

# ARQUITECTURA HARDWARE PARA EL CONTROL REMOTO BASADO EN UN MICROCONTROLADOR CON EL BUS DE CAMPO INTERBUS

I.J. OLEAGORDIA<sup>1</sup>, R. URRETABIZKAYA<sup>2</sup>, J.I. SAN MARTIN<sup>3</sup>

<sup>1</sup> Departamento de Tecnología Electrónica. E.U.I.T.I. de Bilbao Universidad del País Vasco UPV/EHU. España.

<sup>2</sup> Departamento de Tecnología Electrónica. E.U.I.T.I. de Bilbao Universidad del País Vasco UPV/EHU. España.

<sup>3</sup> Departamento de Ingeniería Eléctrica. E.U.I.T.I. de Eibar Universidad del País Vasco UPV/EHU. España.

*El desarrollo en la aplicación del control distribuido que está formado por una gran variedad de topologías y campos de aplicación evoluciona de forma concurrente al de las comunicaciones entre los diversos subsistemas que dan operatividad al sistema. Esta variedad implica distintos niveles de abstracción sobre la integración conforme al concepto de "Computer Integrated Manufacturing "CIM". En este contexto cada día es más necesario la implementación de tarjetas y dispositivos inteligentes para realizar la monitorización y control remoto. Un bus de campo transfiere información secuencial, en serie, por un número determinado de líneas. En este trabajo se describe y analiza la realización de un sistema maestro-esclavo con el bus de campo Interbús y una tarjeta basada en el microcontrolador P89C51RD2.*

## 1. Introducción

El objetivo de este trabajo es la realización de un sistema de control maestro-esclavo basado en un microcontrolador. Dicho sistema cumple una serie de especificaciones que se enumeran a continuación:

- ❑ Control remoto y local por parte de un operador de una aplicación que, en este caso, y por motivos académicos consiste en una maqueta de motores paso a paso previamente implementada.
- ❑ Comunicación entre los diferentes elementos mediante un Bus de Campo (Interbus)
- ❑ Monitorización Remota del Sistema mediante SCADA (Supervisory Control And Data Acquisition). En la figura 1 se muestra un diagrama en bloques del sistema.

### 1.1. Bus de campo.

Un bus de campo es un sistema de transmisión de datos que simplifica la instalación y operación de máquinas y equipamientos industriales utilizados en procesos de producción. El objetivo de un bus de campo es sustituir las conexiones punto a punto entre los elementos de campo y el equipo de control a través del habitual bucle de corriente de 4-20mA. Normalmente son redes digitales bidireccionales multipunto, montadas sobre un bus serie, que conectan dispositivos de campo como PLCs, transductores, actuadores y sensores. Cada dispositivo de campo incorpora cierta capacidad de proceso, que lo convierte en un dispositivo inteligente, manteniendo siempre un costo bajo. Cada uno de estos elementos es capaz de ejecutar funciones simples de diagnóstico, control o mantenimiento, así como de comunicarse bidireccionalmente a través del bus. El bus de campo utilizado, Interbus de Phoenix Contact, se basa en una estructura Maestro-Esclavo para la interconexión de los distintos elementos.

## 1.2. Sistema SCADA.

El sistema SCADA monitoriza y controla la aplicación de forma remota desde un ordenador personal (PC). Es capaz de controlar el sistema de forma automática pero a su vez permite realizar la función de interfaz de usuario entre el operador y la aplicación mediante una serie de controles intuitivos para poder intervenir en el sistema de forma remota en caso de ser necesario.

Se utiliza el programa GraphWorx32 de Iconics basado en OPC para el desarrollo de un interfaz de usuario con el sistema. Dicho interfaz es capaz de controlar y monitorizar el sistema de motores situado en planta mediante una serie de pantallas intuitivas y fáciles de entender.

