

10

capítulo

Tratamiento de datos y software

- *Presentación de ejemplos concretos de arquitecturas (esquemas, productos, y softwares)*
- *Presentación de un sistema de control híbrido sg^2*



10.1	Definición
10.2	Introducción
10.3	Programación, configuración y lenguajes
10.4	Categorías de aplicación

1
2
3
4
5
6
7
8
9
10
11
12
M

10. Tratamiento de datos y software

Este capítulo trata de la función de tratamiento vista en el primer capítulo e incluye una descripción del software industrial y su interacción con los procesos del sistema de automatización. A diferencia de otros capítulos, no entraremos en detalles en lo que se refiere a sistemas, lenguajes de programación, etc. Existen un gran número de publicaciones a disposición del lector.

10.1 Definición

"Automáta programable industrial" (en inglés PLC, "Programmable Logic Controller") es el nombre que recibe un dispositivo electrónico programable para controlar sistemas industriales a través de un tratamiento secuencial de datos.

Envía órdenes a los operadores (Parte Operativa o PO) basándose en datos de entrada (sensores), consignas y un software.

Un autómatas programable es un dispositivo similar a un PC y se utiliza para automatizar procesos como, por ejemplo, controlar las máquinas de una cadena de montaje en una fábrica. Donde antes se empleaban complicados sistemas electromecánicos, ahora con un simple autómatas basta. Las personas encargadas de programar éstos dispositivos son los ingenieros de sistemas de automatización.

10.2 Introducción

Los autómatas programables nacieron en la década de los 70. En un principio, fueron diseñados para asegurar la lógica secuencial requerida para hacer funcionar tanto máquinas como procesos. Inicialmente, su coste los limitó a grandes sistemas. Pero el gran desarrollo que experimentaron más tarde reestructuró radicalmente su función de tratamiento:

- los lenguajes han sido unificados y normativizados; la norma IEC 61131-3 define sus diferentes tipos;
- el enfoque del sistema se ha generalizado y el principio de esquema ha dado paso a los bloques de función;
- los sistemas digitales se utilizan actualmente de forma amplia para el tratamiento de datos tanto digitales como analógicos previa conversión A/D;
- el coste de los componentes electrónicos ha caído tanto que pueden usarse autómatas en vez de relés incluso en sistemas simples;
- los buses de comunicación utilizados para el intercambio de datos son una alternativa competitiva al cableado convencional;
- la tecnología informática se utiliza cada vez más en sistemas de automatización industrial;
- los interfaces hombre-máquina también han progresado en ser programables, de ahí su gran flexibilidad.

10. Tratamiento de datos y software

10.3 Programación, configuración y lenguajes

Un programa de automatización consta de un conjunto de instrucciones que deben ejecutarse en un orden específico por parte de un procesador. A menudo, la palabra "programa" se utiliza como sinónimo de "software". Si bien un software consta básicamente de programas, a menudo requiere ficheros de recursos que contienen datos de todas las clases que no forman parte del programa.

Aquí es donde entra en juego el término "configurar", que no es "programar". La configuración completa el software aportando datos que necesita para funcionar.

Como ejemplo, tomemos un sistema de tratamiento de aguas: puede ser desde lo más sencillo (un programa sencillo para mantener el nivel de agua de un tanque entre dos límites, abriendo y cerrando una electroválvula) hasta lo más complejo (control del flujo de entrada y del flujo de salida del tanque para mantener el flujo de agua a un determinado valor). Una aplicación industrial, como es el caso del tratamiento de aguas residuales, controlará varios tanques a la vez. Cada tanque deberá cumplir un conjunto de condiciones, como por ejemplo:

- nivel de agua entre un límite mínimo y un límite máximo;
- factor pH dentro de un rango establecido;
- tener un determinado flujo de salida, etc.

■ Lenguajes normalizados

La Comisión Electrotécnica Internacional (CEI, en inglés IEC) ha desarrollado la norma IEC 61131 para autómatas programables. La tercera parte de esta norma (IEC 61131-3) define los lenguajes de programación:

- **IL** ("Instruction List") es muy similar al lenguaje ensamblador, se trabaja lo más cerca posible del procesador utilizando la unidad aritmética y lógica, sus registros y acumuladores;
- **ST** ("Structured Text") es similar al lenguaje C utilizado en los ordenadores;
- **LD** ("Ladder Diagram") o escalera se asemeja a los esquemas eléctricos y permite transformar rápidamente un antiguo programa basado en relés electromecánicos. Esta forma de programar permite un enfoque visual del problema;
- **FBD** ("Function Block Diagram") se presenta en forma de diagrama: son una serie de bloques, conectados entre ellos, que realizan cualquier tipo de función desde la más simple hasta la más complicada;
- **GRAFCET** (GRÁFica de Control de Etapas de Transición) es un método de representación y análisis de un sistema de automatización, en particular muy bien adaptado a los sistemas secuenciales dado que se puede descomponer en dos etapas. En la programación de autómatas programables, se puede usar **SFC** de forma similar (GRAFCET, con la IEC848, se convirtió en una norma internacional en 1988 bajo el nombre de SFC, "Sequential Function Chart"). Detrás de cada acción hay un programa asociado escrito en cualquiera de los lenguajes descritos: IL, ST, LD o FBD.

10. Tratamiento de datos y software

10.4 Categorías de aplicación

El progreso tecnológico, impulsado por las necesidades de los usuarios, ha dado lugar a una amplia gama de autómatas caracterizada por:

- un hardware como la potencia de tratamiento, el número y características de entradas/salidas, velocidad de ejecución, módulos especiales (control de ejes, comunicación, etc.);
- un software que, a parte de un lenguaje de programación, tiene unas mejores funciones y capacidades de comunicación e interacción con otros softwares de la empresa.

Éstos serán descritos a través de aplicaciones típicas para ayudar a dirigir la elección del lector. Nuestro consejo es consultar la documentación específica de cada producto.

