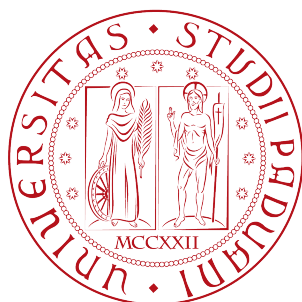


UNIVERSITÀ DEGLI STUDI DI PADOVA

FACOLTÀ DI INGEGNERIA DELL'INFORMAZIONE
Corso di Laurea triennale in Ingegneria dell'Informazione

Interconexión entre aparatos del
estándar *EIB/KNX* y un *API S7-200*
por medio de *Logo!*



Relatore:
Prof. Lorenzo Vangelista

Candidato:
Franco Gobbo

25 Novembre
Anno Accademico 2010/2011

to he who trusted on me
though I was not trusting
either on myself

Resumen

En el mercado no está actualmente disponible un modulo de comunicación que permita a un *API S7-200* de comunicar con aparatos de estándar *KNX*.

Se ha intentado con este proyecto encontrar una solución al problema en objeto, interponiendo entre los dos sistemas un tercer elemento que tenga la posibilidad de comunicar separadamente con ellos y que funcione de nudo de intercambio. El dispositivo elegido como punto de interconexión es un *Logo!* de *Siemens* ya que para eso son disponibles unos módulos de comunicación con el protocolo *KNX* y también con el protocolo *AS-i*.

Se quiere realizar un panel de laboratorio que reúna los tres elementos y permita comprobar la efectiva posibilidad de actuar sobre dos cualquiera de ellos a partir del tercero.

Sommario

Non è attualmente disponibile nel mercato un modulo di comunicazione che permetta ad un *PLC* di comunicare con dispositivi dello standard *KNX*.

Con questo progetto si è cercato di trovare una soluzione al problema in oggetto, interponendo tra i due sistemi citati un terzo elemento in grado di interfacciarsi con entrambi e di tradurne i protocolli. Il dispositivo scelto come nodo di scambio è il modulo logico *Logo!* prodotto da *Siemens* dato che per esso sono disponibili dei moduli di comunicazione sia per lo standard *KNX* che per *AS-i*.

Si desidera infine realizzare un pannello per esercitazioni che disponga dei tre elementi liberamente configurabili e permetta di verificare l'effettiva possibilità di interazione di ciascuno di essi con qualunque dei rimanenti.

Índice general

1. Introducción	3
2. Objetivos	5
3. Dispositivos en examen	6
3.1. KNX	6
3.1.1. Introducción	6
3.1.2. Características	6
3.2. Logo!	7
3.2.1. Introducción	7
3.2.2. Características	8
3.2.3. Elementos de Operación e Indicación	8
3.2.4. Módulos de expansión y de comunicación	9
3.3. S7-200	10
3.3.1. Introducción	10
3.3.2. Elementos de Operación e Indicación	12
4. Protocolos de comunicación	14
4.1. KNX	14
4.1.1. Topología lógica y direccionamiento	14
4.1.2. Estructura de los telegramas	15
4.1.3. Medios de transmisión	17
4.1.4. Mecanismo de resolución de las colisiones	18
4.2. AS-i	19
4.2.1. Estructura	20
4.2.2. Características	21
5. Realización de un panel de practicas	22
6. Programación	25
6.1. Programación de los elementos de <i>KNX</i>	25

6.2. Programación de <i>Logo!</i>	27
6.3. Programación de <i>S7-200</i>	27
7. Ejemplo de integración	30
7.1. Sistema de calefacción	30
7.1.1. Descripción del funcionamiento	30
7.1.2. Componentes	31
7.1.3. Programación de los elementos de la instalación	32
7.1.4. Mejoras	35
8. Conclusiones	37
9. Bibliografía	39

Capítulo 1

Introducción

En la facultad de ITI (Ingeniero Técnico Industrial) de la Universidad de Extremadura en Badajoz se da la asignatura de *Infraestructuras Inteligentes* que trata la investigación de la planificación de servicios para viviendas y edificios.

La domótica es un subconjunto de las Infraestructuras Inteligentes, porque esa se refiere principalmente a las viviendas, proporcionando un control centralizado de las instalaciones, desde cualquier punto y también desde fuera del edificio. Por otro lado un bloque de oficinas puede requerir muchos mas servicios que una simple automatización de las instalaciones eléctricas y de calefacción; de hecho se pueden compartir, y con eso aprovechar más, muchos servicios comunes, cuales reprografía, centro de calculo, guardería, vigilancia nocturna, bar, telefonía . . .

La domótica centra sus objetivos en hacer la vivienda más confortable, mas segura, mas ajustable a las variaciones de las necesidades de sus inquilinos y a conseguir un mayor ahorro energético, encendiendo y apagando los equipos, según la efectiva necesidad de funcionamiento. Esa tiene que adaptarse cuanto mas a las costumbres de sus usuarios: no requerir aprendizaje de instrucciones ni suministrar informaciones demasiado complejas, que no puedan ser entendidas sin una formación previa. En cierto cual modo, los componentes tienen que entender lo que el usuario quiere, y hacerlo, y de este concepto efectivamente nacen las centrales de autoaprendizaje, las cuales aprenden, de las costumbres de los usuarios, la solución que ellos prefieren en cada ocasión.

Otro escenario es lo de los edificios más complejos, como pueden ser bloques de oficinas, escuelas y otros edificios públicos, que requieren por su funcionamiento unos servicios esenciales (agua, electricidad, red informática, . . .) y otros complementarios, pero también importantes. Aprovechar cuanto mas de los servicios comunes es un requerimiento fundamental, ya que, en

cualquier empresa, se quiere bajar los costes y subir los servicios. De hecho, la centralización de estos, casi siempre lleva a una mejor relación precio - ganancia, pero conlleva necesidades de control múltiple, facturación y otros.

En este tipo de edificios, se juntan entonces las necesidades de los usuarios con las de instalaciones de gran tamaño, así que también la automatización de estas tienen que encontrar un balance entre sencillez de uso y fiabilidad de control. Se trata de un compromiso sin soluciones sin realizar un sistema ad-hoc pensado exclusivamente para cada proyecto, lo que lleva a unos costes muy alto de proyecto y de mantenimiento, debido a la particularidad del sistema. Una solución más aceptable sería la de utilizar productos estándar, que ya existen en el mercado, cuales los dispositivos domóticos *KNX* y los *API*. De este asunto se ha concretizado la investigación tratada en este trabajo.

Capítulo 2

Objetivos

El objetivo de esta investigación es de encontrar una solución que permita la intercomunicación entre dispositivos de mando y actuadores basados en el protocolo *KNX* con uno o más autómatas industriales que comuniquen en con un puerto *AS-i*.

Esta necesidad nace de los distintos requerimientos que tienen los aparatos en el entorno del usuario y los colocados en las salas de máquinas de un edificio de cierto tamaño. Los primeros tienen contacto directo con el usuario, lo cual no tiene por que saber nada sobre el funcionamiento de las instalaciones de su vivienda u oficina. Por eso el sistema de mando debe ser sencillo, amigable y casi transparente para el; en muchos casos factores estéticos son la razón principal que guía la elección de estos componentes. Por otro lado, estos edificios casi siempre disponen de una central térmica para la calefacción en invierno. Estas centrales están constituidas por una o mas calderas, bombas, válvulas y tuberías de conexión. Todo este conjunto tiene que ser controlado por un aparato muy fiable, para minimizar el riesgo que un fallo en el control del sistema provoque averías a todo el sistema, falta de continuidad de servicio y, aún peor, posibilidad de incendio. En esta situación es fundamental elegir un componente de control diseñado específicamente para este tipo de tarea, o sea el control de un proceso *industrial*. Es así que el segundo elemento de este estudio entra en el escenario.

Claramente se puede suponer, y de hecho ya se ha comprobado, que los dos sistemas son tan distintos entre ellos, tanto en el funcionamiento que en los objetivos, que no tienen una forma de intercomunicar uno con otro. Se ha entonces querido investigar la posibilidad de encontrar una solución a esta falta, utilizando como puente un tercer elemento, que fuese capaz de comunicar separadamente con los dos y que intentara ser nudo de intercambio para los dos. Este elemento se ha encontrado en el *Logo!* de *Siemens* data compatibilidad con las especificaciones y su bajo coste.

Capítulo 3

Dispositivos en examen

En este capítulo se describirá brevemente los tres distintos sistemas domóticos y sus características peculiares.

3.1. KNX

3.1.1. Introducción

KNX es un sistema domótico descentralizado en el que todos los mecanismos (sensores, actuadores y dispositivos de control) comunican entre ellos por medio de un bus dedicado, aun que es posible vehicular la comunicación a otras tecnologías (ondas portadoras, infrarrojo, ethernet, ...).

3.1.2. Características

En *KNX* cada elemento que compone el sistema está dotado de un microprocesador, esto permite que la instalación no requiera una unidad central de procesamiento, ya que todos los aparatos son “*inteligentes*” y pueden tomar decisiones de forma autónoma. El mayor coste individual de cada elemento, debido al ser microprocesado, es compensado con una mayor tolerancia a los fallos, dado que no hay necesidad de una unidad central y una avería en un aparato solo le afecta a su función y no a toda la instalación. El punto de debilidad en este caso puede ser la línea de bus, que, en caso de interrupción, dejaría inutilizables todos los componentes en el tramo sin fuente de alimentación.