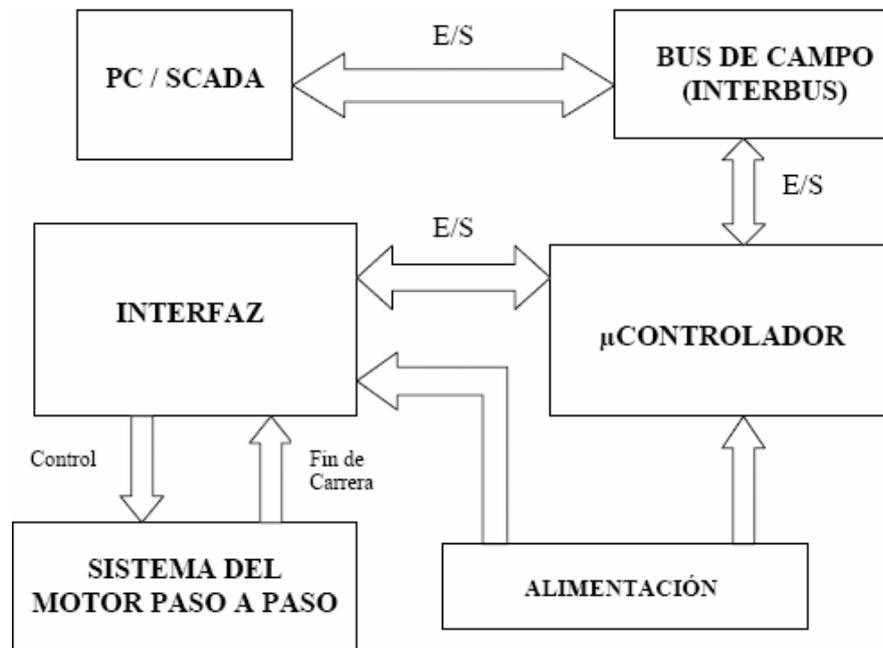


Fig. 1. Estructura general del sistema con todos sus bloques funcionales

## 1.3. Motor paso a paso e interfaz.

Un bus de campo se encarga de comunicar los diferentes sensores y actuadores de un sistema en la planta siendo el nivel más simple en la estructura de comunicaciones industriales. Como planta se emplea una maqueta de motores paso a paso, experimentada desde distintas plataformas y con una finalidad esencialmente didáctica.

## 1.4. Microcontrolador.

Como se aprecia en la figura 1 a través del microcontrolador se enlaza la funcionalidad del sistema SCADA con el proceso a monitorizar y controlar. El uso del microcontrolador posibilita la actualización constante de la aplicación sin tener que alterar elementos de hardware siendo sólo necesario la adecuación del software. El sistema está basado en el microcontrolador P89C51RD2 de la familia 8051 el cual dispone de 64kb de memoria flash y soporta programación ISP (In System Programming) e IAP (In Application Programming) de forma que en el  $\mu C$  se pueden grabar datos en la memoria flash durante el tiempo de ejecución del programa mediante el uso de las funciones adecuadas. Con ello se puede conseguir almacenar datos del proceso que quedan almacenados de forma no volátil y que no forman parte de la programación original. El  $\mu C$  realiza todas las funciones de control tanto del interfaz como del bus de mediante el empleo de sus interrupciones y bloques funcionales, empleados casi en su totalidad.

## 2. Bus de campo Interbus

La solución adoptada en este trabajo ha sido el bus de campo Interbus (IBS) por motivos de disponibilidad. Es un bus de campo estándar IEC 61158, para aplicaciones industriales y procesos de producción. Está basado en el estándar RS-485, requiere doble línea de transmisión (5 cables entre dos dispositivos), con una velocidad de transmisión de 500 Kbit/s, y alcanza una distancia máxima de 400 metros entre dos puntos. EL número máximo de dispositivos es de 512. Este bus de campo abierto es un sistema de automatización que conecta todas las E/S y dispositivos de campo habitualmente usados en sistemas de control. El cable del bus serie puede ser usado para realizar una red de sensores y actuadores, controlar maquinas y componentes de sistema, conectar diferentes zonas de producción o para conectar sistemas de más alto nivel como salas de control <www.interbusclub.com>.

### 2.1. Topología de Interbus.

En términos de topología, Interbus es un sistema en anillo (todos los dispositivos están integrados en un camino cerrado). Cada dispositivo amplifica la señal que recibe y la envía de nuevo, permitiendo altas velocidades de transmisión a mayores distancias. A diferencia de otros sistemas en anillo, las líneas de ida y retorno de los datos en Interbus son enviados a todos los dispositivos mediante un solo cable. Esto hace que la apariencia física del sistema sea la de una estructura de árbol abierto. La línea principal sale del maestro del bus y puede ser usada para formar múltiples redes de hasta 16 niveles. Así el bus puede ser rápidamente adaptado si las aplicaciones cambian. El sistema maestro/esclavo de Interbus permite la conexión de 512 dispositivos en 16 niveles distintos de redes. El anillo es automáticamente cerrado por el último dispositivo.

### 2.2 Elementos básicos de Interbus.

Los elementos básicos de un sistema con Interbus son:

*I) Tarjeta Controladora.* La tarjeta controladora es el maestro que controla el tráfico de datos. Transfiere datos de salida a los módulos correspondientes, recibe datos de entrada y monitoriza la transferencia de datos. A su vez son mostrados los mensajes de diagnóstico y los mensajes de error son transmitidos al sistema principal.