En la introducción de esta guía, analizamos el principio de la arquitectura de un sistema de automatización y las mejores implementaciones, según el perfil del cliente. Las soluciones descritas pueden dividirse en cuatro categorías:

A - Soluciones "electricista"

Las aplicaciones son simples, autónomas y fijas. Los criterios de elección se basarán en la facilidad de los productos a usar, en su bajo coste y en la simplicidad de su mantenimiento.

B - Soluciones "ingeniero de automática / mecánico"

Las aplicaciones son exigentes en lo que a prestaciones mecánicas se refiere (precisión, rapidez, control de movimiento, etc.). Sus arquitecturas y sistemas de tratamiento se escogen en función de las prestaciones.

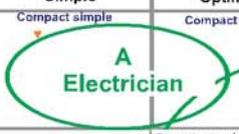
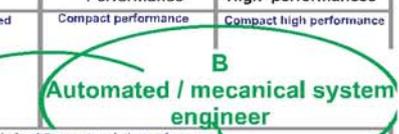
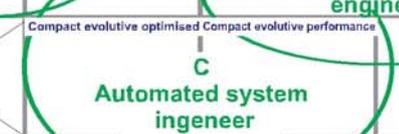
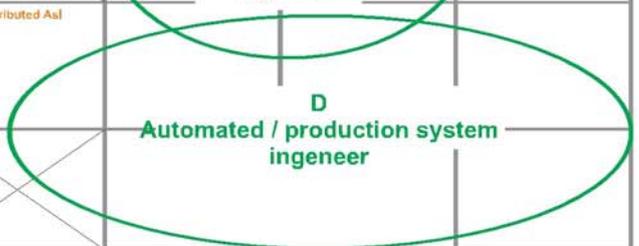
C - Soluciones "ingeniero de automática"

Los automatismos son complejos debido al volumen y la variedad de los datos a tratar como ajustes, interconexiones entre autómatas programables, número de entradas/salidas, etc.

D - Soluciones "ingeniero de automática / producción"

Los sistemas automatizados de producción deben integrarse en la arquitectura informática de la planta. Debe existir interacción entre ellos e intercambiar datos con el software de producción y gestión, etc.

En la *figura 1*, estas categorías se posicionan de acuerdo con las implementaciones descritas en el primer capítulo de esta guía.

Architecture	Simple	Optimised	Performance	High performances
	Compact simple	Compact optimized	Compact performance	Compact high performance
predéfinied	 A Electrician	 B Automated / mechanical system engineer		
Compact				
évolutive	 C Automated system ingeener		 D Automated / production system ingeener	
Distribuee	Distributed Asl			

↑ Fig. 1 Categorías de aplicación y perfiles de usuario

10. Tratamiento de datos y software

■ Soluciones "electricista"

Las soluciones simples utilizan una serie de relés electromecánicos para ejecutar secuencias de automatización. La última generación de pequeños autómatas son fáciles de programar y usar, y resultan competitivos en precio a partir de un determinado número de entradas/salidas y ofrecen nuevas capacidades.

Las aplicaciones típicas las encontramos en los siguientes sectores

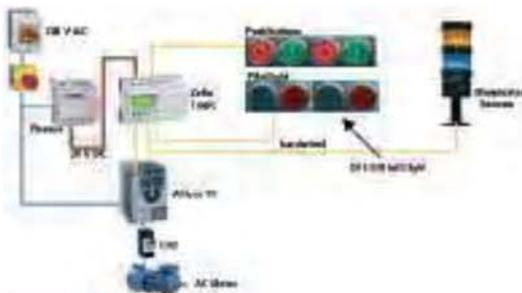
- industria: máquinas simples y funciones adicionales en sistemas de tipo descentralizado;
- edificios y servicios: gestión del alumbrado, control de accesos, vigilancia de las instalaciones, calefacción, ventilación, aire acondicionado.

□ Aplicación con un autómata Zelio

Esta configuración se ajusta a las aplicaciones mostradas en la *figura 2*:

Aplicación	Descripción	Ejemplo
Estación móvil de bombeo	Una aplicación para llenar y vaciar tanques. El uso de un variador de velocidad ayuda a ajustar los grados de viscosidad en fluidos.	
Puerta automática	Abrir y cerrar las puertas de una fábrica.	
Ventana eléctrica	Controlar el aire en un centro de jardinería.	

↑ Fig. 2 Ejemplos con un autómata Zelio



↑ Fig. 3 Aplicación basada en un autómata Zelio

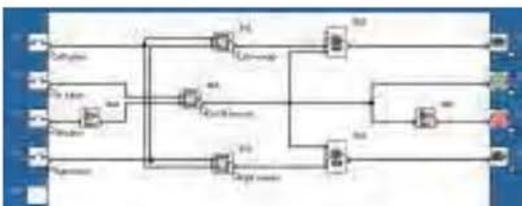
En el diagrama (⇒ Fig. 3), el funcionamiento del motor está gobernado por un variador de velocidad. Para un control "todo o nada", todo lo que requiere es un contactor asociado a un relé térmico.

Esta unidad consta de:

- un autómata Zelio Logic;
- una fuente de alimentación de CC de 24V Phaseo;
- un variador de velocidad Altivar 11;
- un disyuntor motor GV2;
- una columna de señalización XVB;
- un interruptor seccionador Vario VCF.

Las variables del variador de velocidad (tiempo, velocidad, control) se pueden establecer directamente en el Altivar 11 o con el software Powersuite.

El Zelio se puede programar directamente en el módulo o con el software Zeliosoft instalado en un PC. Esta última opción se ilustra en la captura de pantalla que se ve en la *figura 4*, que muestra un proceso lógico realizada por FDB ("Function Block Diagram").



