatapoint" la variable VarEstadoMaquina.

Si se cambia el "SimulationValue", se puede ver en el editor como el texto cambia de un texto a otro.

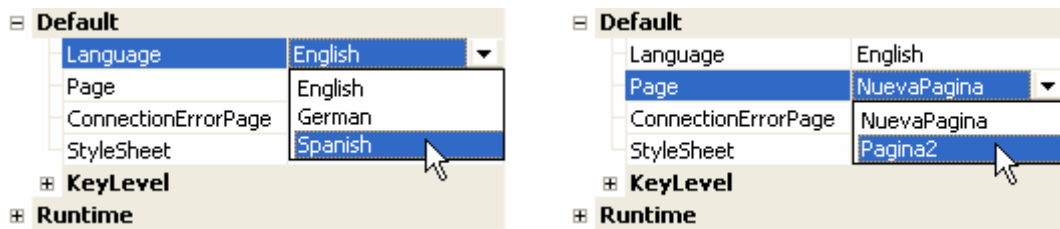
La última opción es el "MultipleStrings", que es parecida a la anterior, pero los textos no se leen de un grupo de texto sino de una variable tipo array de Strings.

## 2.7 Cambio del lenguaje y página de inicio o por programa

El lenguaje y página de inicio, con el que arranca la visualización en el panel, se define pulsando sobre el nombre de la visualización en el árbol de objetos.



Allí aparecen las propiedades, en "Default - Language", "Page", se puede escoger el lenguaje y página que se mostrarán en el panel cuando arranque.

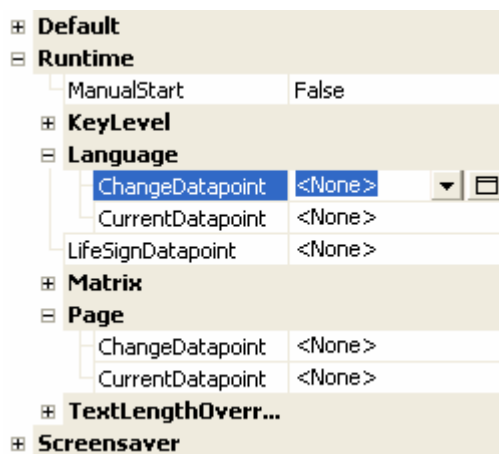


Otra opción para cambiar de lenguaje o de página es hacer el cambio con una variable desde el programa de PLC.

El PLC interactúa con dos variables:

- **CurrentPage, CurrentLanguage:** la variable que se asigne aquí, indicará el índice de lenguaje y página que se tenga actualmente.
- **ChangeLanguage, ChangePage:** esta variable es la que se puede escribir para cambiar de lenguaje o página. La variable siempre tiene el valor 65535 hasta que se escribe el número de la página a la que se quiere cambiar. Cuando se ha cambiado de página, la variable vuelve al valor 65535.

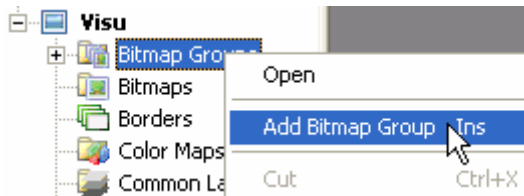
La asignación de esas variables se encuentra en "Runtime - Language, Page"



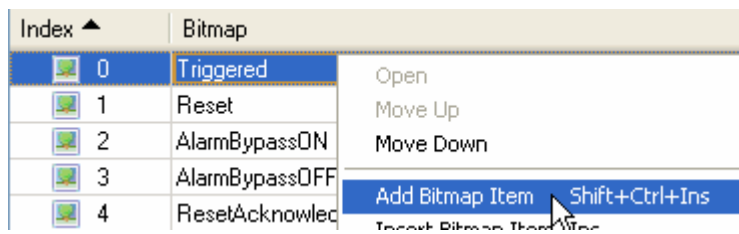
## 2.8 Bitmaps

Al igual que los textos, los bitmaps también se pueden usar como bitmaps solos o como grupo de bitmaps que cambian con un índice. Por ejemplo, el estado de la máquina también se puede representar con un bitmap que irá cambiando según la variable VarEstadoMaquina.

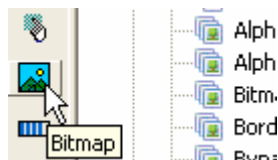
El primer paso será añadir un grupo de bitmaps en el árbol de objetos.



En el grupo de bitmaps se puede añadir los que se quieran, pulsando "Add Bitmap Item". Pueden ser independientes o dependientes del idioma, seleccionando "Language Dependent" en las propiedades.

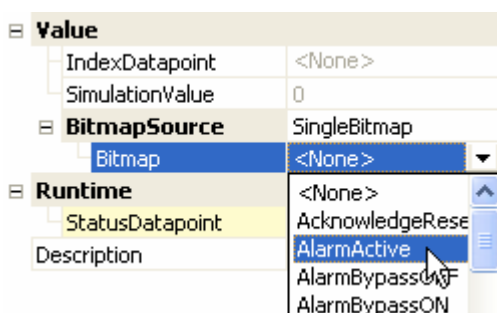


Una vez está creado el grupo de bitmaps, se puede agregar un bitmap a la página pulsando el icono correspondiente.



En las propiedades también, al igual que en grupos de texto, se puede escoger entre "SingleBitmap", "BitmapGroup" o "MultipleBitmaps".

En "SingleBitmaps" se selecciona simplemente el Bitmap que se quiera mostrar en pantalla.



En "GroupBitmap" se selecciona el grupo de Bitmaps y se puede escoger un Bitmap dentro de este grupo, aunque sigue siendo un Bitmap estático.

<b>Value</b>	
IndexDatapoint	<None>
SimulationValue	0
<b>BitmapSource</b>	
BitmapGroup	AlarmEvent
GroupItem	<None>
<b>Runtime</b>	
StatusDatapoint	0 : Triggered
Description	1 : Reset 2 : AlarmBypassON 3 : AlarmBypassOFF

Finalmente, en "MultipleBitmaps" se puede escoger el grupo de Bitmaps y una variable como índice para cambiar el Bitmap según el programa de PLC.