*II) Bus Remoto.* La tarjeta controladora es conectada a los dispositivos mediante el bus remoto. La rama de esta conexión está considerada como una rama de bus remoto. Los datos pueden ser transmitidos físicamente mediante cables (estándar RS-485), fibra óptica, transmisión infrarroja u otros medios. Existen módulos especiales del bus como ciertos módulos de entradas y salidas o dispositivos como robots, etc, que también pueden ser usados como dispositivos remotos. Cada uno de ellos tiene una fuente de alimentación local y un segmento de salida aislado eléctricamente. Además de las líneas de transmisión de datos, la instalación del bus remoto puede también disponer de una fuente de alimentación para los módulos conectados.

*III) Bus Local.* El bus local se ramifica desde el bus remoto mediante un módulo de interfaz y conecta los dispositivos de bus local. En este nivel no son permitidas ramas. Los dispositivos de bus local son normalmente módulos de E/S en una estructura distribuida.

*IV) Lazo Interbus.* Los sensores y actuadores distribuidos en máquinas o sistemas son enlazados a la red mediante el lazo Interbus.

### 2.3 Protocolo de Interbus.

El protocolo de Interbus trabaja de acuerdo con el método de suma de paquetes usando un paquete único para los mensajes de todos los dispositivos. En este método de acceso maestro/esclavo, el maestro actúa como el acoplamiento con un sistema de control de un nivel superior. El uso de este método proporciona un gran nivel de eficiencia de transmisión de datos y permite que los datos sean transmitidos y recibidos simultáneamente (Operación Full-Duplex). El paquete de transmisión se divide en cabecera, datos de los dispositivos, corrección de errores e información final, figura 2.



Fig. 2. Esquema del paquete de transmisión de Interbus

## 2.4 Diagnóstico y errores en Interbus.

Interbus, al ser un sistema en anillo con dispositivos que se acoplan activamente, permite la segmentación de todo el sistema en subsistemas eléctricamente independientes. En el caso de producirse un error de dispositivo, un cortocircuito o interrupción del cable del bus, la comunicación solo falla a partir de la localización del error. Ello permite al personal de mantenimiento la eliminación de errores de forma rápida y precisa.

## 3. Tarjeta IBS UNI DIO

IBS UNI DIO (UNIversal Digital Input/Output) es una tarjeta esclava universal de entradas y salidas digitales que está provista de un completo interfaz para comunicarse y transmitir datos dentro del bus de campo Interbus. Gracias a esto, puede concentrarse principalmente en la aplicación para la que esté trabajando sin ver disminuidas sus capacidades por motivos de comunicación con el Maestro. Gracias a las opciones de configuración posibles permite múltiples combinaciones de entradas y salidas digitales así como su interfaz con un microcontrolador, caso de este trabajo, la tarjeta IBS UNI DIO puede usarse como interfaz con Interbus en la gran mayoría de las aplicaciones, figura 3.

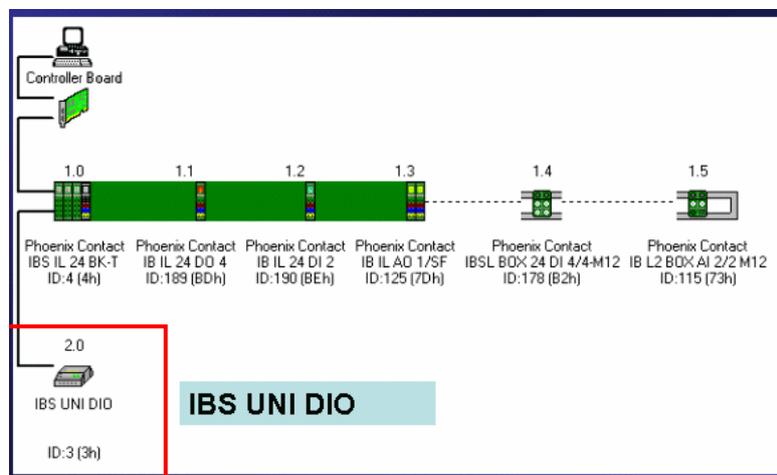


Fig. 3. Integración de la tarjeta IBS UNI DIO en el bus de campo Interbus

El módulo IBS UNI DIO posee como elemento principal un encapsulado de protocolo denominado SUPi (Serial Universal Protocol Interface). Se basa en tecnología RS-485 para conectarse a un dispositivo anterior compatible como bus remoto. El interfaz con las entradas y salidas se realiza a través de un zócalo de montaje superficial de 60 patillas dispuestas en dos filas con paso de 1.54 mm entre ellas. Este zócalo es del tipo SFM 130 FDA de la compañía de SAMTEC. En la figura 4 se muestra el mencionado módulo con el cable de conexión al  $\mu$ C (aplicación) e Interbus (IBS) y su integración, a nivel de hardware, en el bus de campo Interbus.