↑ Fig. 4 Captura de pantalla de Zeliosoft

10. Tratamiento de datos y software

□ Aplicación con un autómata Twido

Este tipo de autómata se utiliza para implementar aplicaciones simples que se pueden supervisar o controlar de forma remota a través de un módem conectado a la red telefónica (RTC). La *figura 5* presenta algunos ejemplos:

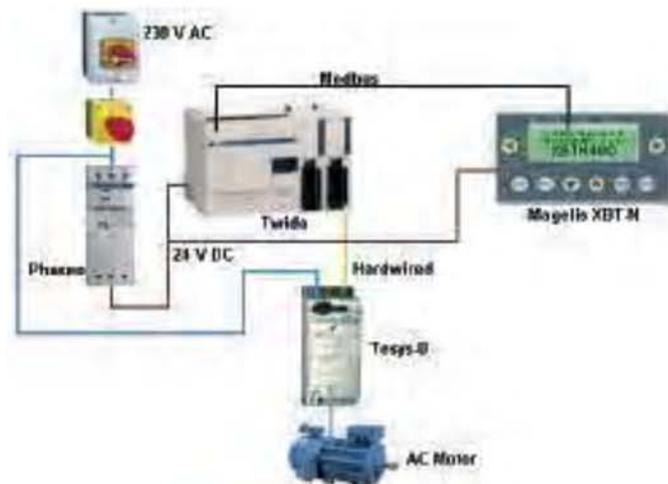
Aplicación	Descripción	Ejemplo
Ventilación	Control de un sistema de ventilación en un edificio industrial. La medida de temperatura gobierna el arranque y parada del sistema.	
Calefacción	Control del sistema de calefacción del edificio.	
Control remoto de una fuente	Control de la infraestructura de una fuente. El sistema se controla de forma remota con un módem.	
Control de la limpieza de filtros en una planta de distribución de agua	La aplicación controla y limpia el filtro en una planta de distribución de aguas mediante una secuencia de limpieza por aire y agua. El sistema puede también controlarse via módem.	

↑ *Fig. 5* Ejemplos con un autómata Twido

□ Esquema típico

El sistema se desarrolla a partir de un autómata Twido (⇒ *Fig. 6*) y se controla y supervisa a través de un terminal Magelis. La seguridad queda garantizada mediante un paro de emergencia sobre el interruptor principal.

El cableado se realiza hilo a hilo y el autómata controla el arranque y gestiona los mensajes del módulo de alarma.



↑ *Fig. 6* Aplicación basada en un autómata Twido

10. Tratamiento de datos y software

Este sistema incluye los siguientes componentes:

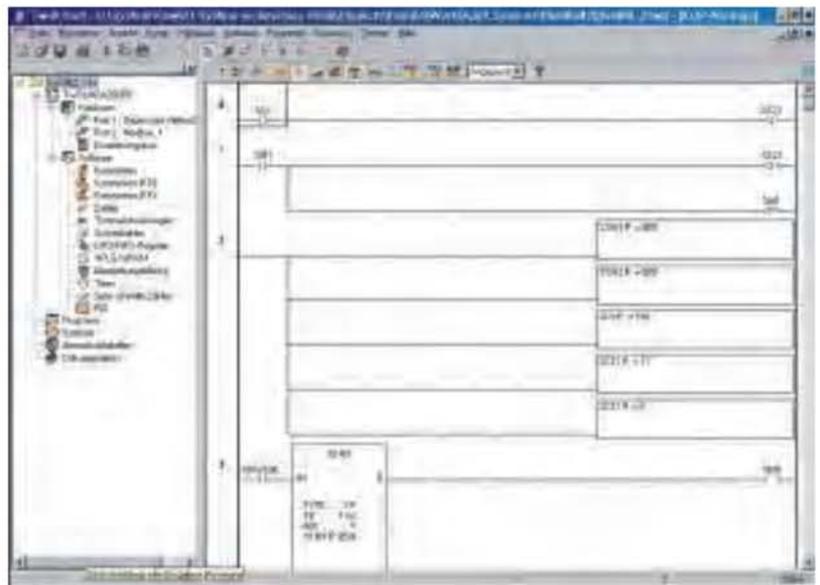
Hardware:

- Twido Modular (autómata);
- Phaseo (fuente de alimentación);
- TeSys-U (arrancador controlador);
- Magelis XBT-N (interface hombre-máquina);
- motor trifásico estándar.

Software:

- Twidosoft V2.0;
- Magelis XBTL1003M V4.2.

La captura de pantalla de la *figura 7* del software Twidosoft ilustra la programación en escalera que es reversible en List. El software incluye un gran número de instrucciones y se usa un navegador integrado para acceder a todos los objetos de forma directa.



↑ Fig. 7 Captura de pantalla de Twidosoft

■ Soluciones "ingeniero de automática / mecánico"

Algunas aplicaciones requieren prestaciones difíciles de conseguir sin asociar de forma estrecha las funciones de tratamiento y adquisición de datos, y control de la potencia. Por esta razón, esta asociación es directamente gestionada por la función de control de la potencia, ya sea analógica, por buses rápidos o buses especiales. Este tipo de arquitectura se encuentra en los variadores de velocidad de motores tanto asíncronos como síncronos (bucle cerrado).

La necesidad de asociar un tratamiento secuencial de variables analógicas y digitales significó que los fabricantes añadieron a los variadores:

- tarjetas de entradas/salidas;
- tarjetas de comunicación;
- tarjetas de tratamiento de tipo autómata programable.

Estas soluciones se pueden usar de forma equitativa en aplicaciones industriales y en infraestructura.

Ilustraremos esta solución con un variador de velocidad con un autómata integrado.