<b>Value</b>	
IndexDatapoint	Local.VarEstadoMaquina
SimulationValue	0
<b>BitmapSource</b>	
BitmapGroup	AlarmEvent
BitmapIndexOffset	0 : Triggered

Si se cambia el "SimulationValue", el Bitmap tiene que ir cambiando en la página.

Con los grupos de Bitmaps y el índice como variable, se puede hacer una animación en un Panel.

Los Bitmaps pueden tener el fondo que llevan, o se puede asignar un color como transparente.

Con doble clic en Bitmaps se despliega una lista con todos los Bitmaps que hay en el proyecto, en las propiedades de un bitmap encontramos la propiedad "Transparent" y "TransparentColor", que es el color que se hará transparente.

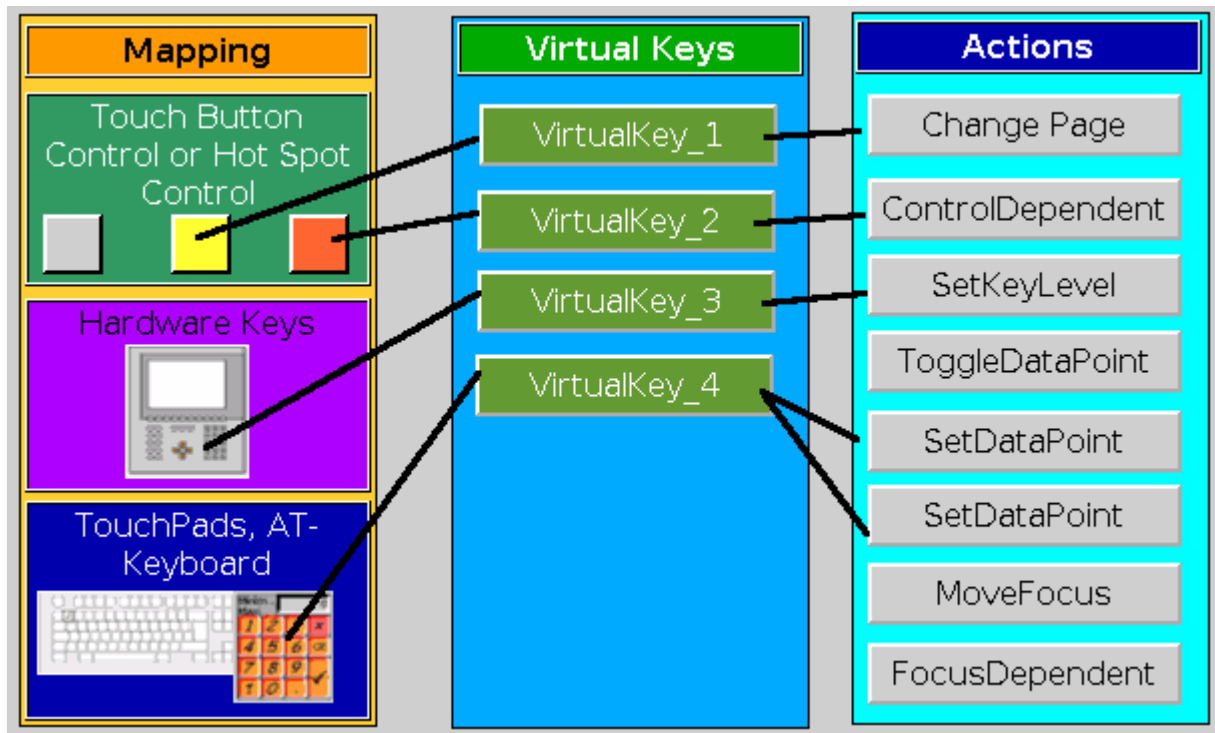
<b>Position</b>	
SourceType	BMP
TargetFormat	Raw8Bit
Transparent	True
TransparentColor	■ 148
FillAreas	0

## 2.9 Botones

El funcionamiento de las teclas y botones en los paneles de B&R funciona usando teclas virtuales (“Virtual Keys”).

Un o varios elementos que pueden ser pulsados, ya sean pulsadores de hardware del panel o botones en la pantalla llevan asociada una Virtual Key, que consiste en una o varias acciones a realizar.

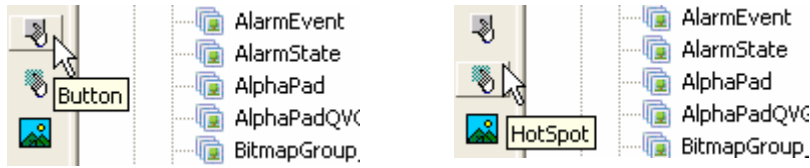
El funcionamiento de las Virtual Keys se puede ver representado en el siguiente esquema.



Por cada botón que se inserta en la visualización, hay que crear primero una “Virtual Key” y después asignarla al botón y definir las acciones.

Para ver un buen ejemplo de todas las acciones posibles con Visual Components se puede probar el proyecto de ejemplo de visualización Demo01.pgp que se encuentra en el directorio de Samples de Automation Studio (c:\Program files\brautomation\AS...\Samples\).

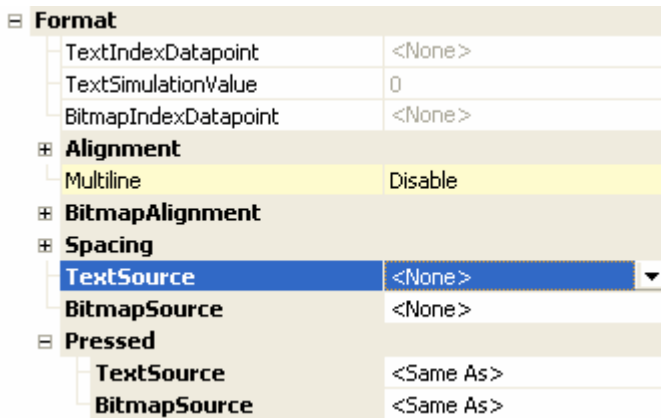
Se pueden insertar dos tipos de botones: el botón estándar y el Hotspot. La diferencia entre ellos es que el Hotspot es transparente y no se ve en la pantalla, o sea que se puede poner encima de la imagen.



Una vez añadido el botón a parte de definir las propiedades normales de color y posición, se puede definir el texto y/o imagen del botón.