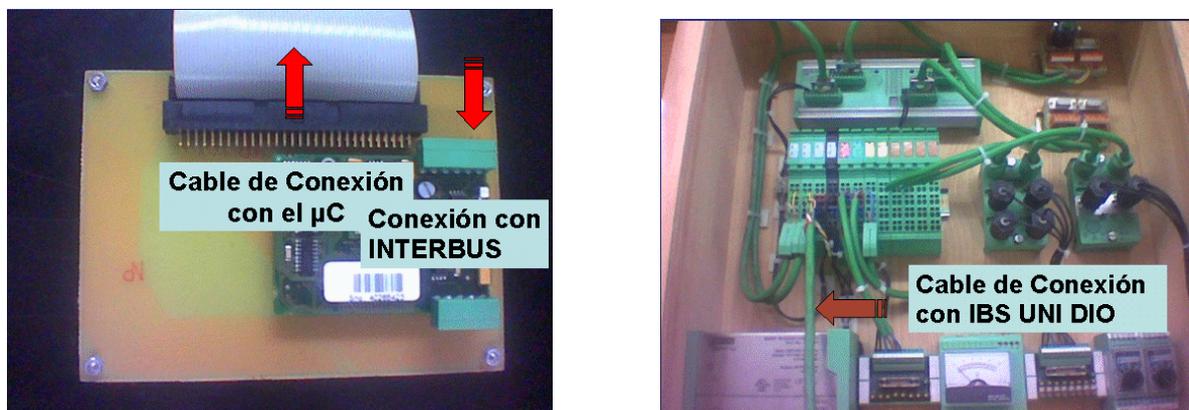


Fig. 4. Tarjeta de conexión IBS UNI DIO con las conexiones al microcontrolador y bus de campo Interbus

### 3.1 Interfaz con Interbus.

El módulo IBS UNI DIO no está diseñado para un uso independiente, sino como una tarjeta capaz de integrar numerosos elementos en la comunicación existente en Interbus, proporcionándoles el interfaz necesario, figura 3. Para la conexión de los conectores del tipo D9 se debe de utilizar cable trenzado, siendo el cable IBS RBC METER-T el estándar de Interbus.

### 3.2 Interfaz con la aplicación.

Las señales relacionadas con la aplicación se pueden agrupar en cinco partes: *Alimentación, Diagnostico, Configuración, Entradas y Salidas, y Registros de expansión externos.*

Para desarrollar el interfaz con la aplicación, una placa base debe ser ideada con la incorporación de los elementos necesarios para la propia aplicación y un conector sobre el cuál puede ser incorporado el IBS UNI DIO. Es también importante observar los requisitos de la inmunidad al ruido (IEC 801-4) y la tensión de aislamiento, por lo menos de 500 Voltios de continua. Las señales transferibles con la aplicación están accesibles mediante un zócalo de montaje superficial de 60 patillas con paso de 1,54 mm entre pines. Entre estos pines se encuentran las 32 entradas y salidas de datos, que se pueden configurar libremente en grupos de 8. En principio, todos las patillas se pueden utilizar como entradas o salidas.

El zócalo del IBS UNI DIO proporciona las señales de interfaz con los pines multifunción MFP del encapsulado IBS SUPI mediante el cual se efectúa la interfaz con la aplicación. Si la configuración es apropiada, un interfaz adicional con un microprocesador puede ser accesible a través de las señales D16 a la D 31 que corresponden a los 16 pines MFP. Como se explicará posteriormente la interfaz con el  $\mu$ C se realiza a través de los puertos P0, P2 y P3.2. En este caso los registros internos del encapsulado de protocolo pueden ser accesibles.

El módulo IBS UNI DIO esta diseñado como un dispositivo para ser usado universalmente. Por eso tiene tres grupos de entradas configurables. El primer grupo determina como se representa el dispositivo en el bus. El segundo grupo determina la funcionalidad de los componentes individuales presentes en la placa del módulo y el tercer grupo define como éstas funcionalidades están unidas.

La tarjeta IBS UNI DIO está configurada para intercambiar dos registros, de tamaño byte, entre la red del bus de campo y el microcontrolador. A dichos registros se les ha efectuado una asignación lógica de variables. En el **Registro 1** se ha codificado la velocidad del motor como aplicación específica y en el **Registro 2** se han codificado los diferentes comandos de control.

### 3.3 Circuito integrado IBS SUPI.

IBS SUPI (Serial Universal Protocol Interface) es un circuito integrado encargado de suministrar el protocolo de Interbus para numerosas aplicaciones. En el caso de este trabajo se hace referencia al SUPI 3 debido a que es la tercera generación de componentes esclavos de Interbus. Este encapsulado es físicamente y funcionalmente compatible con la anterior generación SUPI 2. La cualidad más importante a destacar en la tercera generación es el controlador de diagnostico y generador de informes gracias a un nuevo concepto de diagnostico en Interbus. El encapsulado de protocolo SUPI 3 puede conectarse a Interbus formando un bus local o un remoto. La transmisión de datos se realiza a través de fibra óptica o por medio del protocolo RS-485.