El bus es el principal medio de intercambio de datos y de informaciones. Está formado por dos pares trenzados de $0,8mm^2$ revestidos con un apanallamiento metálico y uno plástico que permite la instalación en proximidad

de cables eléctricos en tensión. El par ROJO Y NEGRO es encargado de suministrar la alimentación a los dispositivos (*DC*) y de transmitir las órdenes (*AC*) mientras que el par *amarillo y blanco* es disponible para aplicaciones de reserva (voz, baja tensión de seguridad, ...). El apantallamiento, la trenzamiento y la *transmisión diferencial* de la señal son las técnicas empleadas para reducir las interferencias debidas a los campos electromagnéticos que frecuentemente se encuentran al trabajar en la cercanía de líneas de potencia.

La alimentación del bus se realiza por medio de dos componentes la *fuentes de alimentación* y el *filtro de acoplamiento*, que a veces pueden ser incorporados en un único elemento. La primera tiene la función de convertir la tensión de red de $230VAC$ en baja tensión de seguridad de $29VDC$, asegurando un aislamiento galvánico entre las dos. Hay fuentes de distinta potencias, entre $160mA$ y $640mA$, para elegir según el número y la potencia absorbida por los elementos en la línea. Es posible poner hasta dos fuentes de alimentación en cada línea, a condición que la distancia entre ellos sea mayor de 200m. El filtro de acoplamiento precisa la función de depurar la tensión de salida de la fuente y, además de esto, está diseñado para tener alta impedancia en alta frecuencia, de modo da confinar los telegramas dentro de la línea. En el filtro hay un interruptor de reset que envía un telegrama que resetea las programaciones y las direcciones físicas de todos los aparatos de la línea.

Los aparatos EIB están preparados para funcionar a una tensión de 24V y toman del bus una potencia constante entre 150 y 200 mW. Los elementos que requieran más potencia tendrán que tener una alimentación auxiliar. Es este el caso, por ejemplo, de las pantallas táctiles o de los módulos de telefonía.

3.2. Logo!

3.2.1. Introducción

Entre los sistemas de automatización centralizados, uno de lo mas sencillos, económicos y versátil es el *Logo!* de SIEMENS. Se trata de un nanoautómata de tipo compacto dotado, en su versión básica, de 8 entradas y 4 salidas, una pantalla de visualización y unas teclas para la programación y configuración. A esa unidad se pueden añadir unos módulos de expansión o de conexión para aumentar el número de *E/S*.

3.2.2. Características

Logo! es un módulo lógico que está pensado para el montaje en carril *DIN* o para el montaje superficial.

Logo! existe en varias versiones, bien en tensión de red de 230V o bien en versión de baja tensión 12/24V AC/DC. En este proyecto se ha utilizado la versión a 24V AC/DC para razones de seguridad y de uniformidad con la alimentación del módulo *CM EIB/KNX*. La tensión de alimentación se elige también en función de la máxima soportada por los sensores, ya que las entradas de *Logo!* tienen un umbral de conmutación proporcional a esa. Y también hay el vínculo de las entradas analógicas, que están disponibles sólo en las versiones en DC.

Las salidas de *Logo!* pueden ser contactos a relés libres de tensión o transistores. A pesar de la corriente soportada por los relés, se recomienda el uso de telerruptores en el caso de cargas inductivas o muy importantes, para que no se gasten rápidamente los contactos. De todas formas, existe un módulo de salidas expresamente pensado para el control de motores y carcas hasta 20A.

El funcionamiento de *Logo!* es derivado de lo que se usa en los autómatas industriales: hay una fase de lectura de las entradas y memorización de sus estados en la memoria interna, la elaboración del programa con los datos en memoria y al final la activación de las salidas según las condiciones del caso. Por esta razón, una variación rápida (inferior a la longitud del tiempo de ciclo) de una entrada puede no ser detectada por *Logo!*, por eso hay *entradas rápidas* a las que se tiene que recurrir en estas ocasiones. *Logo!* tiene dos estados de funcionamiento, no hay ningún estado de “error”, como es usual en los demás autómatas. En el estado de *STOP*: no se leen las entradas, no se ejecuta el programa y las salidas están desconectadas. Mientras que en *RUN* *Logo!* lee el estado de las entradas, utiliza el programa para calcular el estado de las salidas y las activas y desactivas de consecuencia.

3.2.3. Elementos de Operación e Indicación

Logo! tiene una pantalla de 4 líneas y 12 columnas (excepto en la versión *Pure*) que permite, por medio de 6 teclas en el frontal del dispositivo, de visualizar, configurar y parametrizar el funcionamiento del autómata. En el display es posible visualizar también mensajes accionados por eventos y valores instantáneos de las magnitudes en objeto, además de el estado de todas las entradas y salidas, bien digitales o analógicas. Las seis teclas, cuatro flechas direccionales mas la confirmación y el escape, permiten navegar en los menús del dispositivo y también de programar en *FUP* sin necesidad de

usar el ordenador.

La versión *Pure* no dispone de display ni de teclas, solo tiene un led *Rojo* y *Verde* que indica respectivamente el estado de *STOP* o de *RUN* y la programación y configuración solo se puede hacer por medio de *LogoSoft*.

El mismo led se encuentra también en los módulos de ampliación aun que con un significado distinto: el color *Verde* indica que hay comunicación con el dispositivo colocado a la izquierda, el *Rojo* que no la hay y el *Anaranjado* que está todavía en una fase de inicialización.

Los módulos de comunicación, además de este, tienen un segundo led que indica el estado de comunicación con el respectivo bus: *Verde/Comunicación*, *Rojo/Fallo* y *Anaranjado/Dirección a asignar* (en el caso de *AS-i*) o *Anaranjado/Modo de Programación* (en el caso de *KNX*).

3.2.4. Módulos de expansión y de comunicación

Logo! tiene una *Imagen de Proceso* de 24 entradas y 16 salidas, bien digitales o analógicas. En su versión básica las primeras 8 entradas y 4 salidas físicas ya son presentes, pero se pueden extender conectando en cascada unos módulos de expansión o de comunicación. Los primeros permiten extender el número de *E/S* físicas, bien digitales o analógicas mientras que los segundos permiten la conexión del autómatas a un bus *KNX* o *AS-i*. Los módulos de expansión digitales pueden ser de 8 entradas y 8 salidas o de 16 entradas y 16 salidas. Los módulos analógicos pueden ser de 2 entradas o de dos salidas, bien en tensión (0-10V) o en corriente (0/4-20mA), o 2 entradas para PT100.

El módulo de comunicación para el bus *AS-i* permite transferir 4 bits de datos desde *Logo!* hacia el bus y viceversa. El número de las entradas que le corresponde depende de la posición del módulo con respecto a *Logo!* como representado en la tabla 3.1. Este módulo solo actúa solo como esclavo, de forma que no es posible interconectar unos *Logo!*s entre ellos sin un maestro, que en esta realización está representado por el módulo *AS-i* del *S7-200*. Para que el módulo funcione en el bus es preciso asignarle una dirección unívoca, bien con una unidad de direccionamiento a través del conector hembra de direccionamiento o bien por el software *Step7 Micro Win*, en el rango entre 1 y 31.

El módulo de comunicación *CM EIB/KNX* permite la conexión de *Logo!* a un bus *KNX*. Eso completa la imagen de proceso de *Logo!* de forma que todas las *E/S* que no existen físicamente en *Logo!* están disponibles como objetos de comunicación en el bus *KNX*. Por este motivo, el *CM EIB/KNX* se tiene que poner siempre como último módulo, ya que cualquiera que se pusiese a su derecha, ya no tendría ninguna correspondencia libre en la imagen de proceso. Las *E/S* disponibles en el bus *KNX* serán como mucho

Cuadro 3.1: Correspondencia lógica

<i>Logo!</i> device	AS-i interface
Entradas	Bits de datos de salida
I_n	D0
I_{n+1}	D1
I_{n+2}	D3
I_{n+3}	D4
Salidas	Bits de datos de entrada
Q_n	D0
Q_{n+1}	D1
Q_{n+2}	D2
Q_{n+3}	D3

16 entradas digitales, 12 salidas digitales y 8 entradas analógicas, y se tendrá que especificar manualmente en *ETS* cuales estarán asignadas al módulo de comunicación y cuales a *Logo!* y a sus módulos de expansión, como se puede ver en la figura 3.1. Las entradas analógicas que no se utilicen tienen que ser específicamente marcadas como “no utilizado”, si se usaran se tendría que especificar para cada una el tipo de datos como *EIB-flotante* (8 bits con signo) o *Porcentaje* (8 bits sin signo) como se puede ver en figura 3.2.

Todos los módulos de ampliación se colocan a la derecha de *Logo!* uno detrás de otro en el mismo carril *DIN*. Quitando una pequeña tapadera que los envoltorios tienen en el lado derecho y deslizando la corredera del bloque a añadir, se realiza sencillamente la conexión del nuevo módulo al conjunto. Los módulos de ampliación tendrán, además de sus entradas y salidas, unos bornes de alimentación excepto el módulo *AS-i* que se alimenta por si mismo desde el bus. Este último, de hecho, puede ser fuente de problemas, porque en caso de bus failure el módulo *AS-i* se quedaría desalimentado y los módulos a su derecha se quedarían aislados.