10. Tratamiento de datos y software



↑ Fig. 8 Altivar 71 con tarjeta Controller Inside

□ Tratamiento integrado en un variador de velocidad

La opción de una tarjeta programable (⇒ Fig. 8) se puede emplear para convertir un variador de velocidad Altivar en una célula de automatización. Esta tarjeta, llamada "Controller Inside" tiene todas las funciones de un autómata integradas en él:

- el software de programación CoDeSys con capacidad para lenguajes gráficos normalizados según IEC 61131-3;
- el tratamiento cerca del control del motor para rapidez de movimiento;
- un bus CANopen maestro para gobernar otros variadores de velocidad (Altivar 31, Altivar 61 y Altivar 71) e intercambiar todos los datos necesarios;
- la tarjeta tiene sus propias entradas/salidas y accede a las del Altivar;
- las funciones visualización terminal gráfico se guardan para informar/adaptar;
- la supervisión disponible vía Ethernet, Modbus o cualquier otra red de comunicación.

CoDeSys es una herramienta de programación en Windows. Soporta los cinco lenguajes estándar (IEC 61131-3). CoDeSys produce un código nativo para los procesadores más corrientes y se puede usar en diferentes plataformas de controladores. Combina la potencia de los lenguajes de programación avanzados como C y Pascal, con las funciones de los sistemas de programación de los autómatas.

Diversos fabricantes, entre los que destaca Schneider Electric, lo ha elegido para la gama Altivar (variadores de velocidad) y Lexium (control de movimiento).

□ Aplicación con un ATV71 + Controller Inside

Esta configuración se adapta a las siguientes aplicaciones (⇒ Fig 9).

Aplicación	Descripción	Ejemplo
Red de infraestructuras	Uso en estaciones de bombeo de agua potable.	
Máquinas especiales	Diversas aplicaciones: - enrollador/desenrollador - máquinas automáticas de ensamblaje - máquinas trabajo madera.	
Manutención	Utilizado en combinación con equipamiento de elevación y lanzaderas.	

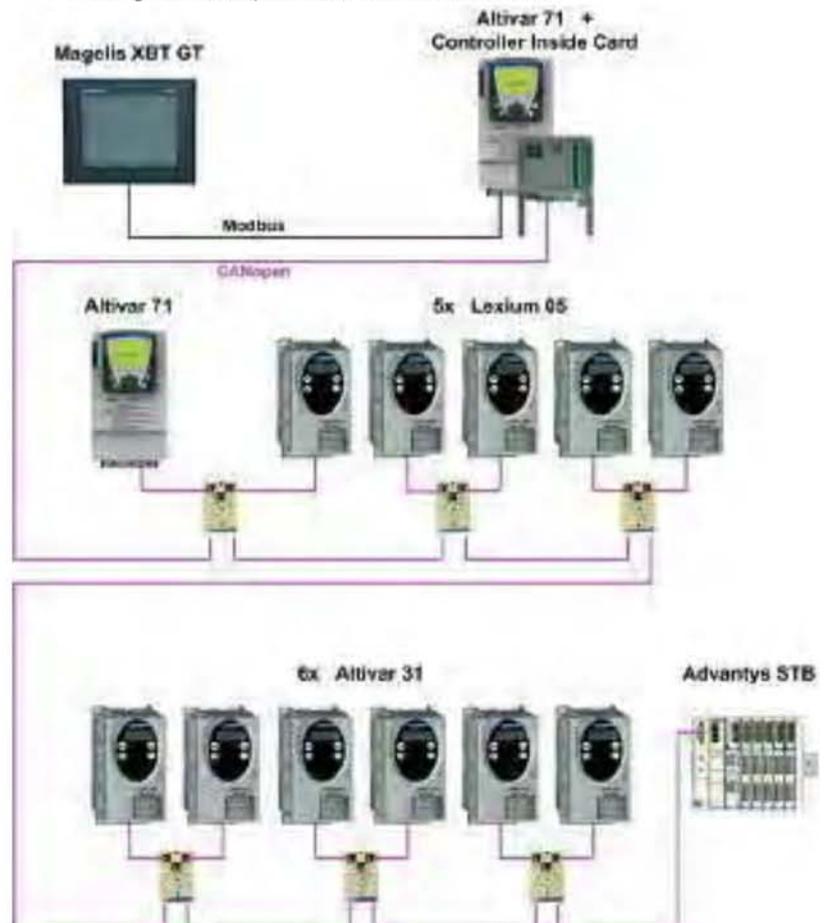
↑ Fig. 9 Ejemplos con un variador de velocidad ATV71 con tarjeta Controller Inside

10. Tratamiento de datos y software

□ Esquema típico

Para ilustrar la solución descrita de una forma más clara, la parte de potencia y su alimentación no se mostrarán en la *figura 10*.

En el siguiente esquema, el sistema consta de:



↑ Fig. 10 Aplicación basada en ATV71 + Controller Inside

Hardware:

- tarjeta Controller Inside instalada en un Altivar 71; el variador de velocidad es el maestro en un bus CANopen;
- variadores de velocidad ATV31 y ATV71 con interface CANopen integrado;
- Servo Lexium05 con interface CANopen integrado. El interface hombre-máquina está gestionado por un terminal gráfico Magelis XBT-GT y conectado con la célula de producción con un enlace Modbus;
- entradas/salidas distribuidas Advantys STB.

Software:

- PS1131 (CoDeSys V2.3);
- PowerSuite para ATV31, ATV71 y Lexium05;
- Vijeo-Designer V4.30 para Magelis;
- Advantys Configuration Tool V2.0.

■ Soluciones "ingeniero de automática"

Los sistemas de automatización actuales requieren un gran número de entradas y salidas de diferentes tipos. Deben ser capaces de procesar secuencias de automatización y de proporcionar la información necesaria para la gestión y el mantenimiento; su complejidad conlleva un diseño y una implementación largos y costosos. Los fabricantes ofrecen dos soluciones para reducir costes:

- gamas modulares de hardware y software para reducir el coste final de enseñanza de todas las aplicaciones;
- herramientas de software para impulsar la productividad, gestionar los registros y facilitar el mantenimiento y las posibles mejoras posteriores.