Este dispositivo comprende la capa 1 y la capa 2 del modelo de referencia OSI/ISO. El dato se proporciona al MDS, (Medium Dependent Sublayer) o subcapa dependiente del medio, de un MAU, (Medium Attachment Unit) o unidad ligada al medio externo, ejemplo: RS-485 ó acceso a fibra-óptica. En la capa 1 se lleva a cabo la reexaminación, codificación y decodificación de la línea así como la definición de las condiciones de tiempo. El SUPI 3 tiene tres canales en el MDS siendo un interfaz de entrada y dos de salida. La capa 2 está representada por el control de acceso al medio o MAC (Medium Access Control). Esta capa realiza el acceso al anillo y se encarga de la seguridad de los datos. El MAC se encarga de suministrar los registros de transmisión y recepción a Interbus, así como de enviar y recibir el registro de 16 bites en los ciclos de identificación y transmisión.

La aplicación y las capas de protocolo más altas tienen acceso a estos registros a través de los 16 bites multifunción del interfaz MFP. El interfaz MFP puede configurarse según los requisitos de la aplicación a través de cuatro pines configurables, bien como puerto de entradas y salidas o como un interfaz para microcontrolador como si de una CPU se tratase. El interfaz MFP contiene un controlador de interrupciones con los correspondientes registros de escritura y de lectura, así como los registros de parametrización y estado para aplicaciones de la CPU. Esto permite configurar el circuito y visualizar ciertos eventos del protocolo.

*I) Acceso con  $\mu P$ .* Como se ha mencionado mediante la configuración del interfaz MFP se determina la manera en que la aplicación, en nuestro caso un  $\mu C$  accede a Interbus a través del SUPI 3. Se distinguen tres clases: Cabecera de bus, Entradas y salidas directas, y Acceso utilizando un  $\mu C$ . A su vez dentro de cada una de las clases mediante la configuración de los pines C3, C2, C1 y C0, del conector de 60 pines, el interfaz MFP adquiere una determinada configuración, que para el caso concreto de interfaz con un  $\mu C$  se indica en la tabla adjunta

C3	C2	C1	C0	Modo del interfaz MFP
1	0	0	0	Cabecera de bus local de 8 hilos
0	0	1	1	Cabecera de bus local de 8 hilos y I/O
0	0	0	0	Cabecera de 2 hilos
1	0	0	1	16 bits de salida
1	0	1	0	16 bits de entrada
1	1	0	1	8 bits de entrada y 8 bits de salida
0	0	0	1	1 bytes de interfaz con $\mu C$
1	0	1	1	2 bytes de interfaz con $\mu C$
0	0	1	0	8 bytes de interfaz con $\mu C$

En el presente trabajo, como se ha indicado anteriormente, se ha realizado la interfaz para 2 bytes. En este modo es posible direccionar el dispositivo SUPI 3 como un componente de entradas y salidas como si de una memoria RAM se tratase. Para este propósito el encapsulado de protocolo consta de una anchura de datos de 8 bites, bus de datos bidireccional, un bus de direcciones de 4 bites, señales de control activa a nivel bajo de habilitación de encapsulado activa a nivel bajo (/CS), lectura (/RD), y escritura (/WR), así como una línea de interrupción (/IRQ) también activa a nivel bajo. En la tabla adjunta se detalla la función específica de cada uno de los pines del puerto MFP.

MFP(n)	Asignación	MFP(n)	Asignación	MFP(n)	Asignación	MFP(n)	Asignación
0	A0	4	/RD	8	D0	12	D4
1	A1	5	/WR	9	D1	13	D5
2	A2	6	/CS	10	D2	14	D6
3	A3	7	/IRQ	11	D3	15	D7

Los registros de datos de la SUPI mantienen el intercambio de datos de entrada y salida entre la aplicación y el Maestro de Interbus. Mientras que el  $\mu C$  lee y escribe en los registros de datos asincrónicamente al ciclo de Interbus se pueden producir incoherencias en los datos si estos coinciden con la fase de actualización del registro en un ciclo de Interbus. Para sincronizar el acceso a los registros de datos, el SUPI posee una interrupción lógica. Las diferentes fuentes que puedan generar una interrupción se habilitan a través de la habilitación de ciertos bits situados en dos registros de habilitación de interrupciones. Al suceder una interrupción la señal /IRQ se pone acero y en los registros Interrupt Event I e Interrupt Event II se puede determinar el evento que la ocasiona, figura 5.