3.3. S7-200

3.3.1. Introducción

Otro sistema de automatización centralizado es el que se puede realizar con el *API*. Hasta ahora hemos estudiado los autómatas en general sin particularizar sobre ninguno en especial. A partir de este tema nos centraremos en

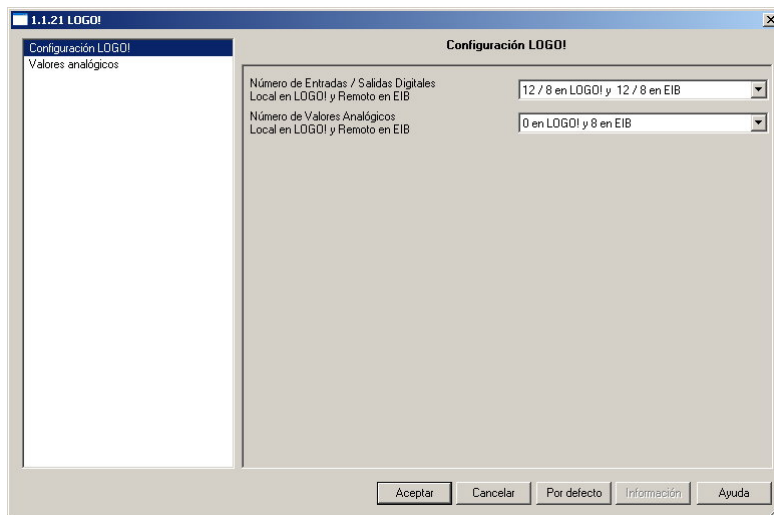


Figura 3.1: Ventana configuración módulo *CM EIB/KNX* en *ETS*

los autómatas del fabricante SIEMENS, denominada también SIMATIC. Como ya comentamos, dentro de la familia SIMATIC existen tres gamas:

- Gama Alta → *S7-400*
- Gama Media → *S7-300*
- Gama Baja → *S7-200*

Dentro de la gama baja, que es donde nosotros vamos a trabajar, se pueden escalar los autómatas, según sus prestaciones, en los peldaños *S7-21x* y *S7-22x*. Nosotros, particularmente usaremos la *CPU S7-214* o la *S7-224*, cuya mayor diferencia entre ellas está en que el segundo posee casi el doble de memoria que el primero, ocupando la mitad.

Estos autómatas se denominan compactos, debido a que, dentro de la misma carcasa se encuentran la fuente de alimentación, la UC, los interfaces de *E/S* integrados y los puertos de comunicación. como interfaces integrados poseen:

- 14 Entradas Digitales Integradas
- 10 Salidas Digitales Integradas.

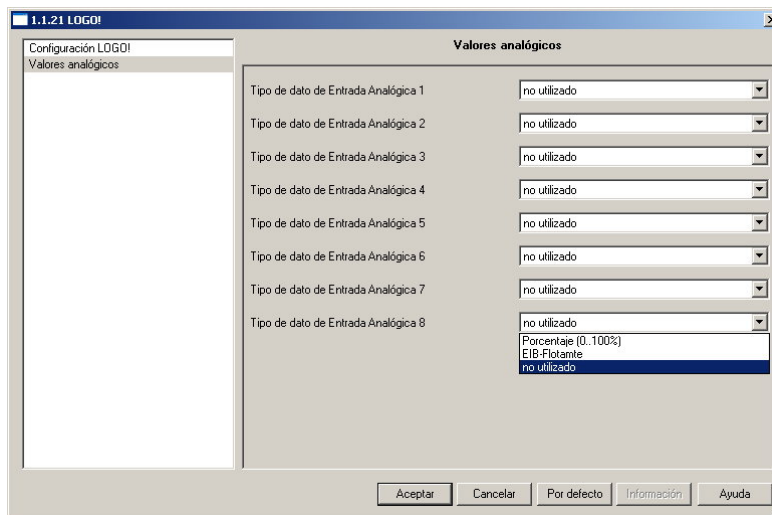


Figura 3.2: Ventana configuración entradas analógicas módulo *CM EIB/KNX* en *ETS*

3.3.2. Elementos de Operación e Indicación

En la siguiente figura puede verse la carcasa común a los modelos de la gama baja de SIMATIC, *S7-200*. Sobre ella podemos resaltar distintas partes, como son:

- El *interfaz PPI* (punto por punto) que mediante un adaptador *rs232* nos permite la conexión con la consola de programación
- El *conmutador de mando*, que nos permite elegir entre los modos de operación *run*, *stop* y *term*. Cuando nos encontramos en la posición *term*, el cambio de modo de operación se efectúa desde la consola de programación.
- Los *indicadores de modo de funcionamiento*, que se iluminarán en *verde* en el modo *run*, *amarillo* en modo *stop* y en *rojo* si se produce algún tipo de error.
- Las *entradas digitales*, que disponen de 14 terminales accesibles estructuradas en bloques de tamaño byte. Con lo cual tendremos 8 bits en el módulo de entrada 0 y 6 bits en el módulo de entrada 1. Los dos bits que falta se pierden irremediabilmente.
- Las *salidas digitales*, compuesta por un módulo de 8 bits (módulo 0) y un módulo de 2 bits. Nótese que, aunque ahora se pierden 6 bits, ésta si son recuperables, usándolos como registro o marcas internas.

- El *indicador de estado* de las *E/S* digitales integradas, compuesto por un conjunto de ledes, cada uno de ellos correspondiente a una entrada o una salida.
- *Potenciómetros analógicos*, usados normalmente para fijar valores de referencia. Su estado se refleja en determinadas posiciones de la memoria de marcas.
- La ranura para cartucho EEPROM, mediante la cual podemos insertar al autómatas una memoria de este tipo (en la cual tengamos un programa de usuario a transferir, previamente depurado) o acoplarle una batería para el mantenimiento de los datos.

En la figura siguiente podemos ver los terminales correspondientes a la alimentación y como este autómatas incluye su propia fuente, podemos conectarlo directamente a la red de 230V AC a través de estos terminales.

Además se incluyen dos salidas a 24V para alimentar algún dispositivo, como podrían ser sensores de entrada o módulos de extensión.

Obsérvese la peculiar conexión de los terminales de salida. al tener tres grupos diferentes y ser salidas a relé, se permite el uso de tensiones tanto continuas como alternas y, además, pudiendo tener valores diferentes de tensión entre los tres grupos.

Como para las entradas necesitamos un nivel de tensión de referencia y el autómatas presenta un par de terminales accesibles, podemos usar estos mismos, como se muestra en el conexionado de la figura siguiente.

Capítulo 4

Protocolos de comunicación

4.1. KNX

4.1.1. Topología lógica y direccionamiento

En una instalación *KNX* cada mecanismo posee un par de direcciones:

- **Dirección Física:** identifica unívocamente al mecanismo dentro del sistema
- **Dirección de grupo:** identifica el/los grupo/s en que se hayan agrupado mecanismos que actúen conjuntamente

En *KNX* se definen la *Topología Lógica*, que describe la configuración de una instalación desde punto de vista del flujo de informaciones y la *Topología Física* que se refiere a la configuración y disposición física de los dispositivos y medios de comunicación.

Topología física, independiente del medio

La topología lógica de una instalación *KNX* se estructura en una Línea Dorsal, unas Líneas principales y unas secundarias. Esta subdivisión se refleja, aun que no necesariamente, en el direccionamiento físico de los mecanismos.

En una instalación hay siempre una y sola línea dorsal.

Las líneas principales están jerárquicamente dependientes de la línea dorsal y están numeradas de 1 a 15, ya que la dorsal tiene numero 0. De cada una de estas se derivan las líneas secundarias, 15 por cada principal. Las interconexiones entre ellas se obtiene por medio de *Acopladores de Área* en la dorsal y *Acopladores de línea* en las líneas principales. Los acopladores de área

tienen dirección de aparato 0 y dirección de línea 0, mientras los acopladores de línea tienen dirección de aparato 0. Sólo los acopladores pueden tener dirección de aparato 0 y pertenecen a la línea jerárquicamente menos importante de las dos que unen. Cada línea principal puede tener un solo acoplador de área y cada línea secundaria puede tener un solo acoplador de línea.

Un área está definida como una línea principal y todas sus líneas secundarias. El área estará numerada según su línea principal. La dorsal no constituye un área. Una subred es una parte cualquiera de esta topología, que tenga el mismo número de línea principal y secundaria.

Los aparatos *KNX* pueden ser conectados en cualquier lugar en esta topología, hasta 255 en cada subred.

Direccionamiento

Cada mecanismo tiene una dirección física unívoca en toda la instalación. Eso está formado por tres números de 0 a 15 que identifican el área, la línea y el número de aparato en la forma (1.1.1). Cada elemento está también asociado por lo menos a una dirección de grupo, que define una conexión lógica entre aparatos que están relacionados a una cierta lógica de funcionamiento. La dirección de grupo tiene forma de (1/1/1) según una subdivisión de grupo-principal/grupo-intermedio/grupo-funcional.

Este direccionamiento permite comunicar con un aparato específico, enviando los datos a su dirección física, o bien comunicar al mismo tiempo con aparatos relacionados entre ellos, enviando un telegrama de multicast a la dirección de grupo que tienen en común: Se evita así de sobrecargar la red de telegramas repetidos.

Un ejemplo para ejemplificar la utilidad del direccionamiento de grupo es un pulsador que enciende unos puntos de luz. Cada luz se puede encender por separado con un pulsador, enviando un telegrama a cada luz con su dirección física o bien se pueden encender todas a la vez, enviando el mismo comando a la dirección de grupo que las unen.