10. Tratamiento de datos y software

Este tipo de solución de automatización se utiliza en procesos industriales donde varias máquinas están conectadas entre sí o en infraestructuras. Se muestran algunos ejemplos en la tabla de la *figura 11*.

Aplicación	Descripción	Ejemplo
Manutención	Usado en un proceso con varios sistemas de cinta transportadora y que usa info externa.	
Máquinas empaquetado, máquinas téxtiles, máquinas especiales	Utilizado para máquinas integradas en una línea de producción.	
Bombas y ventiladores	Usado para los sistemas de circulación de agua y refrigeración gobernado por mediciones externas como flujos de salida.	

↑ *Fig. 11* Ejemplos de aplicación en procesos industriales

□ Aplicación con un autómatas Premium

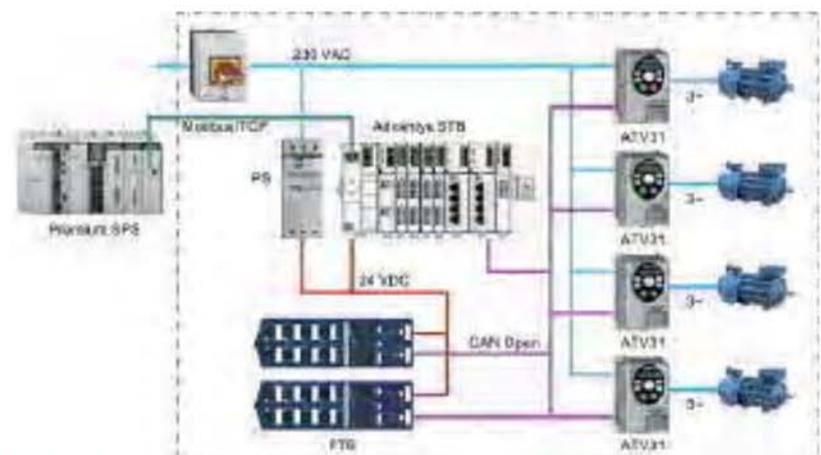
Nuestra intención no es describir una aplicación de forma detallada, sino ilustrar su principio de funcionamiento, por lo que daremos una descripción de una parte significativa.

Un autómatas Premium (⇒ *Fig.12*) se usa para controlar una célula de producción local. Ésta está formada por entradas/salidas Advantys STB, cuatro variadores de velocidad y módulos de entradas/salidas externos. Cada elemento está conectado a un bus CAN. Esta implementación se puede expandir fácilmente añadiendo más variadores de velocidad y entradas/salidas. El autómatas está conectado a la célula de producción a través de un bus Modbus/TCP. Los variadores y motores están alimentados por una red de 230V CA. Se usa otra fuente para obtener alimentación a 24V CC.

□ Esquema típico (⇒ *Fig.12*)

Listado de componentes del sistema:

- TSX Premium (autómatas),
- Phaseo (fuente de alimentación),
- ATV31 (variador de velocidad),
- Advantys STB (célula de entradas/salidas),
- Advantys FTB IP67 (módulo de entradas/salidas),



↑ *Fig. 12* Aplicación basada en un autómatas Premium

10. Tratamiento de datos y software

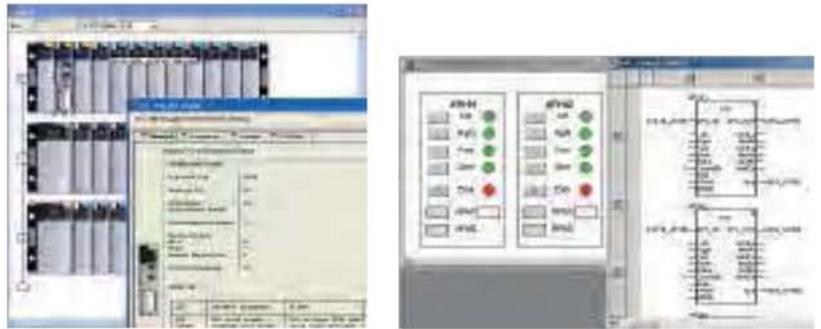
- motor trifásico de jaula de ardilla.

Software:

- Unity Pro V2.0.2 (autómata),
- Advantys Configuration Software V1.20 (célula de entradas/salidas),
- PowerSuite V2.0 (configuración del variador de velocidad ATV31).

□ Unity Pro

Unity Pro es el software común de programación para la puesta a punto y la explotación de los autómatas Modicon Premium, Atrium y Quantum. Basado en los estándares de PL7 y Concept, Unity Pro abre el camino a un conjunto de nuevas funciones (⇒ Fig. 13) para una mayor productividad y colaboración entre softwares.



↑ Fig. 13 Captura de pantalla de Unity Pro

• Características principales de Unity Pro

- interfaces gráficos Windows 2000/XP;
- iconos y barras de herramientas "personalizados";
- perfiles de usuario;
- diseño gráfico de configuraciones de hardware;
- convertidores de PL7 y Concept integrados;
- generación automática de variables de sincronización en Ethernet (sistema Global Data);
- los 5 lenguajes son soportados (IEC61131-3) con editores gráficos;
- integración y sincronización de editores, programas, datos, bloques de funciones de usuarios;
- reutilización "drag & drop" de objetos de la biblioteca;
- importación/exportación XML y reasignación automática de datos;
- automatización de tareas repetitivas por macros VBA;
- sistema de redundancia Hot Standby "plug & play".

Unity Pro ofrece un conjunto de funciones y herramientas que permiten copiar la estructura de la aplicación sobre la estructura del proceso o de la máquina. El programa se divide en módulos funcionales que, ensamblados y jerarquizados, forman la vista funcional y contienen:

- secciones de programa;
- tablas de animación;
- pantallas de operador;
- hipervínculos.