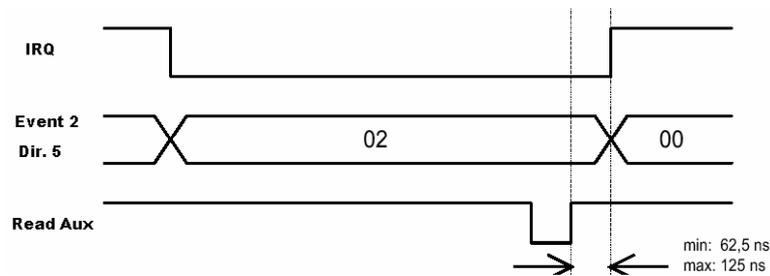


Fig. 5. Cronograma de la señal de interrupción en un ejemplo de ciclo de datos

#### 4. Microcontrolador

La tarjeta del microcontrolador, tiene la función de gestionar las señales adecuadas para el control del sistema de los motores paso a paso, mantener una comunicación de datos con el dispositivo IBS\_UNI\_DIO y ser controlable también por el operario a través de los medios disponibles en la propia tarjeta. A modo de resumen y desde la perspectiva funcional la operatividad de esta tarjeta se resume en: *programar e inicializa la tarjeta IBS UNI DIO, mantener actualizados los registros de I/O, procesar las órdenes recibidas y ejecutarlas posteriormente en la placa de los motores paso a paso (planta del sistema).*

##### 4.1 Interfase de usuario.

El usuario de planta tiene acceso a la configuración y manipulación de los parámetros del motor mediante el módulo LCD-WM-C1602 de 2 columnas y 16 caracteres que incorpora la tarjeta y cuatro pulsadores de: *Confirmación, Escape, Avance y Desplazar* conectados a los pines P3.4, P3.5, P3.6 y P3.7 respectivamente. A través del pulsador de *Confirmación* se avanza a la siguiente opción del menú, con la de *Escape* se retoma la anterior opción de menú, la tecla de *Avance* permite avanzar por las listas de opciones y en la parametrización de valores, para esto último se utiliza también la tecla de *Avance* ya que proporciona una selección de los diferentes dígitos. El envío de datos por parte del microcontrolador a la pantalla se realiza mediante un bus de datos de cuatro bits utilizando solamente la parte alta de las patillas del byte de datos. Esto requiere una descomposición en nibbles para enviar un dato en dos partes. La LCD utiliza los pines P1.2 conectado a RS\_, P1.3 conectado a E, y P1.4, P1.5, P1.6, P1.7 conectados a DB4, DB5, DB6 y DB7 respectivamente de la LCD.

##### 4.1. Conexión con los motores paso a paso.

La tarjeta del  $\mu C$  se conecta a los motores paso a paso a través de estas cinco señales: Pulsos, Giro, Habilitación de placa motores y Sensor que corresponden a los pines P1.0, P1.1, P0.7 y P3.3(/INT1).

Las tres primeras señales son salidas de la tarjeta. La denominada Pulsos (P1.0) proporciona una onda cuadrada generada internamente por el  $\mu C$  para el avance de los motores paso a paso. La señal giro (P1.1) define el sentido de giro de los motores, cuando esta patilla este a nivel alto el motor girará en sentido horario, y a nivel bajo, es decir 0, lo hará en sentido antihorario. La otra salida (P0.7) permiten a su vez tener el control sobre la habilitación de la tarjeta controladora del sistema motor. Estas salidas han sido acondicionadas a través del CI 74HCT373. La señal denominada sensor (P3.3) es una entrada a la tarjeta proveniente del sistema motor que hace referencia a la operatividad de uno de los motores que actúa sobre un brazo mecánico.

##### 4.2. Comunicación con Interbus.

La tarjeta posee un conector para cable plano de 60 patillas que la conecta al dispositivo IBS UNI DIO, figura 4. Puesto que la configuración del modo de funcionamiento del dispositivo esclavo se realiza a través de este conector se ha optado por conectar los 16 pines multifunción MFP al  $\mu C$  y dejar el resto de señales accesibles mediante jumpers. Esto permite una reconfiguración en todo momento del dispositivo esclavo. Las conexiones son [MFP0; MFP6]  $\rightarrow$  [P0.0; P0.6]; MFP7  $\rightarrow$  P3.2, y [MFP8; MFP15]  $\rightarrow$  [P2.0; P2.7]

Las señales de registros de expansión se han dejado accesibles para la interconexión de unos con otros en función de las necesidades o bien se pueden conectar a masa puesto que en ciertos modos de funcionamiento se requiere. Las señales de configuración poseen la opción de conectarse fácilmente a nivel alto o a nivel bajo en función del modo de funcionamiento del dispositivo esclavo que se desee adoptar. Los datos que van desde D0 hasta D15, en el cable plano, también están accesibles a través de los jumpers o bien se pueden conectar a masa en caso de que se configuren como entradas y alguna de ellas no se utilice para evitar problemas.