4.1.2. Estructura de los telegramas

En el estándar *KNX* el intercambio de información entre aparatos ocurre por medio de telegramas, que envían datos en forma de bits. En el mismo instante en que un aparato tiene que comunicar con otro se intenta enviar un telegrama en el bus, para obtener el intercambio de la información.

El elemento básico de información es la *palabra*. Esa consta de 11 bits: un bit de inicio (B1) de valor 0 binario, ocho bits de datos de menor a mayor peso, un bit de paridad par (BF) y un bit de stop (BP) de valor 1 binario.

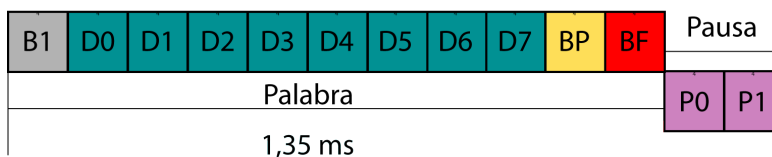


Figura 4.1: Palabra en el sistema *KNX*

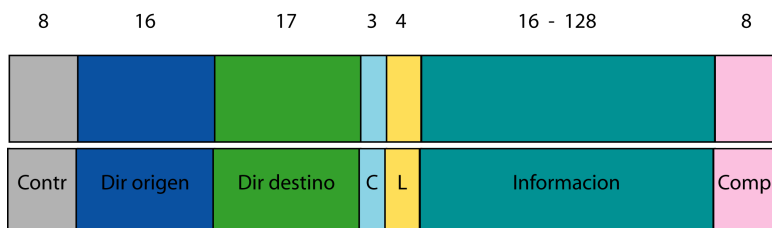


Figura 4.2: Campos del telegrama en el sistema *KNX*

Varia palabras se agrupan en un telegrama, que consta de siete campos. El telegrama puede contener entre 9 y 24 palabras.

El primer campo del telegrama es el de *control*, que ocupa una palabra y contiene información sobre la tipología del mensaje y su prioridad. Le siguen los campos de *Dirección de origen* (dirección física) y *destino* (puede ser una dirección física, pero usualmente será una dirección de grupo), que ocupan dos palabras cada uno y contienen la dirección del mecanismo emisor y receptor del telegrama respectivamente. Seguido de éstos, están los campos de *routing* y *longitud* de la información útil, de 3 y 4 bits respectivamente. El primero indica el numero de saltos de enrutamiento permitidos mientras el segundo indica la longitud del campo *datos* del telegrama (entre 0 y 15 byte). La información útil es el siguiente campo y contiene los datos propios del sensor o actuador. Por último un campo de comprobación suma por paridad vertical impar. La longitud máxima total del telegrama es de 23 byte.

El funcionamiento es el siguiente: el aparato emisor coloca el telegrama en el bus con la dirección del mecanismo receptor. Todos los aparatos reciben el mensaje (excepto si hay un filtrado activo en los acopladores) y el destinatario reconoce su dirección y capta el telegrama del bus, lo interpreta y envía un acuse de recibo, indicando el éxito o no de la transmisión. Si un aparato detecta un error en el telegrama recibido, envía al mecanismo de origen un mensaje de error que activa la retransmisión hasta 3 veces del mensaje original. Si un aparato envía un telegrama y no recibe un acuse de recibo entre un cierto tiempo, interpreta el evento como “mensaje no recibido por el destinatario”. Si el destinatario no puede recibir el mensaje inmediatamente, entonces puede enviar un mensaje de “ocupado” que retrasa el nuevo

re-envío del telegrama. En caso de errores y mensajes urgentes, existe un sistema de prioridades en la transmisión: las alarmas tienen prioridad sobre los telegramas corrientes.

4.1.3. Medios de transmisión

El estándar *KNX* permite el uso de distintos medios de transmisión, también entre la misma instalación:

- PL (Powerline/Ondas portadoras) Este medio de comunicación utiliza la misma red de alimentación de 230V para enviar los telegramas. La modulación es FSK, la transmisión asíncrona, half/duplex bidireccional: PL110 a 110 kHz con velocidad de 1200 bps (desde EIB) o PL132 a 132 kHz a 2400 bps (desde EHS)
- RF (Radiofrecuencia) Los aparatos *KNX* que admiten este medio de transmisión envían los telegramas por medio de señales radio en la banda de 868MHz, con potencia máxima de 25mW, a una velocidad de 32000 bps. La modulación es FSK, la transmisión asíncrona, half/duplex bidireccional.
- IP (Ethernet) Así como explicado en las especificaciones *KNXnet/IP*, los telegramas *KNX* se pueden encapsular en telegramas *IP* de forma que las redes IP se puedan usar para encaminarlos y entunelarlos. Una línea ethernet (o fast-ethernet) puede constituir la dorsal de una instalación *KNX*.
- TP (Twisted Pair/Par trenzado) es el medio principal y mas común en las instalaciones. Según varias topologías de red (excepto anillos), permite la conexión de los aparatos y el alimentación de los mismos. La transmisión asíncrona, half/duplex bidireccional es TP0 a 2400 bps (desde BatiBus) o TP1 a 9600 bps (desde EIB).

Comunicación en par trenzado

El par trenzado es el elemento mas común en las instalaciones *KNX* ya que es muy económico y suministra alimentación y también funciona como bus de comunicación de todos los elementos. La capa física de la pila especifica que el cable tiene que ser un par trenzado y apantallado de 0,8 mm de diámetro y con un tiempo de retraso por kilómetro de 9 μs . La distribución puede ser en línea, en árbol o en estrella o cualquier combinación de estos, menos el anillo, siempre que se respeten unos parámetros fundamentales:

- cada línea puede ser larga como mucho 1000m
- en cada línea se pueden poner como mucho 64 mecanismos
- la distancia máxima entre dos mecanismos es de 700 m
- la distancia máxima entre un mecanismo y una fuente de alimentación es de 350 m
- en cada línea se pueden poner como mucho 2 alimentadores distanciados por lo meno 200 m uno de otro
- Si mas de 30 mecanismos están distribuidos en menos de 10 m, la fuente de alimentación tiene que estar en la cercanía

Estas condiciones están determinadas por los parámetros de tolerancia de funcionamiento de los elementos. La tensión del bus es de 28V entonces se han tenido en cuenta el consumo de cada aparato para calcular la caída de tensión máxima admitida para que todo funcione correctamente. Así también se ha determinado la longitud máxima da la línea dependiendo del la velocidad de transmisión y de la longitud de la palabra, ya que se tiene que asegurar el funcionamiento del protocolo CSMA/CA que garantiza la correcta gestión de los telegramas en el caso de acceso múltiple y contemporáneo de mas dispositivos.

La transmisión de bits a lo largo del par trenzado se obtiene con una superposición te una tensión positiva o negativa al valor constante de alimentación. La lógica es negativa, es decir que una ausencia de perturbaciones significa el envío de un valor “1” binario, mientras que una fluctuación de la tensión corresponde a un valor “0”. La perturbación, actuada por medio de bobinas y condensadores, correspondiente a una subida y a una bajada de 5 V, es simétrica con respecto a la tensión de alimentación, así de garantizar un valor promedio igual a la tensión de alimentación para no variar los parámetros de red en función de los bits enviados.

La longitud de cada bit es de 104 μs y una palabra en total dura 1,35 ms.

4.1.4. Mecanismo de resolución de las colisiones

En *KNX* cada aparato puede enviar datos en cualquier momento. Data la ausencia de token que reglamenten el derecho de habla en el medio, un mecanismo que necesite enviar datos tiene que escuchar si el medio de transmisión es libre y en este caso es automáticamente autorizado a enviar su telegrama. Comprobar que el medio sea libre, no garantiza que no ocurra el envío contemporáneo desde dos aparatos, de dos telegramas distintos: esto

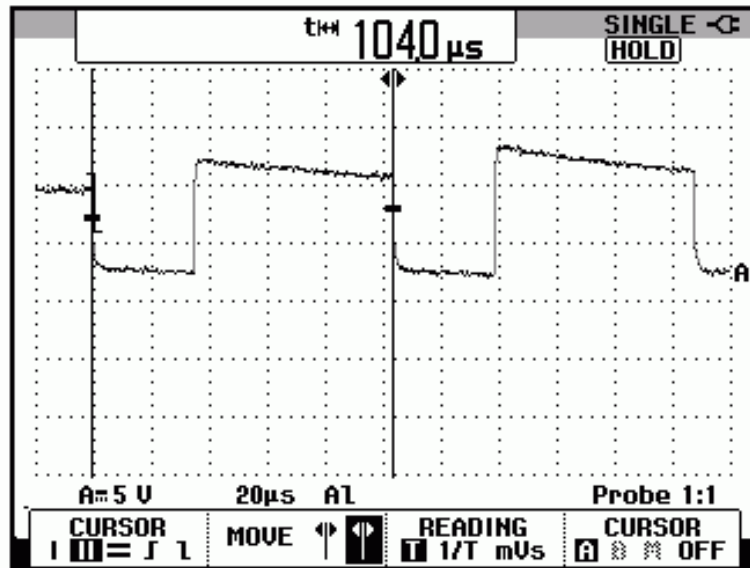


Figura 4.3: Bits en el osciloscopio

es cuanto mas probable al aumentar del número de elementos por segmento y a la distancia entre ellos. En este caso, el sistema de Collision Avoidance permite de solucionar fácilmente el problema: si dos mecanismos emiten al mismo tiempo un bit 0, el resultado es un bit 0 en el bus. Igualmente ocurre si los dos emiten al mismo tiempo un bit 1. Por el contrario, en el caso que los dos envíen dos bits distintos, hay una colisión y de hecho el bit 0 (que corresponde a una perturbación) tiene la prioridad. Los aparatos escuchan al bus al mismo tiempo que transmiten, por tanto, al producirse una colisión, el elemento que estaba enviando un bit 1 abortará la comunicación para intentarla más tarde.