Se puede definir texto e imágenes distintas según el botón esté presionado o no cambiando las propiedades de "Pressed". En el caso que lo queramos igual, debemos dejar "<Same As>" en esas casillas.

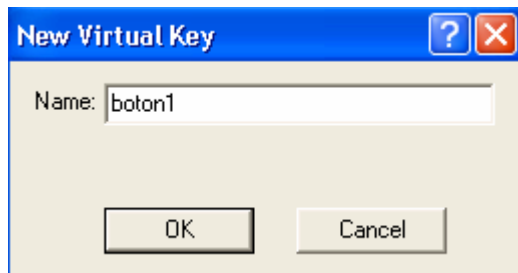
Las propiedades de Texto y Bitmaps (que vemos en "TextSource" y "Bitmap Source") son idénticas que en textos y Bitmaps descritas anteriormente, incluyendo la posibilidad de poner índices de "TextGroups" o "BitmapGroups".



Después de definir la apariencia del botón, hay que definir el comportamiento. El primer paso es crear una "Virtual Key" escribiendo un nombre en la casilla de "Keys". (Si la "Virtual Key" ya está definida, la podemos seleccionar en la lista desplegable)

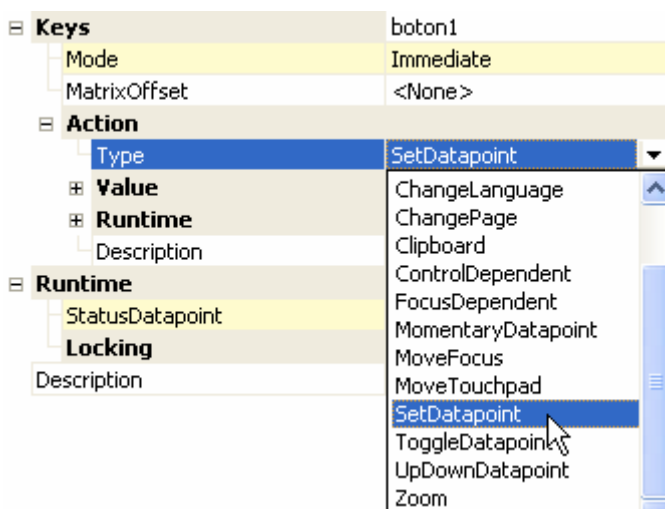


Si la "Virtual Key" es nueva aparecerá un dialogo donde se puede aceptar el nombre.

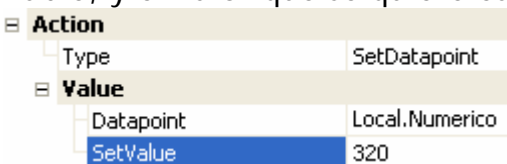


En este momento aparece una nueva opción en las propiedades llamada "Action". Aquí se puede escoger entre los distintos tipos de acciones disponibles.

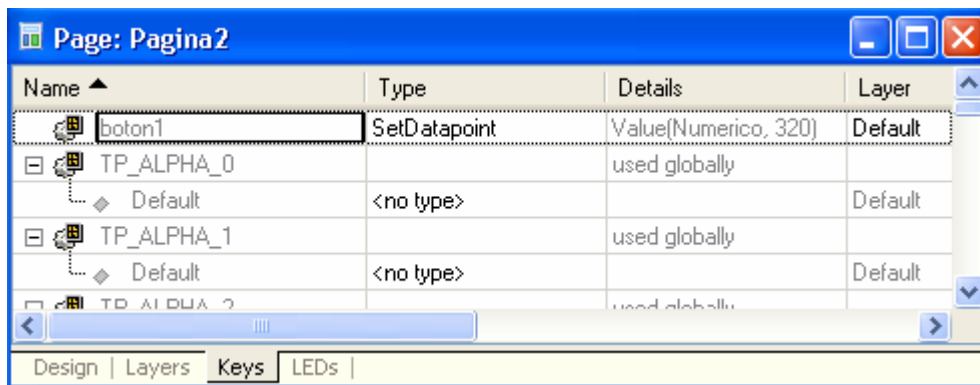




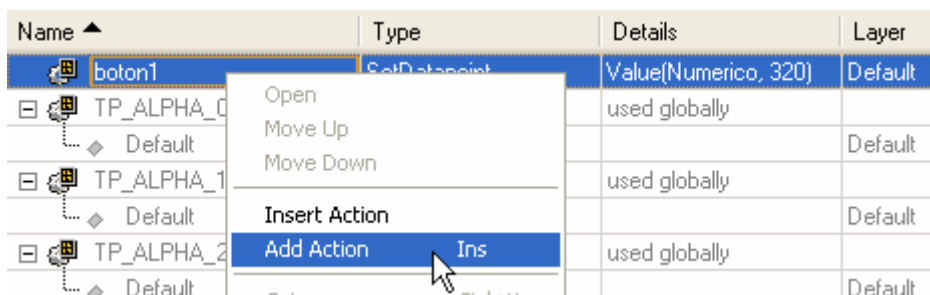
Cada tipo de acción lleva asociadas sus propiedades. En el cambio de página se elige que a que página se desea cambiar, en el cambio de idioma hay que elegir el idioma al que se desea cambiar... por ejemplo en el "SetDatapoint", pide una variable, y el valor que se quiere escribir en ella.



Para insertar más de una acción en una tecla virtual, hay que seleccionar la pestaña de "Keys" en la parte inferior de la página que es está editando.

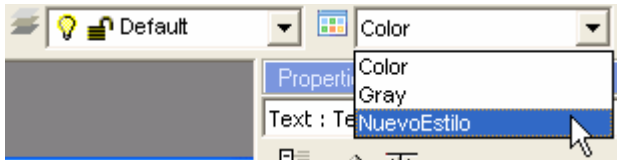


Aquí se puede añadir una acción en la tecla virtual boton1 pulsando "Add Action" con el botón derecho del mouse.