#### 4.3. Programación.

El  $\mu\text{C}$  en el que se basa esta tarjeta controladora es reprogramable sin necesidad de quitarlo del zócalo. Se programa a través del puerto serie del PC por las pines TxD y RxD, esto obliga a situar en la tarjeta un circuito integrado adaptador de tensiones (MAX232) entre el microcontrolador y el puerto serie del ordenador, y por lo tanto también un conector DB9. Se ha empleado un conmutador para fijar el modo de funcionamiento bien sea en programación o ejecución del programa.

Las funciones asociadas al manejo de la memoria FLASH son las encargadas de realizar la programación IAP (In Application Programming). Mediante el uso de las mismas el programa puede acceder a la memoria Flash del  $\mu\text{C}$  y almacenar datos de forma permanente o leerlos. Estas funciones se utilizan en todas las partes del proceso que implican la alteración de una variable de datos para evitar su pérdida en caso de fallo de alimentación del sistema. Las funciones han sido implementadas con el soporte de funciones extraídas de una librería `<rx2iaplib.h>` suministrada por el fabricante del  $\mu\text{C}$  en lenguaje ensamblador [www.esacademy.com](http://www.esacademy.com). A modo de ejemplo, funciones asociadas a la interfase con Interbus son :

- ❑ void **init\_ibs** (void): Configura los registros de la tarjeta de Interbus (IBS UNI DIO) para que funcione en modo microcontrolador dejándola preparada para la comunicación con el sistema.
- ❑ void **lectura\_ibs**(unsigned int dir): Lee la dirección de registro de datos de la tarjeta de Interbus especificada. Los datos que maneja son “veloc” y “modo” de tamaño de byte cada uno de ellos.
- ❑ void **escritura\_ibs**(unsigned char dato,unsigned int dir): Escribe en la dirección del registro de datos de la tarjeta de Interbus especificada. Los datos que maneja son “veloc” y “modo” de tamaño de byte cada uno de ellos.

### 5. Sistema SCADA

El sistema SCADA se ha realizado utilizando el programa GRAPHWORX de Iconics. Las variables se obtienen accediendo a un OPC Server que esta en comunicación con el bus de campo Interbus (IBS). El GRAPHWORX recibe las variables del bus de campo mediante un Servidor OPC, el cual está constantemente actualizado con los datos que le suministra el bus de campo y actúa de enlace entre ambos sistemas. El programa de SCADA puede tanto leer como escribir variables en el servidor, de esta manera se puede realizar un control integral del mismo.

#### 5.1. Interface de usuario.

El sistema SCADA monitoriza el sistema de los motores paso a paso situado en la planta. Todas las acciones realizadas por el operario de planta en el sistema pueden ser visualizadas en el PC.

Así mismo en caso de ser necesario, puede tomarse el control del sistema de forma remota y controlar todas las acciones mediante el sistema SCADA. El usuario tiene control total de todas las posibilidades del sistema, además también se visualizan en pantalla todas las acciones que ha realizado, aunque no tenga visión directa del equipo del motor, asegurándose así de que sus acciones han tenido el efecto deseado. En la figura 6 se muestra una imagen intuitiva de la pantalla donde se monitoriza el sistema.



Fig.6 Pantalla gráfica de monitorización en el SCADA

Los parámetros visualizados son los siguientes:

- ❑ *Modo de Funcionamiento*
  - Modo RPM
  - Modo Vueltas
  - Modo Pulsos
- ❑ *Velocidad del Motor en caso de estar en modo RPM.*
- ❑ *Número de Vueltas en caso de estar en modo Vueltas.*
- ❑ *Número de Pulsos en caso de estar en modo Pulsos.*
- ❑ *Sentido de Giro*
  - Sentido Horario.
  - Sentido Antihorario.
  - Alternancia.

- ❑ Animación en Indicador de sistema en Funcionamiento.
- ❑ Indicador de Estado del Sistema (Habilitado o Deshabilitado)
- ❑ Indicador de Error en el BUS
- ❑ Opción de cambio a modo Parametrización
- ❑ Opción de visualización del estado de todos los dispositivos del bus para la localización del error.

Cuando el usuario entra en el modo parametrización, el sistema pasa a estar gobernado por el usuario remoto. Además de continuar monitorizando el sistema, el operador del SCADA puede realizar las siguientes acciones en el sistema de motores pasa a paso localizado en planta:

- ❑ Cambiar el modo de funcionamiento.
- ❑ Cambiar el parámetro de velocidad, vueltas o pulsos (dependiendo del modo).
- ❑ Cambio del sentido de giro.
- ❑ Habilitar o deshabilitar el sistema.
- ❑ Posicionar el sistema en el Inicio (Posicionamiento en el sensor).
- ❑ Puede devolver el control al operario volviendo a modo de Monitorización.