4.2. AS-i

AS-i (Actuator Sensor Interface) es un sistema de bus de campo que ha sido creado para resolver el problema de cableado de sensores y actuadores de una planta de producción.

Cualquier instalación industrial necesita sensores de proximidad, de nivel, de caudal y de temperatura, para medir los parámetros fundamentales del proceso a controlar. Por otro lado, los datos medidos y procesados tienen que proceder desde el *API* hacia los actuadores que controlan motores, resistencias, válvulas . . . La solución más antigua consistía en cablear cada sensor y cada actuador con un par de hilos hasta el bloque de entradas y salidas del



Figura 4.4: Elementos básicos de una instalación *AS-i*

autómata. Esta solución, aunque sencilla conceptualmente, comportaba un alto coste de montaje y muchas dificultades en averiguar eventuales averías. Se ha preferido entonces encontrar otras soluciones: el bus. *AS-i* es un bus de campo de estándar abierto que permite el intercambio de datos analógicos y digitales.

4.2.1. Estructura

Un sistema *AS-i* está compuesto por cuatro elementos fundamentales: una fuente de alimentación, un maestro, un esclavo y el cable de conexión.??

La fuente de alimentación suministra la tensión de 30V a todo el bus, de forma que se puedan alimentar los aparatos conectados. Esa tiene un filtro que garantiza el desacoplamiento entre alimentación y datos que circulan en el mismo cable.

El maestro (que en nuestro caso es el *S7-200*) está encargado de la gestión del bus. Es el que, por medio de un polling cíclico, efectúa la transferencia de los cuatro bits de entradas y de los cuatro de salidas desde cada uno de los esclavos. Este procedimiento garantiza que cada esclavo tiene un tiempo limitado para suministrar sus datos y no pueda ralentizar todo el proceso, lo que llevaría al riesgo de no garantizar las operaciones en *tiempo real*. El maestro está encargado también de diagnosticar el mal funcionamiento de los esclavos, y señalar las averías detectadas para que se pueda arreglar el problema.

Los esclavos más usualmente son nudos donde los sensores y actuadores físicos se acoplan al bus. Esos pueden ser termopares, sensores de presión,



Figura 4.5: Detalle del cable *AS-i*

válvulas o telerruptores que ya disponen de un acoplador al bus. En cada red *AS-i* se pueden conectar hasta 62 dispositivos distintos.

El cable *AS-i* proporciona alimentación a los dispositivos y también es el medio de propagación de las señales de información. Eso está compuesto por un par de hilos polarizados, no trenzados ni apantallados, cubiertos con un recubrimiento de goma amarilla que tiene una forma de trapecio y con una ranura asimétrica que permite la correcta detección de la polaridad (véase la figura 4.5). La conexión eléctrica de un aparato se obtiene por medio de unas tomas vampiro, que pinchan el cable y garantizan el contacto con los conductores. La ranura en el cable impide que la toma se instale de manera incorrecta. Cables de la misma factura y distintos colores (rojo y negro) están destinados a la alimentación de aparatos que requieran una tensión distinta da la del bus.

4.2.2. Características

AS-i ha sido concebido como un bus de campo, es decir para funcionalidades de tiempo real, gestión de un flujo pequeño de datos y muchos aparatos distribuidos a lo largo del cable, en una configuración distribuida. El número máximo de 62 esclavos (en la modalidad A/B) corresponde a 496 entradas y 496 salidas que vienen actualizadas cada $20ms$.

La topología de la red puede ser múltiple: línea, árbol o estrella. Estas se pueden componer entre si hasta respetar las especificaciones del estándar, permitiendo así futuras ampliaciones de la instalación.

Capítulo 5

Realización de un panel de practicas

Desde los primeros intentos de afrontar el problema propuesto, ha sido inmediatamente claro que la mejor forma de investigar y presentar los resultados del trabajo abría sido la realización de un *panel de prácticas*. 5.1



Figura 5.1: Vista general del panel

Fundamentalmente el panel debía contener los aparatos que habían sido analizados y proporcionar una imagen muy clara de las distintas disposición de cada uno de ellos y sus interconexiones. Se ha creído oportuno entonces, partir el panel en tres partes: arriba un espacio para los componentes de *KNX*, al centro el módulo lógico *Logo!* y abajo el *S7-200*.

Los materiales para construir el panel han sido proporcionados por el laboratorio de Infraestructuras Inteligentes, y han sido elaborados en el mis-

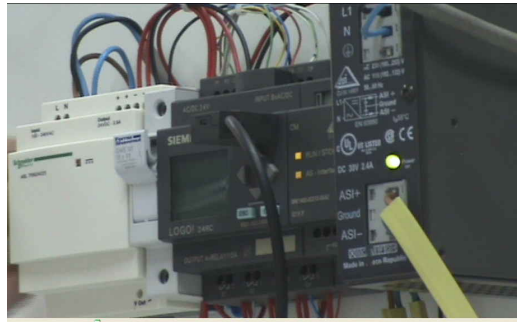


Figura 5.2: Detalle del carril con *Logo!* y las fuentes de alimentación

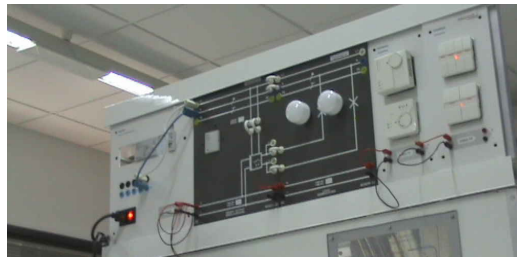


Figura 5.3: Detalle del carril para los módulos de *KNX* del laboratorio

mo laboratorio. El panel está constituido por una tabla de madera en la cual han encontrado sitio, en el lado derecho, dos recortes de metacrilato transparente, en lo cual están dispuestos unos pulsadores, unos leds y unas bornas de conexión. Estos paneles de metacrilato tienen su propia serigrafía que destaca la función de cada componente y, al ser transparentes, permiten comprobar en cada instante la disposición del cableado.

En la parte izquierda del tablero de madera están los dos *API*: arriba el módulo lógico *Logo!* y abajo el *S7-200*. *Logo!* está enganchado a una barra de carril *din* (figura 5.2) y en el mismo carril cogen sitio un alimentador de 24V para *Logo!* y el alimentador de el bus *AS-i*, además que los módulos de comunicación del mismo *Logo!*. Mas abajo, en la misma zona, hay el *S7-200* que sigue enroscado en su panel original en lo cual están disponibles unos pulsadores de mando y el bornero de salida.

Mas arriba, en el panel, hay dos carriles en los que se pueden enganchar todos los módulos de estándar *KNX* disponibles en el laboratorio. De esta forma se ha dado la máxima flexibilidad a la composición que se quiere investigar (figura 5.3).

El panel ha sido pensado también como oportunidad de prácticas para los alumnos de la asignatura. El ejemplo aplicativo en este relato, ha sido diseñado para ser realizado con los elementos disponibles, y ha sido compro-

bado en su factibilidad y funcionamiento. Se cree, de esta forma, proporcionar a los estudiantes unas oportunidades siempre actuales sobre el tema de la domótica, agregando a el temario, los últimos resultados de la investigación.

Capítulo 6

Programación

6.1. Programación de los elementos de *KNX*

El software de programación del bus *KNX* es ETS (EIB Tool Software) 6.1. Eso sirve para la planificación, diseño y puesta en marcha de una instalación *KNX*. Consta de cuatro módulos de aplicación:

- Gestor de Carpetas de Proyectos
- Administración y Diseño
- Instalación y Diagnósis
- Bases de datos *KNX*

El Gestor de Carpetas de Proyectos es un gestor de archivos que permite almacenar ordenadamente los proyectos junto a los que contienen los datos específicos del proyecto y productos utilizados en el mismo.

El Módulo Administración *KNX* permite guardar la información administrativa (direcciones, fabricantes, fechas de inicio y terminación de proyectos, ...). Contiene el módulo interno Diseño, que está dotado de un programa CAD que asiste al proyectista en el diseño de la planta y los esquemas de distribución del bus y presenta la ventaja de que desde este módulo se puede realizar la comprobación de la instalación sin necesidad de cargar los datos en los componentes. Administra también la transferencia a unidades externas de un proyecto diseñado cuando se utiliza para la instalación un PC diferente al de diseño.