Las funciones básicas, utilizadas de forma repetitiva, pueden ser encapsuladas en bloques funciones usuario (DFB) en lenguaje IEC61131-3.

Para crear una base referencias aplicativos, Unity Pro soporta bibliotecas de proyectos y aplicaciones localmente o en el servidor.

Tiene alrededor de 800 funciones estándar, y se puede enriquecer con los estándares del cliente (variables, tipos de datos, bloques de función).

10. Tratamiento de datos y software

Además incluye:

- variables simbólicas independientes de la memoria física;
- tipos de datos estructurados definidos por el usuario (DDT);
- gestión de versión de los bloques de funciones DDT y DFB en la librería;
- una biblioteca de objetos gráficos preanimados para pantallas de operador;
- protección contra lectura/escritura de los datos de programación para prevenir modificaciones;
- posibilidad de desarrollo de bloques de funciones en C++ con la opción Unity EFB Toolkit.

Colocados en librerías en un PC local o en un servidor remoto, los objetos de la aplicación y sus propiedades son utilizados y compartidos por todos los programas y cualquier cambio hecho a un objeto de una librería es efectivo en los programas que lo usan:

- los módulos funcionales pueden reutilizarse en la aplicación o entre proyectos por importación/exportación XML;
- los bloques de funciones se insertan mediante "drag & drop" desde la librería;
- todas las instancias heredan automáticamente los cambios en la librería (como escoja el usuario);
- selección y parametrización de los objetos gráficos para pantallas de operador por "drag & drop".

Dentro de Unity Pro, se integra un simulador de autómatas programables, que se utiliza para probar la aplicación al máximo antes de la puesta en servicio in situ. Reproduce de forma exacta el comportamiento del sistema. Todas las herramientas de puesta a punto se pueden utilizar en la simulación:

- ejecución paso a paso del programa;
- punto de paro y visualización;
- animaciones dinámicas para visualizar el estado de las variables y de la lógica en la ejecución.

Las pantallas de explotación facilitan la puesta a punto mediante vistas que representan el estado de las variables en forma de objetos gráficos.

Igual que para la configuración, los módulos específicos de la aplicación tienen pantallas especiales para su puesta a punto: las funciones disponibles están adaptadas al tipo de módulo implementado (entradas/salidas digitales y analógicas, contaje, comunicación, etc.).

Las acciones del operador quedan registradas y archivadas en un archivo de seguridad estándar de Windows. Los enlaces de hipertexto se usan para enlazar la aplicación a todos los documentos y herramientas de operación y mantenimiento.

• Herramientas de diagnóstico

Unity Pro incluye una biblioteca de DFBs de diagnóstico de aplicaciones. Ésta está integrada en el programa y, dependiendo de su función, se utilizan para controlar las condiciones de seguridad permanentes y el desarrollo del proceso.

Una ventana de visualización muestra todo defecto del sistema y de la aplicación de forma explícita y cronológica en tiempo real desde la fuente. Un click en la ventana abre el editor del programa donde sucedió el error (búsqueda de condiciones que faltan en la fuente).

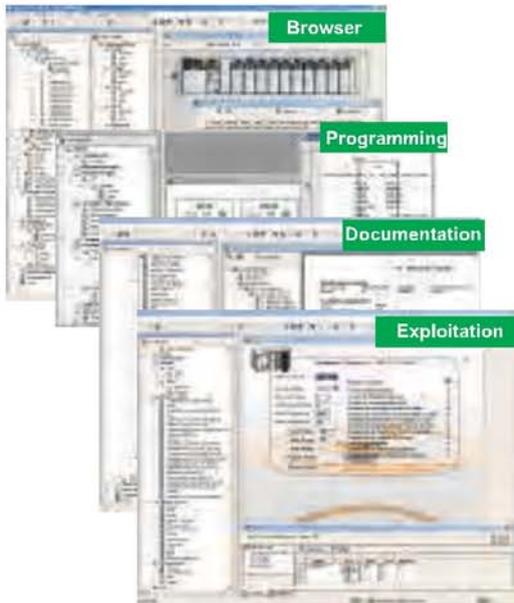
Los cambios online se pueden agrupar coherentemente en modo local en un PC y se pueden transferir directamente al autómata en una operación a incluir en el mismo ciclo de ejecución.

Los enlaces de hipertexto integrados en la aplicación dan acceso remoto o local a recursos útiles (documentación, herramientas adicionales, etc.) para minimizar el tiempo de paro.

Existe una amplia gama de funciones para controlar la explotación:

- registro de las acciones del operador sobre Unity Pro en un archivo securizado;
- perfil del usuario con elección de funciones accesibles y protección con contraseña.

10. Tratamiento de datos y software



↑ Fig. 14 Pantallas de explotación, vistas funcionales

Las pantallas de explotación y las vistas funcionales permiten un acceso gráfico directo a los elementos de la aplicación (⇒ Fig. 14).

La arquitectura cliente/servidor de Unity da acceso a los recursos de software a través de los interfaces de programación en VBA, VB o C++; por ejemplo, es el caso de:

- automatización de tareas repetitivas (configuración, traducción, etc.);
- integración de aplicaciones específicas (generador de código, etc.).

• Intercambio con otros softwares

El formato XML, el estándar universal W3C para el intercambio de datos a través de internet, se usa como formato fuente de las aplicaciones Unity como por ejemplo variables, programas, entradas/salidas, configuraciones, etc. (⇒ Fig. 15).

La simple importación/exportación se usa para intercambiar todo o parte de la aplicación con otro software del proyecto (CAD, etc.).

El "Unity Developer's Edition" (UDE) y sus interfaces de programación en C++, Visual Basic y VBA se pueden usar para desarrollar soluciones personalizadas como los interfaces con CAD eléctrico, un generador de variables, un programa de un autómatas o la automatización de tareas de diseño repetitivas. Muchos editores de software usan UDE para simplificar el intercambio con Unity Pro (⇒ Fig. 16).