En la figura 7 se indica la correspondiente pantalla gráfica



Fig.7. Pantalla gráfica de parametrización en el SCADA

En caso de producirse algún fallo en el funcionamiento del sistema, el mismo queda registrado en la pantalla de forma que es posible localizar la posición exacta del dispositivo que esta fallando, figura 8.

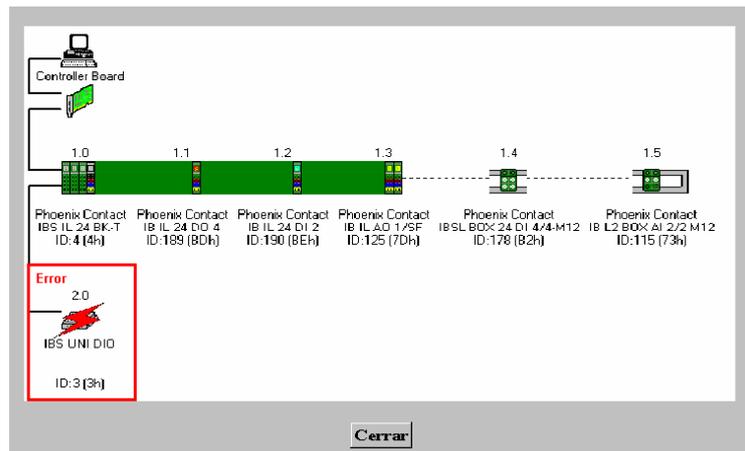


Fig.8. Pantalla gráfica del estado del bus en el SCADA

La tarjeta maestra utilizada está basada en PC. Para el control de la misma y de la red Interbus se ha utilizado el programa PC WORKS suministrado por el fabricante Phoenix Contact. El programa PC WORKS permite gestionar los diferentes dispositivos, definiendo puerto físicos y asignarlos a variables para su posterior uso en la programación de la red. Así mismo muestra el estado de los diferentes dispositivos y su ubicación lógica en el sistema. La programación de la red se ha dividido en 2 subprogramas. El primero se encarga de gestionar el intercambio de variables con la tarjeta del microcontrolador para su posterior uso en el sistema SCADA. El segundo programa está orientado a la detección de errores en el bus y su visualización, también, en el SCADA

## 6. Conclusiones

En esta comunicación se ha presentado un sistema de control y monitorización de ámbito industrial aplicado a la docencia universitaria, donde se ha querido poner de relieve la versatilidad del empleo de los microcontroladores y su operatividad conjunta con una amplia gama de subsistemas y dispositivos. Como aspectos más relevantes cabe citar:

- ❑ Es posible monitorizar y controlar un conjunto de motores paso a paso tanto local como remotamente.
- ❑ Se ha diseñado e implementado un SCADA para el control remoto.
- ❑ Se ha comunicado el SCADA con el conjunto de motores paso a paso mediante un bus de campo Interbus.
- ❑ Se ha diseñado e implementado un sistema económico y fiable.
- ❑ En el desarrollo del sistema se le ha dotado de flexibilidad y capacidad de ampliación. El entorno es fácil de manejar por el operario.
- ❑ Se han integrado los diferentes módulos del sistema para formar un conjunto funcional.
- ❑ Se ha realizado una puesta a punto final.
- ❑ De las pruebas finales y del funcionamiento global del sistema se constata que el sistema implementado cumple los objetivos deseados y por lo tanto la operatividad del trabajo. Este trabajo se ha desarrollado en el marco del proyecto de investigación de la Comunidad Europea dentro del Programa ALFA N° CRIS 96302, contrato N° AML/B7-311/97/0666/II-0354-A

## Referencias

- [1] M. Barrón, J. Martínez, *Aplicaciones prácticas con el  $\mu$ C-8051. Programación en lenguaje C*. ISBN: 84-699-1512-6. Ed. Artes gráficas Michelena, 1998.
- [2] S. Yeralan, P.E., Ph.D, H Emery, *Programming Anad Interfacing the 8052 Microntroller in C an Assembly* ISBN 0-9633257-1-X. Ed. Rigel Press, 2000.
- [3] T. W. Schultz, *C and the 8051. Hardware, Modular Programming and Multitasking. Volume I*. ISBN 0-13-754839-7 Ed. Prentice Hall PTR 1998.
- [4] T. W. Schultz, *C and the 8051. Building Efficient Applications. Volume II*. ISBN 0-13-521121-2 Ed. Prentice Hall PTR 1999.