Las funciones principales que permite realizar el módulo de Instalación son:

- Cargar las configuraciones deseadas en los aparatos bus

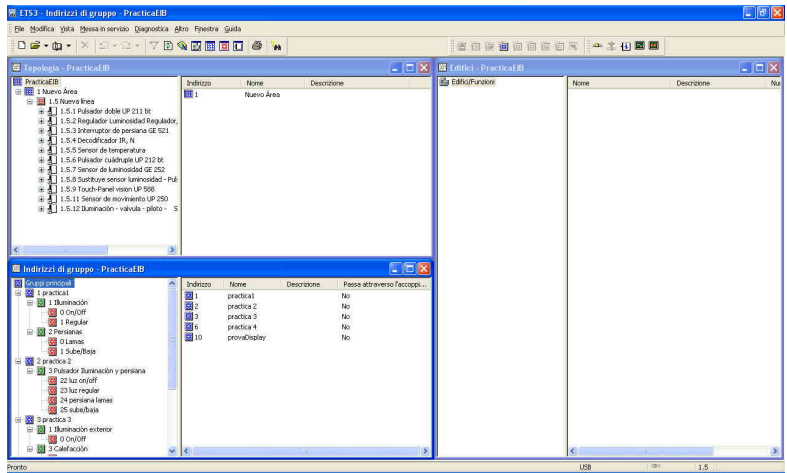


Figura 6.1: Ventana principal de ETS3

- Cambiar datos definidos durante el diseño del proyecto
- Añadir o eliminar aparatos bus
- Diseño de proyectos pequeños
- Gestión de la transferencia del proyecto a otro ordenador

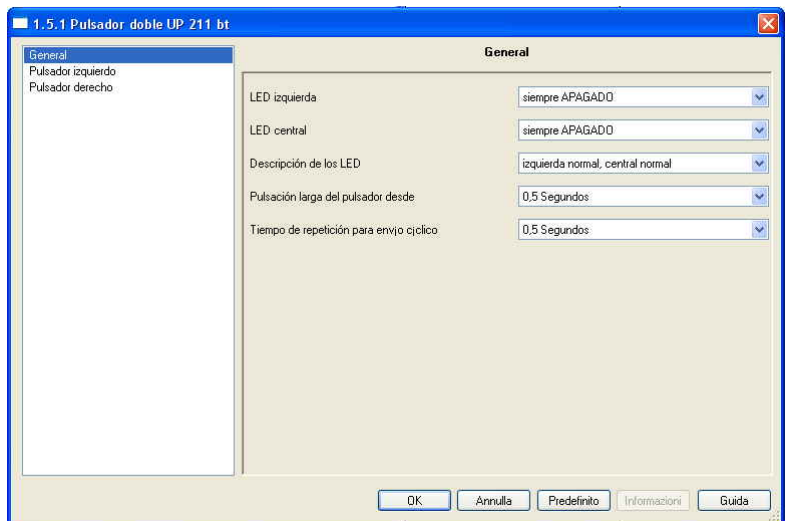


Figura 6.2: Ventana de parametrización de un elemento de la instalación

Contiene, además, un menú desplegable Diagnosis con el cual se puede comprobar el correcto funcionamiento mediante el envío de telegramas de-

nominados seguimientos a una línea, un componente, o bien, a un grupo de aparatos interrelacionados.

Las Bases de datos *KNX* son la base de datos de productos donde están almacenados todos los datos específicos de los productos así como suministrados por los fabricantes y la base de los datos específicos del proyecto que se está diseñando (datos relativos al propio proyecto: dirección, departamento, etc, lista de aparatos utilizados y símbolos utilizados).

6.2. Programación de *Logo!*

Logo! se programa por medio de *LogoSoft* una herramienta gráfica, proporcionada por SIEMENS que permite configurar todas las funciones sin tener que escribir ni una línea de código. El software dispone de dos métodos de programación gráficos *Ladder* o *Bloques Funcionales* (como se puede ver en figura 6.3) que son derivados de los esquemas de contactos el primo y de la lógica digital el segundo. La programación se reduce a colocar los distintos bloques (entradas, salidas, funciones) en el plano y trazar las conexiones lógicas entre ellos, determinando así la lógica de funcionamiento. *LogoSoft* permite testar y depurar el programa gracias a un simulador incorporado en el que se pueden activar manualmente las distintas entradas o los valores analógicos de entrada y ver como evoluciona el sistema, además es posible, conectando el autómatas al ordenador con su cable serial o USB, visualizar en tiempo real los estados internos de cada bloque del programa, una vez en ejecución en el mismo autómatas. El mismo cable permite también la transferencia del programa desde *LogoSoft* hacia *Logo!* y viceversa, de forma que se puede modificar sencillamente un programa ya insertado en el autómatas. Para que *LogoSoft* reconozca y pueda comunicar con el autómatas, este último tiene que encontrarse en modo de *STOP*. Desde el panel de *LogoSoft* es posible cambiar en cualquier momento el modo operativo del autómatas de *STOP* a *RUN*, habilitando así la ejecución del programa.

6.3. Programación de *S7-200*

La programación de *S7-200* se consigue por medio de *Step7 MicroWin*, un programa proporcionado por SIEMENS que permite programar todos los autómatas de la familia *S7-200* según varios métodos: *KOP*, *FUP*, *LAD* y *AWL*, los tres primeros gráficos y textual el último. El software consta de distintos módulos que, juntos, permiten gestionar todo el proceso de automatización que se quiera resolver:

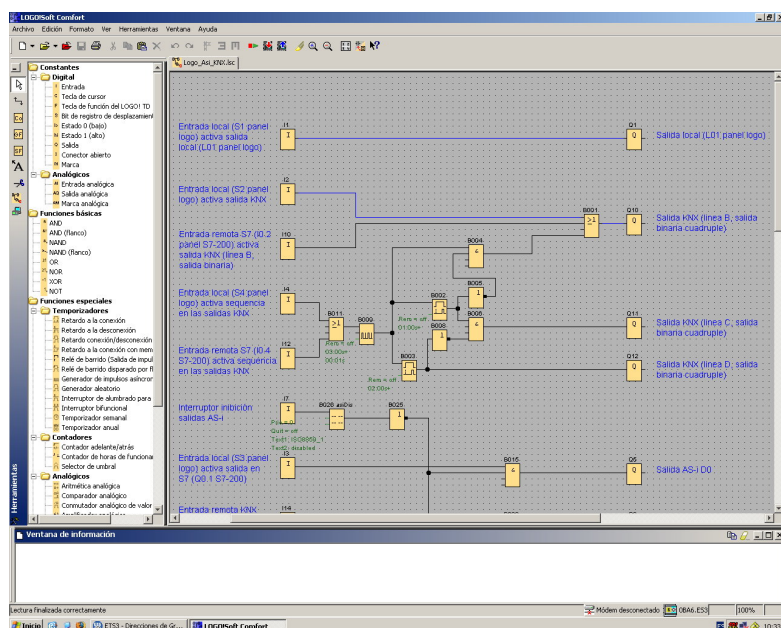


Figura 6.3: Ventana principal de *Logosoft* y bloques funcionales

- Creación y gestión de proyectos
- Configuración y parametrización de los dispositivos
- Diseño y depuración de programas
- Transferencia de programas hacia y desde los dispositivos
- Diagnóstico de correcto funcionamiento de toda la automatización
- Búsqueda y solución de averías

La creación y gestión de proyectos permite almacenar ordenadamente los proyectos junto a los que contienen los datos específicos del proyecto y productos utilizados en el mismo. De esta forma es posible memorizar todas las informaciones de un dato proyecto, sin necesitar archivos externos.

El módulo de configuración y parametrización de los dispositivos permite configurar los puertos, las direcciones de entradas y salidas, la dirección de cada aparato de un bus y los parámetros de los módulos de expansión y comunicación.

El módulo de diseño y depuración de programas es el corazón de la herramienta, ya que permite programar el autómatas según distintos lenguajes de programación bien sean gráficos o textuales. En la misma pantalla es posible depurar el programa, ya que este módulo tiene herramientas que verifican

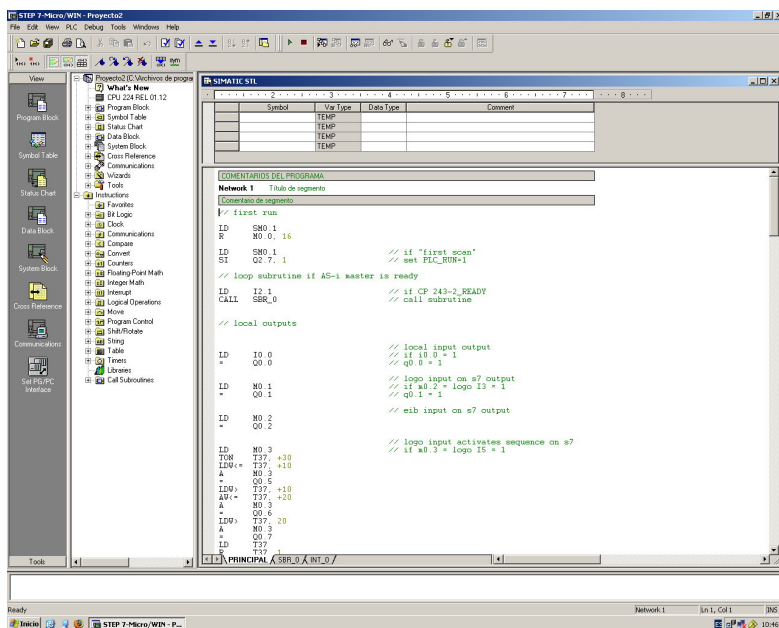


Figura 6.4: Ventana principal de *Step7 Microwin* y código *AWL*

constantemente el programa para evitar los errores más comunes y facilitar la detección de errores.

