

XIV JORNADAS DE AUTOMÁTICA  
CARTAGENA, 1993

**OCRE: OPERACIÓN Y CONFIGURACIÓN REMOTA DE CONTROLADORES**

S. Martín, J. García, F. Morilla

Dpto de Informática y Automática, UNED

Avda Senda del Rey s/n, 28040 Madrid

Tfn: 91-3987156, Fax: 91-3986697, E-mail: Fernando.Morilla@human.uned.es

**Resumen:** OCRE, Operación y Configuración Remota de Controladores, está pensado para programar y supervisar controladores adaptativos desde un IBM PC o compatible. En esta primera versión se puede comunicar con controladores tipo Electromax V Plus de Leeds & Northrup y puede controlar una red de hasta 254 controladores con la posibilidad de guardar información referente a cada controlador en ficheros ASCII.

OCRE ha sido presentado por S. Martín como proyecto fin de carrera en Mayo de 1993 en la ETSI de Telecomunicaciones de Madrid, y ha sido tutorizado por J. García (Ingeniero de Telecomunicaciones, que actualmente trabaja en el TID y es alumno de tercer ciclo del Dpto.). El proyecto surgió como una idea de F. Morilla y de J. García con el fin de facilitar la labor investigadora y docente, dentro de la línea del control adaptativo, en el Departamento de Informática y Automática de la UNED.

## 1. INTRODUCCION.

La comunicación de datos entre dispositivos inteligentes ha supuesto una revolución en todos los campos de la actividad humana. Los entornos industriales son un claro ejemplo. Dentro de una planta industrial, existen infinidad de aparatos con capacidad de procesamiento, como pueden ser sensores, reguladores, aparatos de medida, estaciones de trabajo, grandes ordenadores, etc. La posibilidad de intercambiar información entre los distintos equipos, ya sean órdenes, medidas, resultados o alarmas, por citar algunos ejemplos, hace surgir la necesidad de instalar un medio de transmisión que permita interconectar todos ellos.

Para grandes plantas donde se llevan a cabo una gran cantidad de procesos la solución óptima, desde el punto de vista técnico, es una red de área local (LAN). La ventaja de una LAN es que ofrece una elevada capacidad de transmisión (del orden de Mbits/s) y facilita la comunicación entre equipos de distintos fabricantes. Existen estándares en el mercado de redes industriales, como la red Token-Bus (IEEE 802.4), pero dado el alto

coste de este tipo de redes sólo son rentables cuando el volumen de datos a transmitir es elevado y el tiempo de reacción es un parámetro crítico. Por otro lado, otras redes más populares como Ethernet (IEEE 802.3) y Token-Ring no son adecuadas para la industria. Si, además, el número de estaciones no es muy elevado y la velocidad de transmisión de datos exigida es moderada, una alternativa económica y fiable es la comunicación serie. Las normas RS-422 y RS-485, pensadas especialmente para entornos industriales, resultan muy adecuadas en esta situación.

En el Departamento de Automática de la UNED se están utilizando desde hace varios años una serie de controladores industriales en trabajos de investigación sobre control adaptativo. Entre éstos podemos destacar al Electromax V Plus, fabricado por Leeds & Northrup, que dispone de una opción de comunicaciones vía RS-422.

Este tipo de controladores son en cierto modo autosuficientes, puesto que incorporan mecanismos de actuación suficientes para mantener al sistema "siempre" bajo control. No obstante, una vez configurado el controlador, es necesario realizar una labor de supervisión sobre él, consistente en vigilar que las variables del proceso evolucionen correctamente y en observar anomalías o situaciones de alarma. Esta labor viene facilitada por la visualización numérica y de estados del controlador, véase figura 1, pero se complica cuando hay varios controladores y más aún si los equipos están dispersos. Por otra parte, la visualización que remite el controlador es instantánea, no permite al usuario seguir una evolución del proceso a no ser que esté permanentemente pendiente de él.

Cuando, como es normal en un laboratorio, el principal objetivo es el estudio de distintos procesos y estrategias de control, el controlador se debe configurar frecuentemente y de forma diferente para cada caso concreto. El cambio de configuración suele traer consigo la modificación de una gran cantidad de parámetros. Esto hasta ahora se ha hecho manualmente, en el caso del EMAX mediante un teclado de membrana y la ayuda de dos visualizadores alfanuméricos, véase figura 1. Esta labor puede ser bastante tediosa si se cambia con frecuencia de proceso y de controlador.