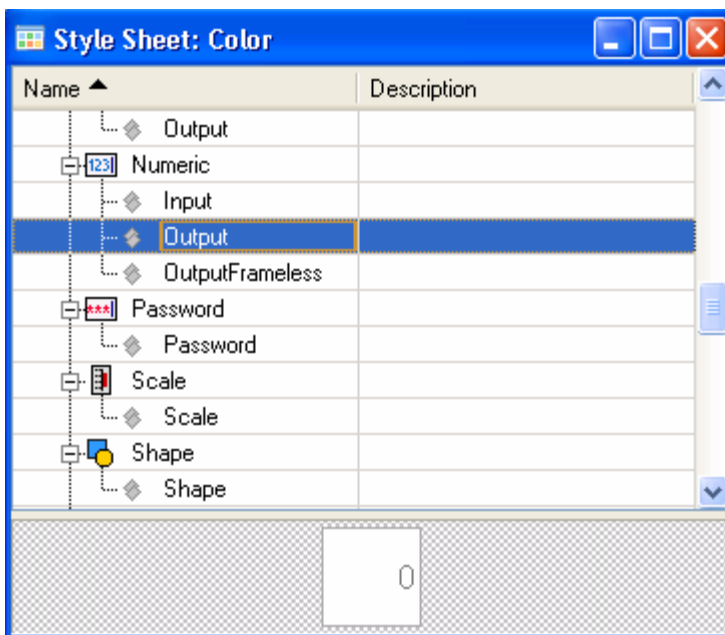


### 2.10 Estilos

En Visual Components, se puede definir el color de cada botón o forma que hay en la pantalla, pero también se pueden crear estilos, de forma que todos los elementos que se vayan añadiendo tengan el mismo color y aspecto. Se puede cambiar de estilo en la parte superior del editor de Visual Components.



Para crear o editar los estilos se puede seleccionar en "StyleSheets" en el árbol de objetos.

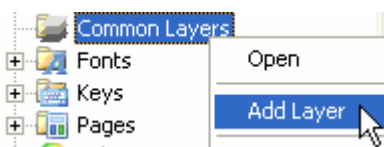


Por cada tipo de objeto se pueden editar las propiedades de que vienen por defecto definidas en el estilo.

### 2.11 Capas

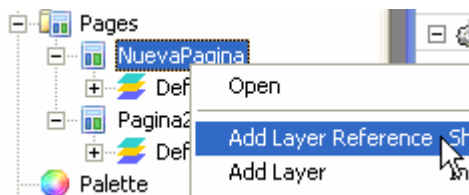
Para trabajar con elementos que se repiten en distintas páginas o para tener diálogos que pueden aparecer en cualquier momento, se puede trabajar con capas.

Para crear una capa, hay que seleccionar "AddLayer" en "Common Layers".



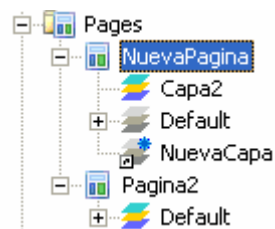
Visual Components creará una ventana exactamente igual a una página donde se pueden añadir los objetos que se deseen.

Una vez creada la capa, puede añadirse a las páginas que se desee pulsando en "Add Layer Reference".



Entonces, la capa que ya teníamos dibujada en "Common Layers" quedará añadida en la página.

También se pueden crear capas dentro de una misma página, pulsando "Add Layer".



Cada capa que se añade a la página se le asigna un orden que se puede modificar con la propiedad "Z-Order".

Name	Capa2
Active	False
Z-Order	2
Editing mode	Normal
Visibility mode	Normal

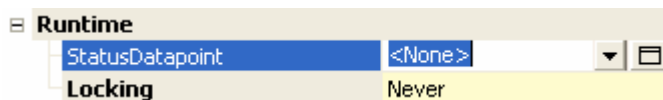
Para poder trabajar cómodamente en una página, se puede esconder o desactivar las capas por separado en el desplegable bajo el menú.



Para activar o desactivar capas por programa se dispone del "Status Datapoint", que se explica en el siguiente capítulo.

## 2.12 StatusDatapoint, mostrar o esconder objetos

El "StatusDatapoint" se encuentra en todos los objetos o en capas definidos en la visualización. Todos los objetos lo tienen en sus propiedades "Runtime - StatusDatapoint".



El "StatusDatapoint" es una variable entera donde cada bit tiene un significado.

Por ejemplo,

Bit 0: esconder control. Si se pone este Bit a 1, entonces el control desaparece.

Bit 1: bloquear control. Si se pone este Bit a 1, entonces no se puede pulsar o introducir datos en el control aunque se sigue viendo en la pantalla.

La descripción de todos los Bits por cada tipo de objeto se puede encontrar en la ayuda.

Si se quiere que aparezca un dialogo en la pantalla, se crea una capa con el dialogo y se le asigna un "StatusDatapoint". Cuando se ponga el "StatusDatapoint" a uno, la capa desaparecerá y cuando se ponga a 0 aparecerá encima de la página actual.

### 2.13 Passwords

En Visual Components se pueden configurar varios niveles de password, para que el usuario pueda acceder a unos controles o a otros.

Con un control de entrada de password, el plc se podrá en el nivel de password correspondiente.

Hay que crear tres elementos:

- **Variable Nivel de password.** Es una variable entera que contiene el nivel de password y puede ser cambiada por el control de entrada de password o por programa de PLC.
- **Grupo de texto passwords.** Es un grupo de texto que contiene el password correspondiente a cada nivel.
- **Grupo de texto Nivel de passwords.** Es un grupo de texto que contiene el texto relacionado con cada nivel de password.

Primero hay que crear los grupos de texto de passwords y de niveles de password.

Index ▲	English	German	Spanish
0			
1	195489		
2	M6tG23Xdt		
3	98hgsrli-jghu12		

Después hay que crear el grupo de texto con los nombres de los niveles de password.

Index ▲	English	German	Spanish
0	Usuario		
1	Usuario Avanzado		
2	Mantenimiento		
3	Administrador		

También se creará una variable entera llamada NivelPasswordActual.

Ahora ya se puede añadir el control de entrada de password,



que se relacionará con el grupo de texto de password y con la variable NivelPasswordActual. También se puede poner que el carácter q aparece en pantalla sea el asterisco.

<b>Value</b>	
<b>Source</b>	MultipleTexts
TextGroup	Passwords
TextIndexOffset	0 ;
<b>Text</b>	
LevelDatapoint	Local.NivelPasswordActual
MaxChar	<None>
MaxLevel	3
SimulationValue	0
<b>Input</b>	
Start	Any Key
Confirm	Enter
Next	Enter
Cancel	Lost Focus
Mode	Normal
TouchPad	AlphaPadQVGA
<b>Format</b>	

Para ver en pantalla cual es el nivel actual, se crea un texto en la pantalla que indicará el nivel actual.