Desde *Step7* es posible transferir el programa al autómeta y también descargar el programa que ya sea contenido en él, a menos que no haya sido protegido con una contraseña de lectura o escritura.

Desde la pantalla principal se puede controlar también el estado de *RUN* y *STOP* del mismo y visualizar, por medio de otra herramienta, los parámetros internos del *API* mientras tanto que el programa es en ejecución. Se puede así diagnosticar el correcto funcionamiento del programa, según las especificaciones deseadas.

La búsqueda y solución de averías es posible investigando las tablas internas del autómeta, mientras tanto que el programa es en ejecución y también está permitido modificar los parámetros manualmente.

Capítulo 7

Ejemplo de integración

7.1. Sistema de calefacción

Supóngase el siguiente este escenario: en la escuela de Ingenierías Industriales se quiere instalar un sistema experimental para optimizar la calefacción en dos de las aulas, intentando ahorrar combustible y asegurando una temperatura de confort en el horario de clase.

7.1.1. Descripción del funcionamiento

El sistema se compone de unos sensores y actuadores KNX instalados en las aulas, que se encargarán de medir la temperatura (termostatos), seleccionar la temperatura mas adecuada (pulsadores) y activar las válvulas de calefacción (salidas binarias). La caldera estará controlada por un *S7-200* que activará el quemador y dos bombas para la circulación del agua. En la conserjería se colocará un *Logo!* que tendrá la función de interconexión entre los dos sistemas y realizará unas tareas de control y de aviso de posibles averías. Cada aula estará dotada de un termostato de ambiente para el control de la calefacción y de un pulsador en la pared, con el que es posible seleccionar entre las distintas temperaturas de confort de 18, 20 y 22°C o bien apagar la calefacción activando la modalidad de protección contra heladas (7°C). En el pasillo cercano se pondrá una salida binaria doble, controlada con *Logo!*, que activará las electroválvulas de los radiadores.

El *Logo!* colocado en conserjería comunicará con el bus *KNX* y *AS-i* realizando las siguientes funciones:

- Un interruptor horario selecciona automáticamente de lunes a viernes temperatura de confort de 20°C a partir de las 7 de la mañana y la vuelve a poner a 7°C desde las 9 de la tarde, en ambos los termostatos.

El sábado la temperatura de confort de 20°C se activa desde la 7 de la mañana hasta las 2 de la tarde. En el intervalo de tiempo en que la temperatura seleccionada es de 7°C, se enviará cíclicamente, cada 15 minutos, un telegrama que asegure esta condición de funcionamiento.

- Con un interruptor conectado a una entrada del *Logo!* se posibilita la activación de una modalidad “festividad” que configura la temperatura a 7°C en el caso de días de vacaciones o fiesta.
- Se recibirán las señales de activación de los termostatos y se activará la salida binaria que controla la electroválvula del aula correspondiente y la salida del *S7-200* de la respectiva bomba. Además, se activará el quemador si uno cualquiera de los dos termostatos está activo. La electroválvula tiene que abrirse inmediatamente así como el quemador, mientras que la bomba tiene que ser activada con un retraso de 90 segundos. Al apagar, la bomba y el quemador se apagarán inmediatamente, mientras que la válvula se cerrará 20 segundos más tarde. Esta precaución evita los golpes de ariete que ocurrirían al poner en marcha una bomba mientras que la válvula no está completamente abierta.

En proximidad de la caldera se dispondrá un *S7-200* que comunica con el *Logo!* por medio del bus *AS-i*. El *S7-200* usará cuatro salidas, una para el quemador, dos para las bombas y una para una alarma, y dos entradas una para el termostato del quemador y otra para una señalización de avería. El termostato del quemador enciende y apaga el quemador para mantener la temperatura del agua a los 70°C (siempre cuando el quemador esté activo desde el *Logo!*). En caso de que haya señalización de avería, se apagarán las bombas, el quemador y se activará la salida de alarma local y también una salida (piloto) de alarma en el *Logo!* y un mensaje de texto en el display.

7.1.2. Componentes

Para realizar la practica utiliza el panel de interconexión entre *KNX* y *AS-i*. El panel ya está dotado de un *Logo!* con módulo *KNX* y *AS-i* y de un *S7-200* con módulo *AS-i*. En la zona reservada para los dispositivos *KNX* se pondrá una fuente de alimentación, la salida binaria doble y un panel de dos bombillas para visualizar el estado de la salida simulando las electroválvulas, y, además, un panel con dos termostatos y otro con un pulsador cuádruple para una aula y un pulsador doble para la otra.

La relación de los elementos necesarios para la automatización del sistema de calefacción descrito es la siguiente:

- Panel fuente alimentación
- Salida binaria - Siemens - 560-1AB01
- Panel bombillas - Siemens
- Termostato - Siemens - 250-2AB11
- Termostato - Siemens - 252-2AB13
- Pulsador cuádruple - Siemens - 245-2AB11
- Pulsador doble - Siemens - 243-2AB11

En la figura 7.1 se puede ver la disposición de estos elementos a lo largo del edificio.

7.1.3. Programación de los elementos de la instalación

Logo!

La programación se empezará por el *Logo!*, ya que a este elemento le asignaremos todas las funciones de temporización y de control.

En la figura 7.2 se puede ver una posible implementación de las funciones requeridas. Tendremos la única entrada física (I1) en *Logo!*, que es la llave de activación de la modalidad “festividad”. La lógica del programa, utilizando un temporizador semanal y un generador de impulsos, gestionará las temperaturas de confort como deseado. Las demás entradas serán las señales que llegan desde los termostatos (I12 y I14) que indicarán la necesidad de activar la calefacción. Estas entradas serán temporizadas a la conexión o a la desconexión respectivamente para activar las bombas (Q5 y Q6) o las válvulas (Q11 y Q12), mientras que la activación del quemador (Q7) no será temporizada. La alarma de avería, procedente del bus *AS-i* (I9) simplemente activará una bombilla de alarma (Q1) y visualizará el mensaje “Avería caldera!” en el display de *Logo!*, ya que las bombas y el quemador serán desactivados por el mismo *S7-200* y no se ha considerado importante el estado de las electroválvulas.

S7-200

La programación del *S7-200* es muy sencilla y se limita a la activación de las bombas, del quemador y de la alarma. No hay ninguna temporización, simplemente se utilizan las señales procedentes del bus *AS-i*.