Se imponía por tanto, la creación de un sistema que permitiera recoger la evolución del sistema de control y que facilitara, en lo posible, el manejo, configuración y seguimiento de este tipo de controladores. Todo ello repercutiría favorablemente en la actividad investigadora de Departamento y en el mantenimiento de una red de controladores.

Teniendo en cuenta que estaba disponible en el mercado a un precio bastante asequible el "hardware" adecuado para conexión en serie con un ordenador personal, se podía conseguir una red de bajo coste y ajustada a las necesidades de las plantas de pequeña escala. Esta es la solución que propone el proyecto OCRE (Operación y Configuración Remota de Controladores vía serie). Su ámbito de aplicación es, principalmente, un laboratorio de investigación y docencia o una pequeña planta industrial donde el número de equipos es pequeño y por tanto, el alto coste de los grandes sistemas de control distribuido no puede ser rentabilizado.

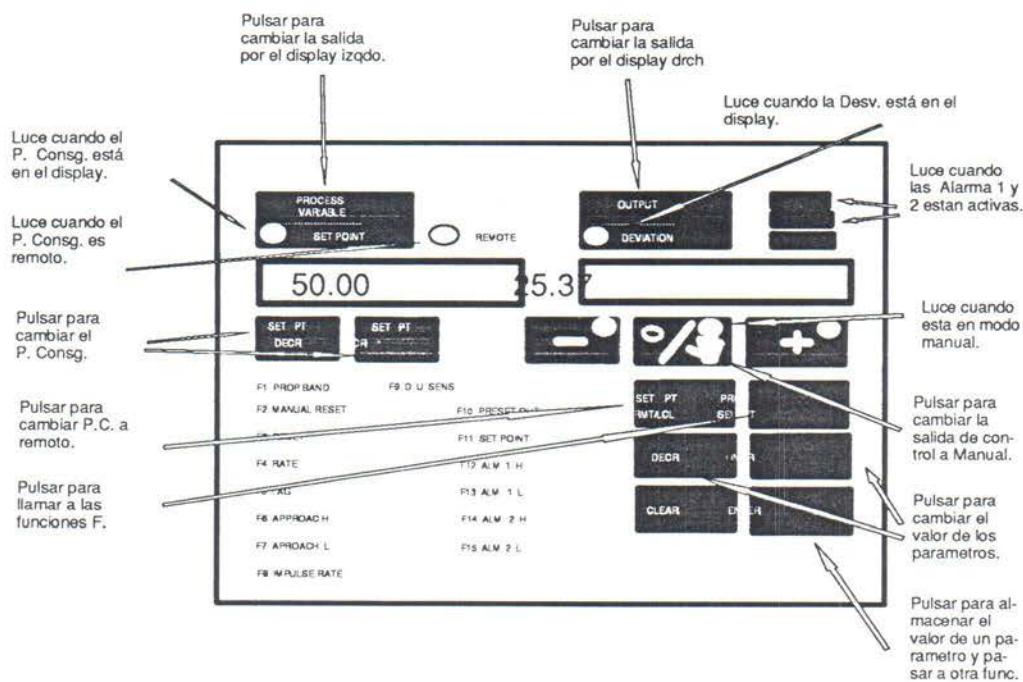


Fig. 1: Parte frontal del controlador Elecromax V Plus.

## 2. PROYECTO OCRE.

Antes de entrar en detalles, hay que puntualizar que el proyecto OCRE se ha contemplado como la primera piedra de un proyecto más grande y ambicioso dentro del cuál tendrán cabida no sólo las labores básicas de supervisión y de configuración sino todas las posibles y además otro tipo de aplicaciones, como por ejemplo, la comunicación con distintos tipos de controladores, aplicación de estrategias de control combinadas, monitorización de varios controladores a la vez, etc.

Por todo lo dicho, el proyecto se ha planteado de la forma más modular y estructurada posible para así, en el futuro aprovechar al máximo el trabajo ya realizado. Esto ha afectado directamente a las comunicaciones, que siguen el formato OSI, a los formatos

de presentación de datos al usuario, como menús, ventanas, diagramas, etc., y a las operaciones complementarias como manejo, almacenamiento e impresión de ficheros.