El texto se asocia a la variable NivelPasswordActual.

<b>Value</b>	
<b>Source</b>	MultipleTexts
TextGroup	NivelPassword
TextIndexOffset	0 ; Usuario
IndexDatapoint	Local.NivelPasswordActual
SimulationValue	0

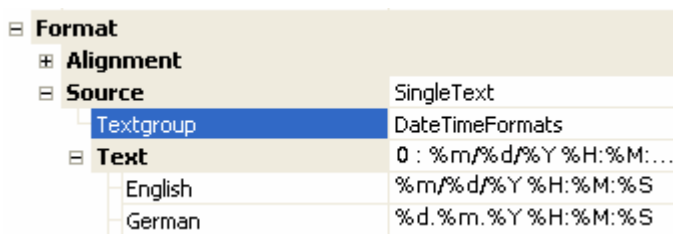
## 2.14 Fecha y hora

Hay muchas formas de representar la hora, incluso distintas según cada idioma. Visual Components permite representar la hora de forma variable con el objeto de fecha, que se encuentra en la barra de objetos.



El objeto de fecha es un cuadro de texto que muestra una variable tipo tiempo, pero la muestra según un texto codificado donde se puede escribir la fecha u hora de distintas maneras según el idioma.

En el texto, que se encuentra en las propiedades del objeto,



se escriben las partes de la fecha/hora que se desee (Día, hora, minuto, año...) con un código en el orden deseado y con cualquier carácter de separador. Los códigos se pueden ver en esta tabla que se encuentra en la ayuda.

Format	Meaning
%a	Shortened name of the weekday
%A	Complete name of the weekday
%b	Shortened name of the month
%B	Complete name of the month
%d	2-digit day of the month (01-31)
%H	Hour in 24-hour format (00-23)
%I	Hour in 12-hour format (01-12)
%j	Day of the year (001-366)
%m	Month (01-12)
%M	Minute (00-59)
%p	"AM" or "PM" text (for 12-hour time format)
%S	Seconds (00-59)
%U	Week of the year (Sunday is the first day of the week) (00-53)
%w	Weekday (0-6, Sunday is 0)
%W	Week of the year (Monday is the first day of the week) (00-53)
%y	Year without the century specification (00-99)

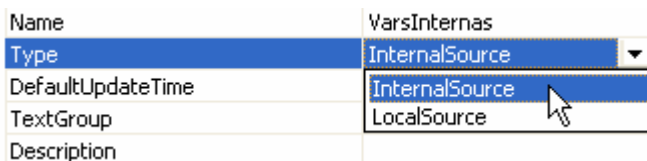
Format	Meaning
% Y	Year including the century specification
%%	%

Por ejemplo, si se quiere escribir una fecha en el formato 23 - 03 - 2007 - 11:53, el texto correspondiente sería:  
 %d - %m - %Y - %H:%M.

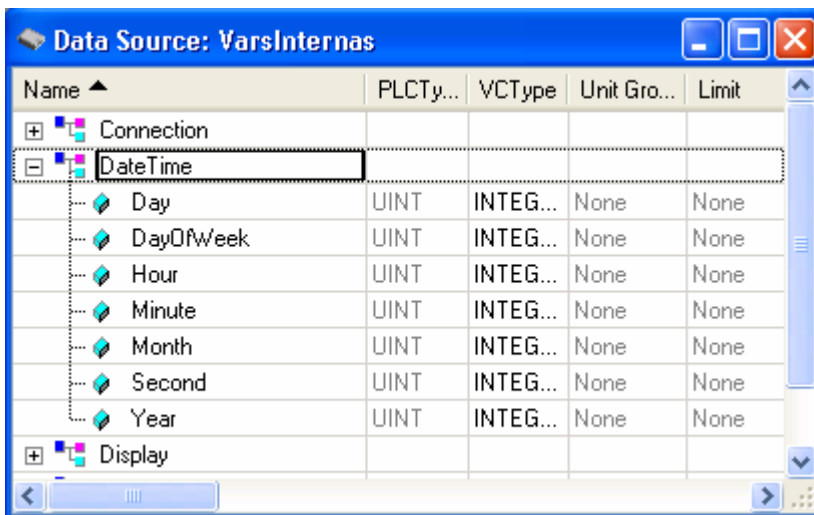
Es interesante también tener la posibilidad de cambiar la hora del autómata, ya que si se agota la batería se perderá la hora. Para eso hay que escribir la fecha y hora en las variables internas del equipo. En el árbol de objetos, en "Datasources", se añade un nuevo "Datasource" pulsando en "Add Datasource".



En las propiedades del nuevo "Datasource", se selecciona tipo "Internal".



Ahora que ya está creado, hay que clicar en "Update Datapoints" y aparecerán las variables internas del panel.



Esas variables pueden ser usadas en un cuadro numérico donde se podrá introducir la nueva fecha y hora del panel.

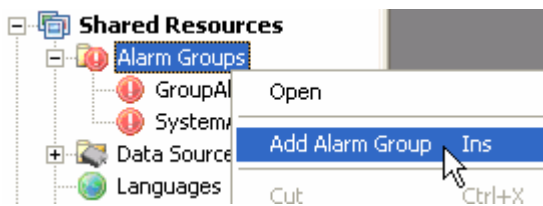


### 2.15 Gestión de alarmas

Para la gestión de alarmas, Automation Studio proporciona un sistema de alarmas que se muestra las alarmas activas o históricas a través de un vector de alarmas.

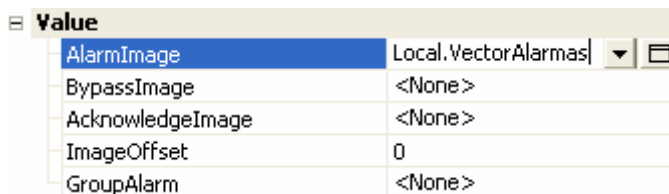
El primer paso es crear un vector de alarmas en una tarea del PLC. El vector de alarmas consiste en un array, donde cada elemento se pone a 1 si existe esa alarma.

El segundo elemento que hay que crear, es un grupo de alarmas con los textos de las alarmas.