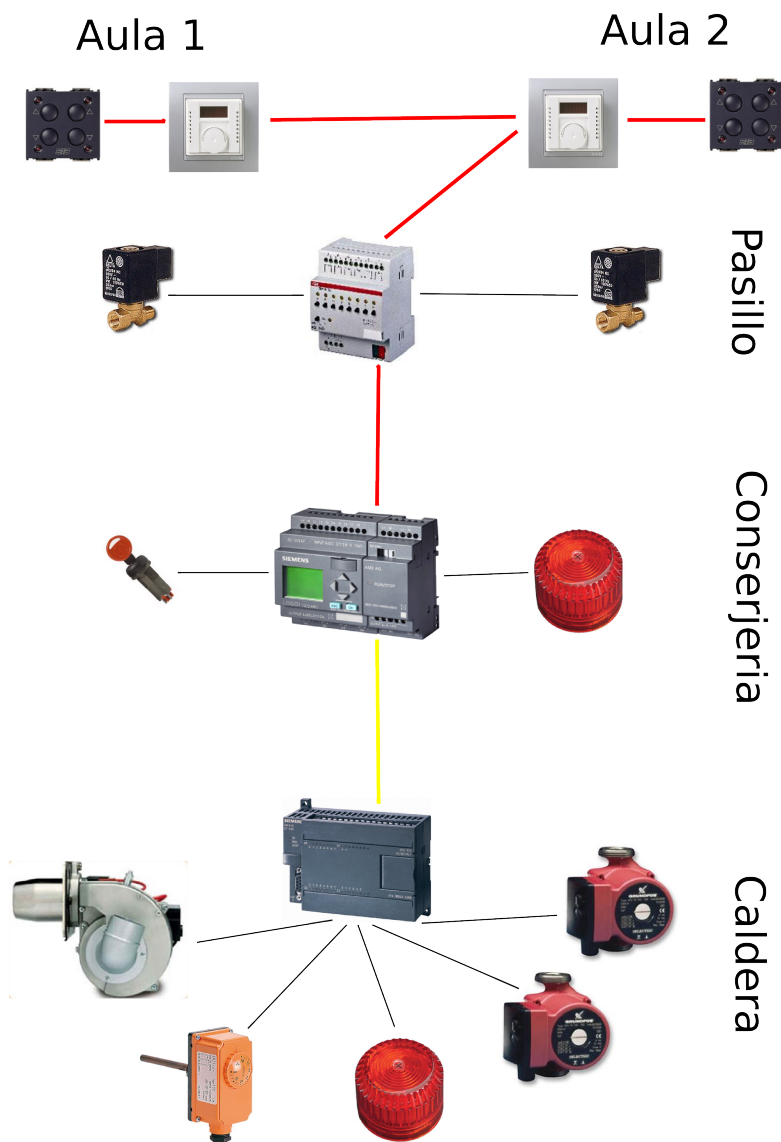


Figura 7.1: Esquema disposición aparatos

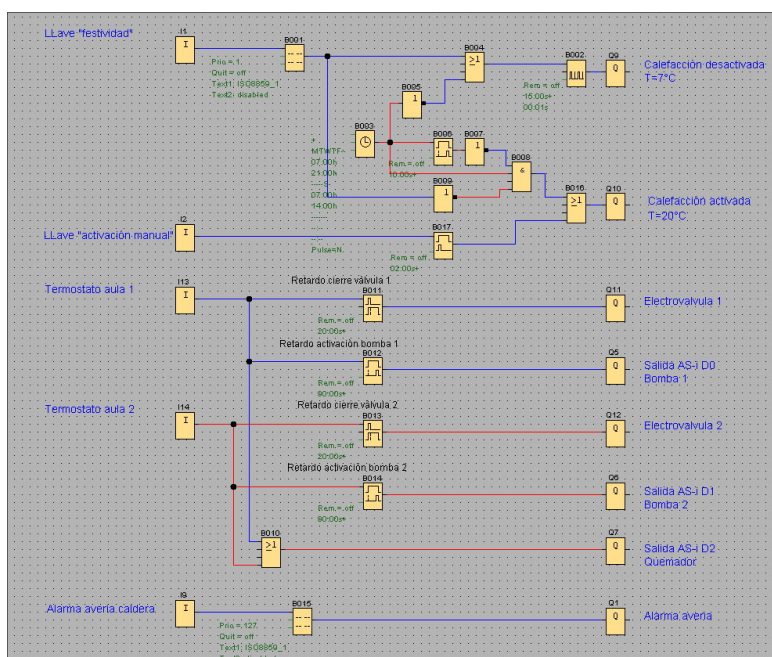


Figura 7.2: Esquema lógico en *Logo!*

Como se puede ver en la figura 7.3, la primera instrucción es la activación del quemador, que depende de la salida Q7 de *Logo!* en conjunto con el termostato de la caldera y la señalización de averías. Luego se controlan las bombas, activando la respectiva, según las salidas Q5 o Q6 de *Logo!*, las cuales corresponden a la dos aulas, y siempre que no se hayan detectados averías. De último se activa la bombilla de señalización (Q0.3) en el caso que se detecte una avería y se envía la alarma a *Logo!* por el bus AS-i.

KNX

En *KNX*, siguiendo las configuraciones de grupo visibles en la figura 7.4, empezaremos por las funciones de selección de temperaturas de confort por medio de los pulsadores en los termostatos. Ya que los termostatos no son iguales, y por eso tienen distintas modalidades de funcionamiento, en la aula 01 se seleccionarán entre 3 temperaturas de confort y la modalidad contra heladas, mientras que en la aula 02 el pulsador seleccionará entre temperatura de confort y la modalidad contra heladas. El segundo grupo intermedio está dedicado a la configuración de los termostatos por medio de *Logo!*, como previsto en los distintos horarios. El tercer apartado contiene la configuración de la salida binaria, que controla las electroválvulas, según el estado de las salidas Q11 y Q12 de *Logo!*. En fin el grupo “Termostatos_Logo” permite

Cuadro 7.1: Entradas/Salidas en *Logo!*

<i>Logo!</i> device	Tipo de entrada/salida
Entradas	
<i>I1–I8</i>	Entradas físicas
<i>I9–I12</i>	Entradas procedentes del bus <i>AS-i</i>
<i>I13–I24</i>	Entradas procedentes del bus <i>KNX</i>
Salidas	
<i>Q1–Q4</i>	Salidas físicas
<i>Q5–Q8</i>	Salidas hacia el bus <i>AS-i</i>
<i>Q9–Q16</i>	Salidas hacia el bus <i>KNX</i>

enviar a *Logo!* la variable de salida de los termostatos, para que pueda elaborar la activación o desactivación de los distintos aparatos (válvulas, bombas, quemador).

7.1.4. Mejoras

Dada la naturaleza experimental de la instalación, es interesante la introducción de mejoras en las funcionalidades, la flexibilidad del sistema y la seguridad del mismo. A continuación se exponen ejemplos de funcionalidades añadibles:

- Se distingue unas vacaciones largas (por ejemplo la Navidad, en las que la calefacción se queda apagada por una larga temporada) de las de un día. Así se evita que el día siguiente se pueda poner en marcha automáticamente la calefacción, hasta que llegue un empleado en conserjería a reactivar el sistema. Se usarán, entonces, una entrada con pulsador para deshabilitar el sistema por un solo día y una entrada con interruptor para deshabilitarlo por un tiempo indeterminado.
- El lunes por la mañana la calefacción se pone en marcha una hora antes, ya que se ha quedado apagada todo el fin de semana. Lo mismo ocurre el día después de un día de vacaciones.
- La configuración de la temperatura de confort a 20°C se verifica por la mañana y también a las 15 horas, ya que los estudiantes de la mañana probablemente no son los mismos que los de la tarde y se tiene que asegurar una temperatura confortable para ellos también.

```

// local outputs

//      Q0.0 quemador
//      Q0.1 bomba aula 01
//      Q0.2 bomba aula 02

LD      M0.3          // quemador
A       IO.0          // if m0.3 = logo Q7 = 1
AN      IO.1          // and termostato quemador
=       Q0.0          // y no hay averias
                          // q0.0 = 1 - activa quemador

LD      M0.1          // bomba aula 01
AN      IO.1          // if m0.1 = logo Q5 = 1
=       Q0.1          // y no hay averias
                          // activo la bomba de la aula 01

LD      M0.2          // bomba aula 02
AN      IO.1          // if m0.2 = logo Q6 = 1
=       Q0.2          // y no hay averias
                          // activo la bomba de la aula 02

// local inptus

//      I0.0 termostato quemador
//      I0.1 alarma averia

LD      I0.1          // s7 input on logo output
=       M1.1          // if i0.1 = 1 - si hay una averia
                          // set q1 (Logo_I9) on AS-i device = 1

```

Figura 7.3: Programa en *S7-200*

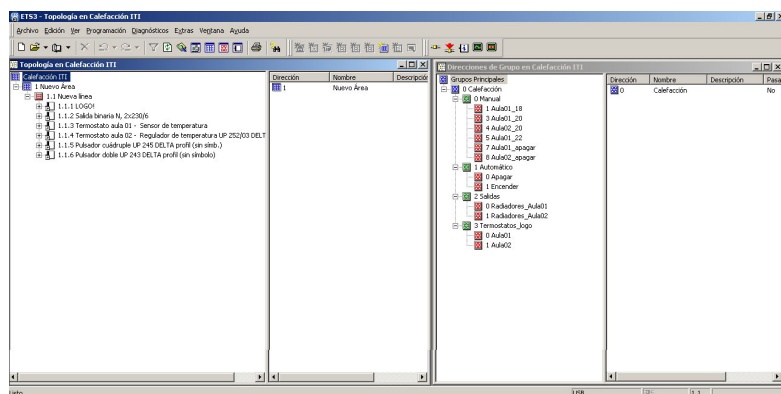


Figura 7.4: Configuración de grupo en *KNX*

Capítulo 8

Conclusiones

Se ha conseguido el objetivo propuesto. Es decir que es posible poner en comunicación aparatos del sistema *KNX* con un *S7-200* por medio de un módulo lógico *Logo!*.

Se ha dado origen así a un sistema muy flexible, que integra las características mas agradables de los aparatos *KNX* con la fiabilidad de un *API*.

Es oportuno subrayar la única falta que se ha encontrado, que es la limitada capacidad de transferencia, que es de solo 4 bits en un sentido y 4 en el otro. Todavía, se puede pasar por encima de esta limitación agregando más módulos *Logo!*, dado sus bajo coste. Y además, se tiene que tener en cuenta que ha sido posible enviar los parámetros fundamentales como *marcha*, *paro* y *alarma*, que son los más utilizados en este tipo de instalación.

Se quiere destacar también que el proyecto se ha concretizado en la realización de un panel de laboratorio donde se puede seguir investigando otras formas de comunicación y mejoras a esta solución, además de efectuar las practicas de la asignatura de “Infraestructuras Inteligentes”.

Conclusioni

L'obiettivo proposto è stato raggiunto. Ovvero si è riusciti a far comunicare apparecchi del sistema *KNX* con un PLC per mezzo di un modulo logico *Logo!*.

Si è dato vita in questo modo a un sistema molto flessibile, che integra le caratteristiche di semplicità, tipiche di *KNX*, con l'affidabilità di un *PLC*.

Si ritiene opportuno evidenziare l'unica pecca che si è incontrata, che consiste nella ridotta capacità di dati trasferibili, che è limitata a 4 bits in un verso e 4 in quello opposto. Tuttavia questo limite può essere sorpassato, utilizzando più moduli logici *Logo!* in parallelo, tenuto conto del basso costo degli stessi. Oltretutto, si vuole evidenziare come sia stato possibile inviare i parametri essenziali per la gestione di un impianto, quali *marcia*, *arresto* e *allarme*, che sono gli unici veramente fondamentali in questi casi.

Si vuole ricordare inoltre che il progetto si è concretizzato nella realizzazione di un pannello da laboratorio, sul quale si possono eseguire, oltre alle esercitazioni per gli studenti del corso di "Infraestructuras Inteligentes", nuove indagini al fine di migliorare questa soluzione o di conseguire una migliore.

Capítulo 9

Bibliografía

Apuntes de las clases de “Automatización Industrial” y de “Infraestructuras Inteligentes” de la Escuela de Ingeniería Industrial de la Universidad de Extremadura.

Manuales técnicos procedentes de:

<http://www.abb.com>

<http://www.schneider-electric.com>

<http://www.siemens.com>