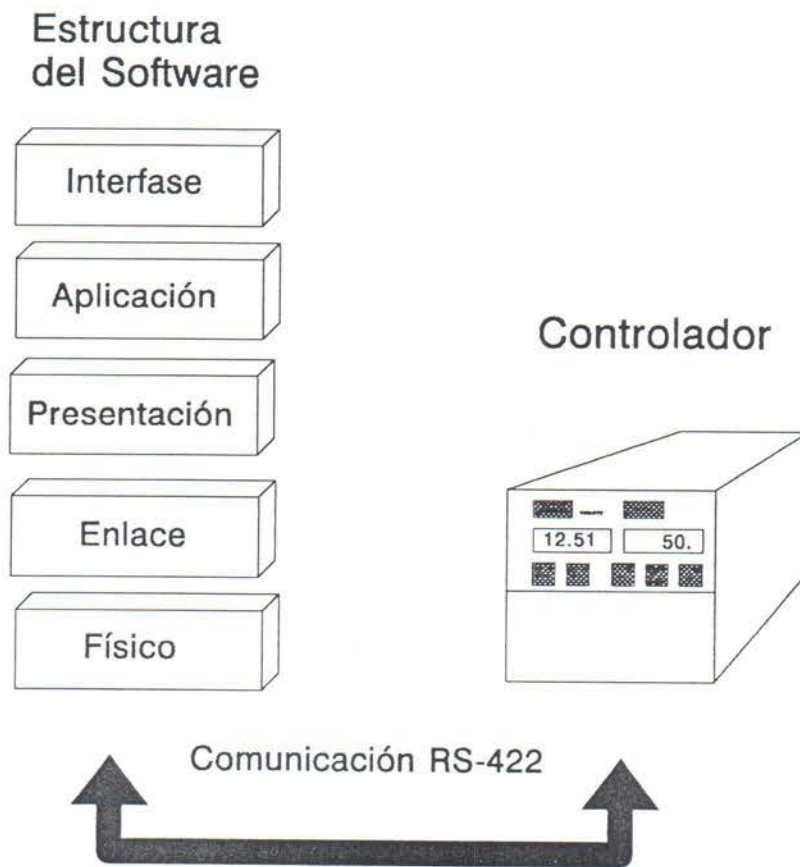


Fig. 2: Objetivos del proyecto OCRE.

Partiendo de la enumeración de los elementos de los que se iba a componer el sistema: un conjunto de controladores con la opción de comunicación RS-422, una tarjeta de comunicaciones RS-485, un ordenador personal tipo PC y un usuario ávido de poder manejar dichos controladores desde su PC. Y siguiendo una línea de diseño de abajo a arriba, los objetivos del proyecto OCRE se estructuraron como se indica en la figura 2, por tanto:

- Había que realizar un estudio del "hardware" que nos dijera hasta que punto eran compatibles los protocolos del controlador y de la tarjeta de comunicaciones y así tener asegurada la comunicación en su nivel más bajo.
- En la plataforma "software" a desarrollar se debían distinguir dos bloques funcionales, uno de comunicaciones y otro de presentación al usuario.
- El "software" de comunicaciones se estructuraría en niveles según el modelo de referencia OSI.

- Siendo uno de los principales fines del proyecto facilitar la labor del Ingeniero de control de procesos, el diseño de la interfase con el usuario habría de ser lo más amigable y funcional posible. Por ello, se implementaría a base de menús y ventanas permitiendo, también, el uso de ratón. Además, se implementarían funciones de apoyo para manejo de ficheros de configuración y funciones de ayuda dependiente del contexto.

En los próximos subapartados se describen brevemente las soluciones aportadas por OCRE, así como las características principales del software desarrollado. La memoria del proyecto incluye una información muy detallada y también un manual de usuario.

## 2.1. COMPATIBILIDAD ENTRE RS-485 y RS-422A.

Aunque los protocolos EIA RS-485 y EIA RS-422A son eléctricamente compatibles, existe una diferencia fundamental entre ellos. En el primero, el canal de recepción y transmisión conviven en el mismo cable cosa que no ocurre en el segundo.