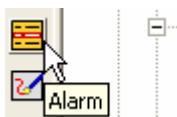


En este grupo de alarmas es donde hay que escribir los textos correspondientes a cada alarma con el índice que corresponda del vector de alarmas. Cuando el valor situado en el índice del vector de alarmas número "i" se ponga a 1, el texto del grupo de alarmas con índice "i" aparecerá en pantalla.

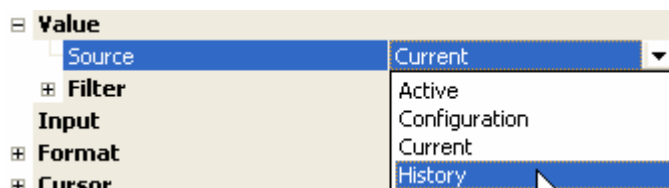
Después de crear el grupo hay que asociarlo al vector de alarmas (variable tipo BOOL[n]), en el ejemplo se llama VectorAlarmas. Esa asociación se pone en las propiedades del grupo de alarmas.



Para mostrar las alarmas en pantalla, hay que añadir un objeto de alarmas en la página apropiada pulsando sobre el icono de "Alarm".



El las propiedades del objeto de alarmas nos permite escoger entre cuatro tipos de representación de alarmas.



- **Active:** muestra una sola alarma que es la última que se ha producido.
- **Configuration:** parecida a current, pero donde se puede hacer bypass.
- **Current:** una lista con las alarmas activas en el momento.
- **History:** muestra el histórico de alarmas, cuando han aparecido o se han aceptado.

Otra opción importante son los campos que se verán representados en el objeto de alarmas. Se puede escoger si sale la hora o la fecha, entre otras cosas, y escoger el orden y tamaño de cada columna.

Columns	
GroupNumber	Column 1
DateTime	Column 2
AlarmNumber	Column 3
AlarmText	Column 4
Priority	<Not Used>
AlarmState	<Not Used>
AcknowledgeState	<Not Used>
BypassState	<Not Used>
Event	<Not Used>

Si se pulsa sobre "AlarmGroups" se puede escoger en las propiedades si se desea que el histórico de alarmas se mantenga aunque se pare el panel y el número de entradas que van a guardarse en el PLC.

History	
Length	1000
Persistent	False

### 2.16 Uso de unidades

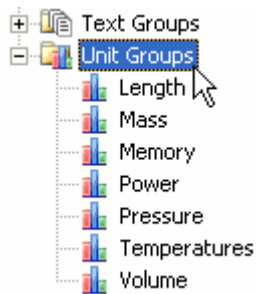
Visual Components permite el uso de variables que se gestionan como un valor que tiene unidades. Eso significa que existe una variable del programa de PLC que corresponde a una magnitud física pero en la pantalla se quiere representar en uno o en varios sistemas de unidades.

Por ejemplo, una temperatura que en el PLC se mide en décimas de grado, pero en la pantalla queremos mostrar el valor en grados Centígrados o Fahrenheit.

Para no tener que crear tres variables calculadas y relacionarlas en Visual Components, se puede definir la variable como perteneciente a un grupo de unidades y que el PLC lo gestione automáticamente.

Se pueden crear grupos de unidades o usar los que ya lleva predefinidos Visual Components.

Se encuentran en "Unit Groups".



Por cada grupo de unidad se puede tener varias unidades y se puede definir el escalado a través de una recta, de dos puntos o de forma dinámica.

Index	Name	Unit Abbreviation	Unit Description	Default Precision	Descr
0	PLCUnit			2	
1	Kelvin	K	kelvin	2	
2	Celsius	°C	celsius	2	
3	Fahrenheit	°F	fahrenheit	2	

Conversion Test

Internal value : 298

Scaled value : 24.85

Conversion parameters

Scaled = 1 x internal + -273.1

Scaled

30

24.85

0

298

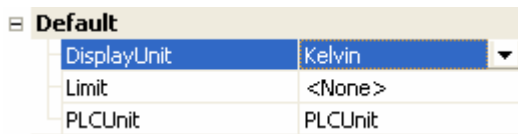
35E

Internal

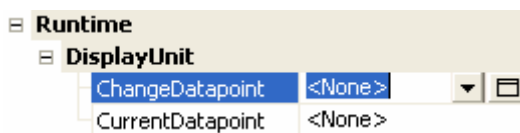
Cuando se introducen los parámetros de la conversión, se pueden testear los valores en el "Conversion Test" para comprobar si la conversión es correcta. También es interesante definir la precisión por defecto que pone los decimales automáticamente para cada unidad. Por ejemplo los grados centígrados se pueden poner con un decimal, en cambio los grados Kelvin pueden ir sin decimales.

El texto que se ponga en "Unit Abbreviation" y en "Unit Description" puede aparecer automáticamente al lado de un cuadro numérico para mostrar con que unidad estamos trabajando.

Si se clicca en las propiedades del grupo de unidades, se puede escoger cual es la unidad por defecto que se mostrará en la visualización.



También en las propiedades se puede configurar las variables que se van a usar para cambiar en funcionamiento la unidad que va a usarse.



Cuando el grupo de unidades está creado, hay que asociar la variable al grupo de unidades, que se realiza en los "Data Sources - Local".

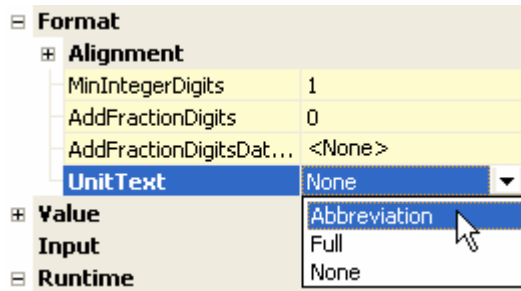
Name	PLCTy...	VCType	Unit Group / Sub...	Limit	PLCUnit
Programa1					
SensorTemperatura	UINT	INTEGER	None	None	None
Timer		<None>			
VarEstadoMaquina	INT	CHAR	None	None	None
Variable	INT	INTEGER	None	None	None
		SCALED			

Hay que seleccionar que la variable es de tipo "Scaled" para que se pueda seleccionar a que grupo de unidades pertenece el valor.

También se pueden seleccionar límites para esa variable y cual es la unidad que mide el PLC internamente.