↑ Fig. 15 Utilización del formato XML en Unity Pro

Ámbito	Empresa	Producto
Electrical CAD	ECT	Promise
Electrical CAD	EPLAN	EPLAN
Electrical CAD	IGE-XAO	SEE Electrical Expert
Electrical CAD	AutoDesk	AutoCAD Electrical
Electrical CAD	SDProget	SPAC Automazione
Process Simulation	Mynah	Mimic
Change Management	MDT Software	AutoSave
Application Generator	TNI	Control Build
SCADA/Reporting	Iconics	GENESIS BizViz Suite
SCADA	EuropSupervision	Panorama
SCADA	Arc Informatique	PCVue32
Graphical User Interface	ErgoTech	ErgoVU
SCADA	Areal	Topkapi
SCADA	Afcon	P-CIM
MES	Tecnomatix/UGS	XFactory
Historian/RtPM	OSISoft	PI
Web Services	Anyware	PLC Animator

↑ Fig. 16 Editores de software que utilizan el interface UDE

10. Tratamiento de datos y software

• Compatibilidad con aplicaciones existentes

Las aplicaciones PL7 y Concept IEC61131 son importadas a Unity Pro por un convertidor integrado como característica estándar. La actualización del sistema operativo, proporcionada con Unity Pro, está disponible para la mayoría de los procesadores de los autómatas Modicon Premium y Quantum.

Los periféricos existentes de E/S, específicos de la aplicación, comunicación y módulos de buses de campo permanecen compatibles con Unity Pro.

■ Soluciones "ingeniero de automática / producción"

Aquí discutiremos arquitecturas complejas que cuentan con varios autómatas que se comunican entre ellos y con estaciones de trabajo de supervisión (SCADA). Estas arquitecturas se intercomunican también con los sistemas de gestión de la empresa.

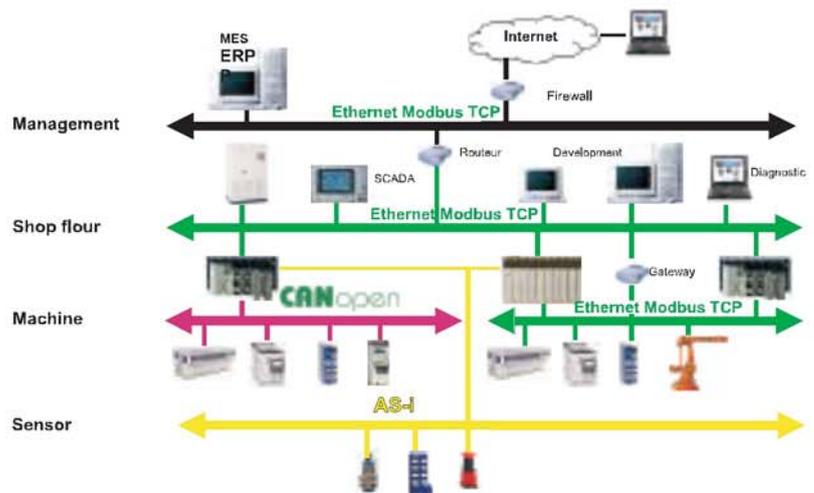
La optimización del sistema requiere una visión global para interrelacionar todos los departamentos del negocio, socios y medios. Podemos distinguir dos tipos de interacción:

- tiempo real: en la fase de explotación, esto caracteriza el enlace con los sistemas de gestión de la relación con el cliente (CRM), gestión del stock y de la producción (MES) y gestión contable para conseguir la optimización de los flujos;
- colaborativo: las herramientas de software soportan la relación entre los actores relevantes en las etapas de diseño, construcción, funcionamiento y mantenimiento para recortar el tiempo y los costes correspondientes a las herramientas de producción, así como para mejorar la calidad.

Las aplicaciones están tan diversificadas que no es fácil comprender la localización de cada una de ellas en relación a su entorno. La necesidad de intercambio con otros softwares conduce a un enfoque colaborativo en toda la empresa.

□ Tiempo real y software corporativo

La ilustración de la *figura 17* muestra el software que se puede encontrar de forma más común en un taller o factoría. Se puede separar en cuatro niveles:



↑ Fig. 17 Los cuatro niveles de la empresa

10. Tratamiento de datos y software

- el **nivel de gestión** (nivel corporativo) se caracteriza por un enorme flujo de información. Las normas de la ofimática y de Internet son ahora básicas. El software se ejecuta en un PC o en servidores más potentes;
- el **nivel de taller** está caracterizado principalmente por las herramientas de supervisión y los autómatas que gobiernan el proceso. Ethernet es ahora el estándar de comunicación entre el ordenador y los autómatas;
- en el **nivel de máquina**, el concepto de tiempo real condiciona la elección de las herramientas de comunicación. Las tareas se distribuyen entre PCs industriales y autómatas, las conexiones se realizan por buses de campo (CANopen para Schneider Electric) o por Ethernet asociado a capas de aplicación específicas;
- en el **nivel de sensor** las conexiones hilo a hilo compiten con el bus AS-Interface, que es particularmente adecuado a este tipo de uso.

□ Diseño y construcción colaborativos

El software del autómata descrito antes se limita a controlar conjuntos concebidos de forma separada para funcionar como unidades autónomas, aunque se requieran en algunos casos enlaces de comunicación.

Se han desarrollado nuevas herramientas alrededor de los software de programación para tratar con las dificultades que supone trabajar en paralelo en el diseño y en la construcción. Permiten ayudar a los desarrolladores a simplificar y agilizar su trabajo, a garantizar el rastreo y a reducir los errores.