El RS-485 está pensado para una conexión multipunto en la que puedan existir colisiones, siendo éstas resueltas por un protocolo de nivel superior. Con esta norma cada unidad conectada consta de emisor y receptor enganchados al mismo bus. El RS-422A permite solo conexión punto a punto, los canales de recepción y de transmisión que conforman el enlace están separados en cables diferentes.

Dado que el protocolo RS-422 es inferior al RS-485 y que el protocolo del nivel de enlace en el controlador, el ANSI X3.28, es punto a punto y no puede manejar colisiones, la única solución posible era adaptar la comunicación RS-485 a la RS-422. Para ello se hicieron los mínimos cambios posibles en la placa COM-485 de Metrabyte hasta conseguir una comunicación completamente compatible, sin perder la capacidad de comunicación en RS-485 y sin necesidad de incorporar un adaptador externo.

## 2.2. NIVELES DE COMUNICACION.

Una vez que estaban claros los niveles de los que se iba a componer el "software" de comunicaciones el siguiente paso fué definir y especificar lo más completamente posible las funciones que debía realizar cada uno. Además, había que ceñirse a la óptica OSI del problema y hacer los distintos niveles con la mayor independencia. Estos fueron los resultados:

**Nivel físico.** Este nivel es el que maneja directamente la placa de comunicaciones RS-485. Esta placa está regida por el microprocesador de comunicaciones 8250, un buffer de entrada y otro de salida, ambos de un sólo carácter. Excepto el número de dispositivo COM que ocupará en el mapa de memoria y el número de interrupción, que son configurables mediante "switches", todo lo demás se hace escribiendo directamente en los registros de la interfase adaptadora del 8250, accesibles como direcciones del COM1 ó COM2. En estos registros se puede elegir la velocidad de transmisión, la paridad, número de bits de parada y número de bits por carácter. Además hay una dirección disponible para lectura y escritura y otra que marca el sentido.

El nivel físico consta de las funciones que manejan directamente el chip 8250 de la tarjeta de comunicaciones, fundamentalmente para la detección de condiciones del registro de transmisión, y del manejador del puerto del dispositivo COM del DOS, en el que se efectúan lecturas y escrituras de caracteres. Concretamente incluye:

- Una función que inicializa todos los parámetros estables de la comunicación, velocidad, paridad, bits de parada y bits por carácter.
- Una función de lectura de un sólo carácter.
- Una función de escritura de un sólo carácter. Una función de lectura de tramas.
- Una función de envío de tramas.

**Nivel de enlace.** Este nivel tiene dos tareas básicas. Primero implementar todos los tipos de tramas, de lectura, de escritura y ACKs y NAKs tanto de enlace como de aplicación. Segundo, realizar la máquina de estado finito que sigue la evolución del protocolo en el tiempo. Dado que se trabajará con pocos controladores y que el modo "Repoll" implica más tiempo, sólo se utiliza el modo "Simple Poll". Incluye funciones para:

- Creación de las tramas de lectura y escritura de datos.
- Lectura o recepción de ACK y NAK del nivel de enlace.
- Escritura o envío de ACK de enlace.
- Escritura o envío de NAK de enlace.
- Lectura o recepción de ACK y NAK de aplicación.
- Lectura/Escritura ó Recepción/Envío de tramas de datos.

**Nivel de presentación.** Como los formatos de representación de la información son distintos en el ordenador y en el controlador, este nivel consta de una función para conversión de datos.

**Nivel de aplicación.** El controlador EMAX V contiene una gran cantidad de parámetros. Entre todos ellos hay algunos que rara vez son utilizados y otros que, una

vez configurados, no se volverán a cambiar. Dado que el uso de unos y otros está estrechamente relacionado con las actividades y experimentos que se desarrollan dentro del laboratorio, el primer paso fué hacer una clasificación de dichos parámetros y descartar los realmente innecesarios. A continuación se nos planteó el problema de cómo agrupar las lecturas y escrituras de los diferentes parámetros. Para darle solución imaginamos el proceder habitual del usuario con el controlador EMAX:

- Lo primero que debe hacer es configurar el aparato para el tipo de operación a realizar. Esto, se hace adaptando los parámetros de configuración (CF-xx) a sus necesidades.
- A continuación debe calibrar las unidades y rango de las distintas entradas y salidas, y así puede tener medidas manejables y coherentes de las variables implicadas. Para ello tiene que determinar y rellenar los parámetros de calibración (C-xx).
- Una vez hecho esto, si quiere controlar un proceso, deberá fijar los parámetros de control (F-xx).
- Por último, se dedica a observar los valores instantáneos de las variables implicadas. En este grupo estarán, el punto de consigna y su tipo, la salida del controlador, la variable del proceso... etc., es decir, todas aquellas que son directamente visualizables en las pantallas alfanuméricas y LEDs del controlador.
- En el caso del EMAX el usuario tiene además la posibilidad de efectuar presintonías antes de empezar a controlar, y de utilizar las prestaciones del funcionamiento adaptativo (sin actualización directa de parámetros de control o con actualización automática).

Teniendo en cuenta los razonamientos anteriores, se han clasificado los parámetros en los cinco grupos que aparecen recogidos en la figura 3. Partiendo de esa clasificación, el nivel de aplicación se ha implementado con una función de lectura y escritura para cada conjunto de parámetros, debido a la conveniencia de que puedan ser leídos o escritos independientemente los unos de los otros. Esta exigencia es debida a que en muchas ocasiones sólo se modifican las funciones de control y las variables visualizadas en el EMAX, permaneciendo constantes los parámetros de configuración y calibración.

Cada parámetro se corresponde con una variable del mismo tipo. En el caso de los parámetros de configuración, que se corresponden con bits, y otros que indican un estado, es decir, los que no son números reales, se harán equivalentes a variables tipo "string" que tomen un valores explicativos para facilitar la labor al usuario.

El nivel de aplicación se encarga también de mantener siempre la coherencia entre el valor de las variables identificadas con los parámetros y la presentación de éstas al usuario.