Al crear un cuadro numérico en una página automáticamente nos va a usar las unidades que estén definidas con la precisión también definida para la unidad actual.

La única propiedad que se puede cambiar es el texto que aparece al lado del número en el cuadro numérico.



Si se selecciona "Abbreviaton", aparecerá el texto abreviado al lado del número. Si se selecciona Full, aparecerá el nombre entero.

35.70°C

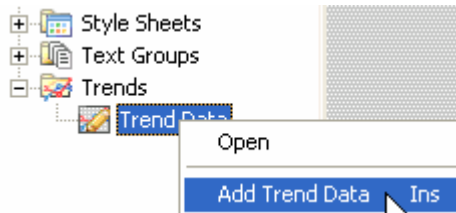
96.26°F

### 2.17 Gráficas de tendencias o Trends

En la versión 3 de Automation Studio, se pueden añadir gráficos de tendencias directamente en Visual Components.

El primer paso para crear una gráfica, consiste en añadir uno o varios datos de "Trends" que se corresponderán con las distintas líneas de un gráfico.

Los datos de los "Trends" los encontramos junto con los objetos de visualización.



Para el ejemplo se añaden dos "Trends" para realizar un gráfico con dos líneas.

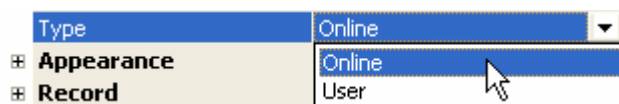
Index ▲	Name	Type
0	TrendData_1	Online
1	TrendData_2	Online

En las propiedades de cada "TrendData" se debe escoger la variable cuyo valor se va a representar en la gráfica.

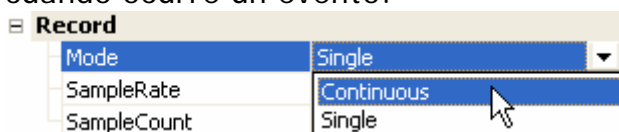
Value	
Datapoint	Local.SensorTemperatura
MinValue	0
MinDatapoint	<None>
MaxValue	100
MaxDatapoint	<None>
SimulationValue	Random

Otras propiedades importantes en la definición de los datos del "Trend" son:

- Type: "Online" o "User". Si se escoge el tipo "Online", Visual Components va a graficar la variable que se haya introducido con su valor actual. En el caso de "User", se grafica un vector de valores.

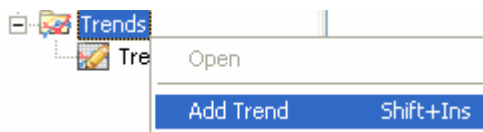


- Record Mode: "Single" o "Continuous". Permite escoger entre almacenamiento continuo que se va repitiendo, o un almacenamiento solo cuando ocurre un evento.



Después de configurar los datos del "Trend", se debe configurar el entorno del "Trend".

Se añade un nuevo "Trend" pulsando el botón derecho sobre el menú de Trends.



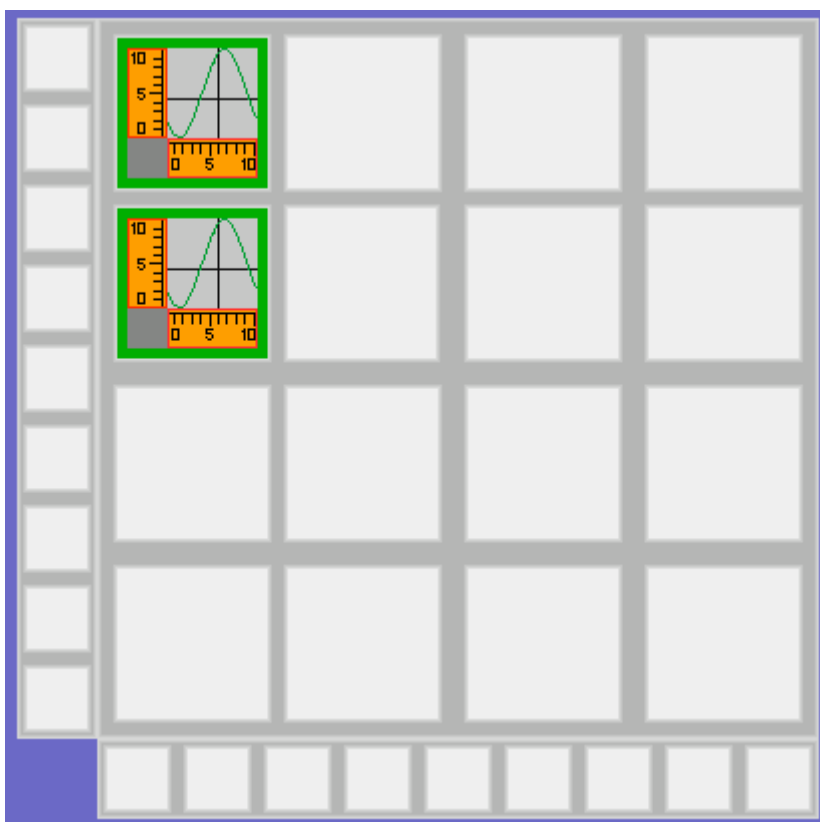
Al crear el nuevo "Trend", se forma un entorno que consiste en un área para "Trends" y dos espacios para colocar los ejes. Ahora se pueden arrastrar una o varias "TrendCurve" al entorno desde el icono de gráfica.



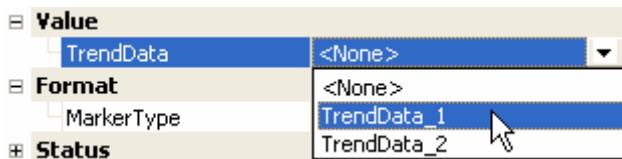
Dependiendo de si se quiere usar uno o varios ejes de valores en la gráfica se pueden arrastrar más "TrendScaleContainers".