En efecto, consideran el ciclo desde la fase de estudio hasta el final de la vida de una máquina o proceso (⇒ Fig. 18). La gran diversidad de tareas que implica este ciclo de vida requiere la colaboración interdepartamental. Esta colaboración se hace posible gracias a herramientas como CAD mecánico y eléctrico, ERP, MES, SCADA y otros.

En el centro de la oferta de Schneider Electric, Unity Pro y sg² (sistema de control híbrido) proporcionan una plataforma de automatización (soft y hard) basada en la apertura y la colaboración entre las herramientas que cubren las diferentes necesidades desde la fase de estudio hasta el mantenimiento del sistema (⇒ Fig. 18).



↑ Fig. 18

Las dos plataformas Unity Pro y sg²

10. Tratamiento de datos y software

Se han desarrollado varias herramientas de software adicionales para asegurar la colaboración. La tabla de la *figura 19*, que describe sus funciones, estará seguida de una breve explicación de cómo funcionan.

Los principales estándares en la informática como Ethernet TCP/IP, Web, OPC, SOAP, XML, etc. se utilizan para facilitar la colaboración vertical en cada nivel corporativo y ofrecen:

- más información visible en tiempo real;
- interoperatividad entre el proceso y los sistemas de información (como es el caso de MES, ERP, etc.);
- intercambio con herramientas de diseño como el CAD.

Componentes	Función
Unity Pro	Taller de desarrollo de aplicaciones monoautomata en entorno colaborativo
sg ² (sistema de control híbrido)	Desarrollo de aplicaciones multiautomata y SCADA orientado a objetos Cumple con la norma ISA S88
UDE "Unity Developer Edition"	Software de desarrollo en los lenguajes informáticos VBA, VB y C++
OFS "OPC Factory System"	Servidor OPC de Schneider para interrelacionar los entornos automata e informático
Factory Cast entorno web	Asegura que la información pasa entre el entorno del automata y el entorno informático
CITECT SCADA	Software SCADA
AMPLA	Software MES
Tarjeta Ethernet	Tarjetas que usan servicios Factory Cast
Software para la configuración y la parametrización	
XBT L1000	Creación de diálogo operador
Vijeo Designer	Creación de diálogo operador
Vijeo Look	Mini SCADA
Power Suite	Configuración de los variadores Altivar, de los arrancadores Altistar y del arranque motor Tesys U

↑ *Fig. 19* Herramientas de software complementarias

UDE ("Unity Developer's Edition")

La gama Unity se enriquece con "Unity Developer's Edition" (UDE), un software especializado para programadores en VBA, VB or C++. Proporciona acceso a todos los servidores de objetos en el software Unity Pro para desarrollar soluciones personalizadas.

sg² (sistema de control híbrido)

sg² es una herramienta de diseño basada en un enfoque por módulos reutilizables (PID, válvula, motor, etc.) y cumple con la norma ISA 88. sg² genera el código del programa de los autómatas en la arquitectura así como los gráficos del SCADA. Además, con una entrada única gestiona una base de datos común a todos los elementos, para asegurar la consistencia de la aplicación.

La entrada de datos única asegura la velocidad y la consistencia entre los dos entornos.

Con este enfoque de diseño estructurado y modular, sg² ofrece ahorros significativos en costes de desarrollo, y facilita la validación y el mantenimiento de los proyectos de automatización.

10. Tratamiento de datos y software

OFS ("OPC Factory System")

OFS, OPC ("OLE for Process Control") adaptado al entorno de Schneider Electric, es un programa para la comunicación entre el entorno informático y los automatismos industriales. Se originó con Microsoft, y deriva de DDE; después se pasó de OLE a OLE Automation a través de la utilización de Windows COM/DCOM ("Component Object Model/Distributed Component Object Model").

Una fundación, integrada por distribuidores y editores de software, gestiona las mejoras de OPC y garantiza la compatibilidad ascendente y la interoperatividad entre los diferentes softwares.

Las mejoras están condicionadas por los siguientes requerimientos industriales:

- interfaces de aplicación basados en estándares abiertos que ofrecen un acceso simple y común a los datos de nivel taller;
- mayor interoperatividad entre las aplicaciones de automatización y control, el equipamiento de campo y las aplicaciones informáticas;
- arquitectura de múltiples clientes/servidores;
- acceso a un servidor local o remoto;
- flujo de información en tiempo real.

Factory Cast: entorno web

Es un conjunto de herramientas que permiten a las aplicaciones comunicarse por Internet, y que cumplen los siguientes requerimientos:

- comunicación entre aplicaciones;
- compatibilidad web e Internet;
- utilización de soluciones no propietarias basadas en normas y modelos;
- implementación sencilla.

Las aplicaciones deben ser capaces de comunicarse independientemente de:

- el lenguaje en el que fueron desarrolladas;
- el sistema operativo sobre el que se ejecutan.

La tecnología compatible con Internet está basada en el protocolo XML SOAP ("Simple Object Access Protocol"), que cohabita con HTTP, y que permite a las aplicaciones comunicarse entre ellas.

Una descripción estándar de los servicios e interfaces es proporcionada por una aplicación WSDL ("Web Service Description Language"), que es una biblioteca de funciones de acceso a los datos estandarizados y autodocumentados.

La combinación de estas tecnologías se conoce como "servicios web" y es independiente de las plataformas, lenguajes y sistemas operativos.

SCADA: software de supervisión

El SCADA ("Supervisory Control And Data Acquisition") es un software industrial diseñado para optimizar la gestión de la producción. Se utiliza para controlar un taller de producción en tiempo real, a partir de las necesidades de producción y los medios disponibles.

Tarjetas de Ethernet

La gama de tarjetas de Ethernet ofrece modernas arquitecturas abiertas a las diferentes tecnologías informáticas actuales y proporcionan una serie de herramientas a los usuarios que les permiten realizar sus propias funciones. Esto permite organizar objetos de forma compatible con MES y ERP IT.

10