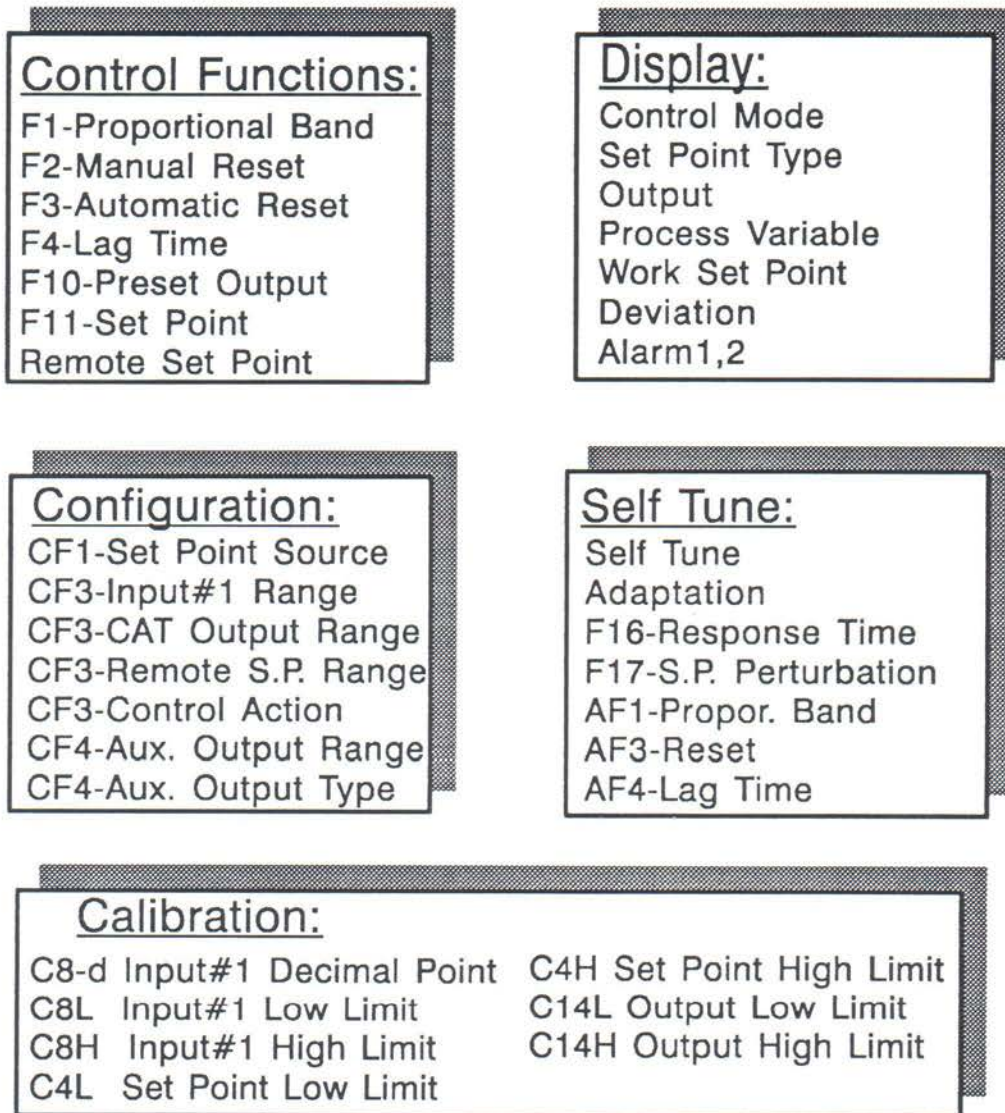


Fig. 3: Clasificación de parámetros en el nivel de aplicación.

### 2.3. LA INTERFASE CON EL USUARIO.

Según se explicó en el subapartado anterior, se puede dividir la labor del usuario del controlador en dos partes bien diferenciadas. La primera, es una fase de preparación del experimento, que conlleva la configuración, calibración y establecimiento de los parámetros o funciones de control. La segunda, es una fase de observación y supervisión de la evolución del proceso con los parámetros introducidos.



Partiendo de la clasificación anterior, se divide el funcionamiento del programa en dos modos exclusivos: modo de "Control" y modo de "Supervisión". El primero se encargará de la lectura y modificación de todos los parámetros seleccionados y el segundo de la supervisión del proceso. A cada modo se le asocia una ventana distinta que es seleccionable por el usuario.

Al **modo de "Supervisión"** se accede por defecto. En él se están leyendo continuamente, a intervalos marcados por el usuario, los parámetros visualizados por el panel frontal del controlador, que son: el Modo de Funcionamiento (manual ó automático), el Tipo de Punto de Consigna (local ó remoto), las Alarmas, el Punto de Consigna, la Salida del Controlador, la Variable del Proceso y la Desviación con respecto al punto de consigna. Véase la figura 4.

Para facilitar aún más la labor del usuario, las tres variables más importantes (Punto de Consigna, Salida del Controlador y Variable del Proceso) se visualizan en forma de diagrama de barras. Esto, permitirá ver rápidamente y desde lejos una instantánea de su evolución relativa. Además, para que quede constancia de dicha evolución, también se visualizan dichas variables en una gráfica respecto a un eje de tiempos. Las escalas utilizadas son las definidas por los parámetros de calibración.

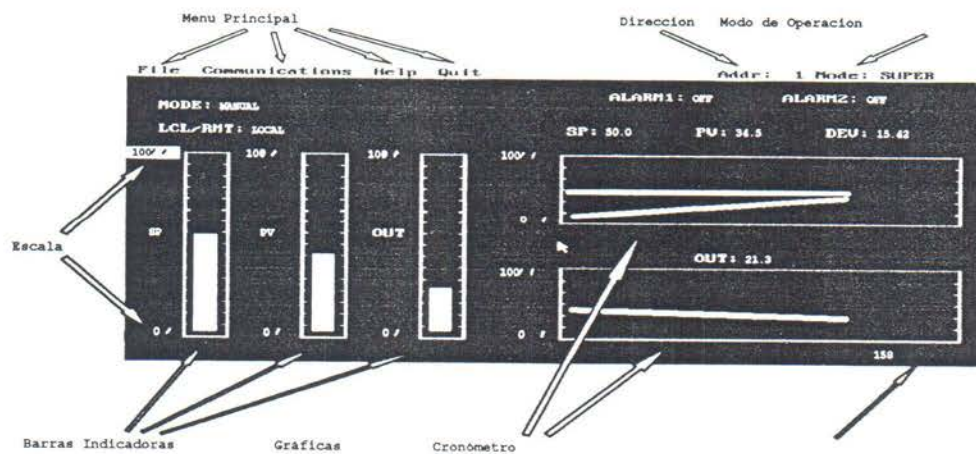


Fig. 4: Pantalla cuando el modo de Supervisión es el activo.