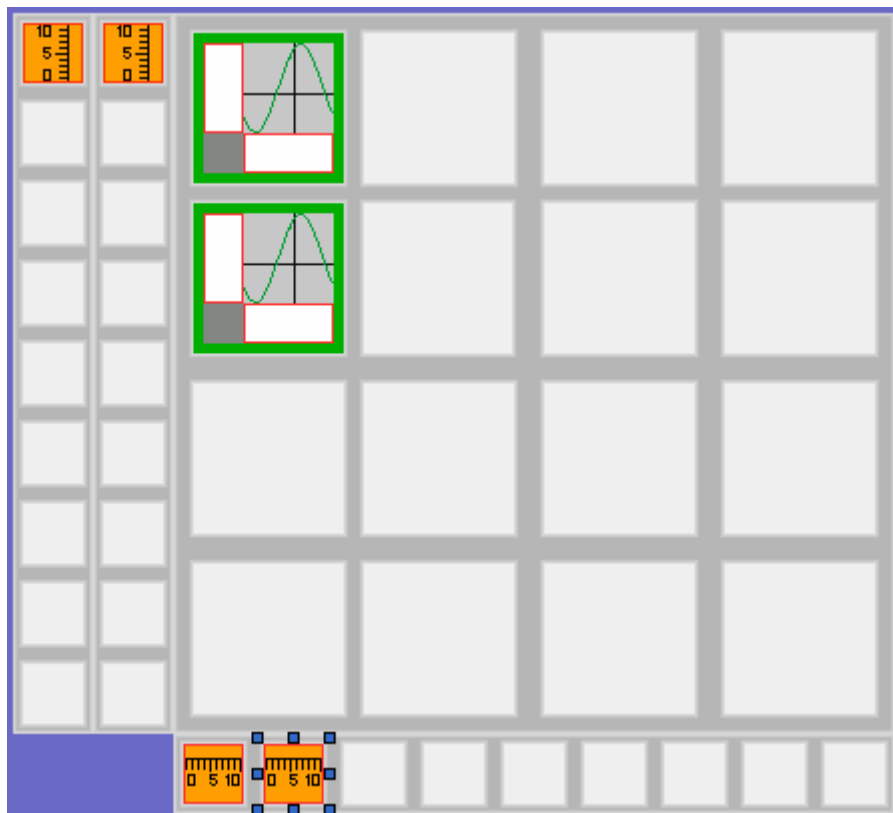
En este ejemplo se han añadido dos "Trend" y dos ejes de valores para el eje y.



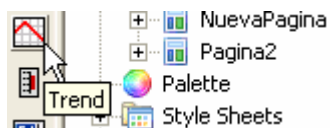
En las propiedades de cada "Trend" que se ha añadido en el área de "Trends", se debe asignar el "TrendData" que le corresponda.



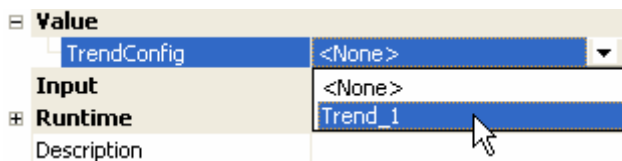
Para relacionar los ejes con los "Trends" hay que arrastrar las escalas a los espacios correspondientes. Los dos ejes X se encuentran en la misma escala, y los ejes Y están en escalas distintas.



Ahora ya se puede añadir el "Trend" a la página correspondiente arrastrando el icono de "Trend".



En las propiedades del "Trend" se selecciona cual de los que se hayan creado se quiere mostrar.





### 3. GLOSARIO

- AS: Automation Studio
- SG3: Generación 3ª de las CPUs de B&R. La arquitectura se basa en procesador Motorola y no disponen de Compact Flash. En la actualidad sólo algunas CPUs son SG3.
- SGC: Generación Compacta de las CPUs de B&R. También basadas en microprocesador y sin Compact Flash. Esa generación solo se usa en las CPUs compactas de B&R.
- SG4: Generación 4ª de las CPUs de B&R. La arquitectura se basa en procesador Intel (arquitectura PC) y llevan Compact Flash. Son la gran mayoría de las CPUs de B&R y permiten todos los servicios Tcp/Ip.
- X2X: Backplane (o se puede llamar Bus) donde se conectan las entradas y salidas ya sean físicamente enganchadas al PLC o separadas por cable de un máximo de 100m entre islas de E/S.
- ST: Lenguaje de programación Texto Estructurado, pertenece al IEC 61131.
- SFC: Sequential Function Chart. Lenguaje de programación gráfico pensado para tener secuencias de estados. Pertenece al estándar IEC 61131, también llamado Grafset.
- C: Lenguaje de programación de alto nivel. Usado ampliamente en PCs, se puede usar como lenguaje de programación de PLCs en B&R.
- AR: Automation Runtime. Es el sistema operativo que llevan las CPUs dentro. Es un sistema operativo propio para los equipos de B&R que soporta "Real Time".
- AR000: Automation Runtime que funciona en el mismo PC donde se usa Automation Studio. Está orientado solo a simular la aplicación de control y también de visualización (usando VNC).
- VNC: Virtual Network Computing. Estándar de control remoto de PCs que permite transferir por ethernet la pantalla del PC con su teclado y ratón. Las CPUs SG4 de B&R incorporan servidor de VNC.
- Visual Components: Editor de visualización incluido en Automation Studio.
- NC Software: Elementos de software incluidos en Automation Studio dedicados al control de Motion.
- IF: Interface, vía de comunicación que tiene una CPU.

Bernecker + Rainer Automatización Industrial S.L.U.  
Pol.Ind.Circuit de Catalunya  
Can Cabanyes, 88  
08400 - Granollers  
Tel: +34 93 5689965  
Fax: +34 93 5689973  
[www.br-automation.com](http://www.br-automation.com)

©2005 by B&R. Alle Rechte vorbehalten.  
Alle eingetragenen Warenzeichen sind Eigentum der jeweiligen Firma.  
Technische Änderungen vorbehalten.

140 offices in more than 55 countries - [www.br-automation.com/contact](http://www.br-automation.com/contact)



Australia • Austria • Belarus • Belgium • Brazil • Bulgaria • Canada • Chile • China • Croatia • Cyprus • Czech Republic  
Denmark • Egypt • Emirates • Finland • France • Germany • Greece • Hungary • India • Indonesia • Ireland • Israel • Italy • Korea  
Kyrgyzstan • Malaysia • Mexico • The Netherlands • Norway • Pakistan • Poland • Portugal • Romania • Russia • Singapore  
Slovakia • Slovenia • South Africa • Spain • Sweden • Switzerland • Taiwan • Thailand • Turkey • Ukraine • United Kingdom • USA