En el **modo de "Control"** se visualizan todos los parámetros aunque permite su lectura y escritura por grupos separados, como ya se vio en el Nivel de Aplicación. En este modo, tras una lectura general inicial, la lectura ó escritura de parámetros es siempre

iniciada por el usuario, véase figura 5. Los parámetros se pueden leer o escribir sobre un fichero ASCII. También se pueden imprimir.

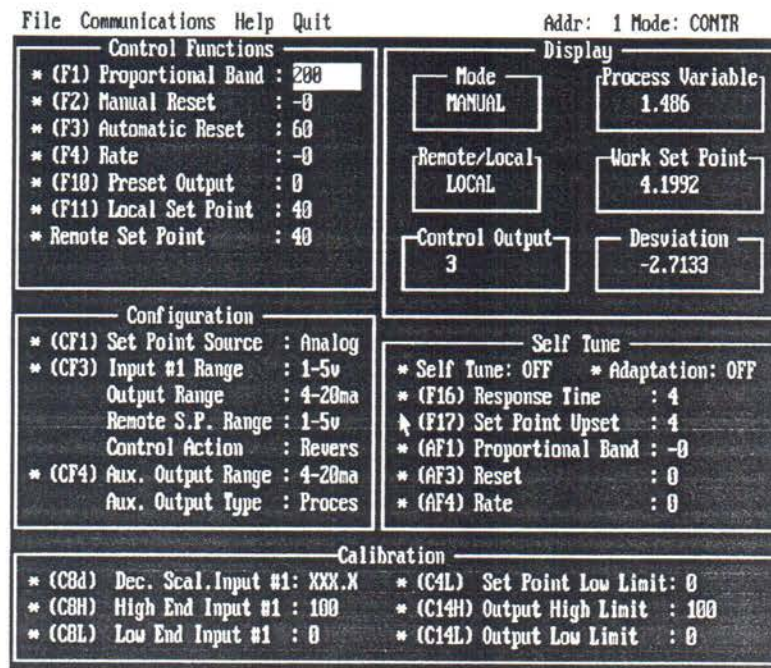


Fig. 5: Pantalla cuando el modo de Control es el activo.

Funciones comunes a los dos modos son la selección de los parámetros de la comunicación serie, velocidad, paridad... etc. Todas estas funciones junto con la dirección del controlador y el modo de funcionamiento del programa son accesibles desde un menú principal. Existe una ayuda dependiente del contexto y de uso del teclado.

Todos los parámetros, funciones... etc de la interfase están en inglés. Esto se debe a que todos los manuales del controlador están en este idioma y el usuario está mucho más familiarizado con esta terminología; por tanto, de esta forma, se evitan confusiones. Para mantener la coherencia, también las ayudas y menús están en inglés.

### 3. CARACTERISTICAS DEL "SOFTWARE" DESARROLLADO.

Todo el "software" se ha desarrollado en lenguaje C dentro del entorno Turbo C Versión 2.0 de Borland. Se consideró que este lenguaje era el más apropiado pues existía la posibilidad de tener que hacer en ensamblador las partes de más bajo nivel del código. También se tuvo en cuenta, a la hora de tomar la decisión, el contar con una librería para realizar interfases amigables escrita en este lenguaje como es C-scape y la librería gráfica

del mismo Turbo C. El conocimiento de ciertas funciones de estas librerías es fundamental a la hora de comprender la codificación de las funciones que componen la interfase con el usuario. Para una información detallada ver el manual "The C-scape Interface Management System".

El código está dividido en doce ficheros diferentes con la intención de mantener estructurado el programa y agrupar las funciones por niveles según lo establecido en los objetivos del proyecto OCRE, aunque, en algún caso, esta división ha sido obligada ante la imposibilidad de compilar demasiadas funciones juntas debido a la limitación de memoria. La figura 6 muestra los enlaces entre los diferentes niveles.

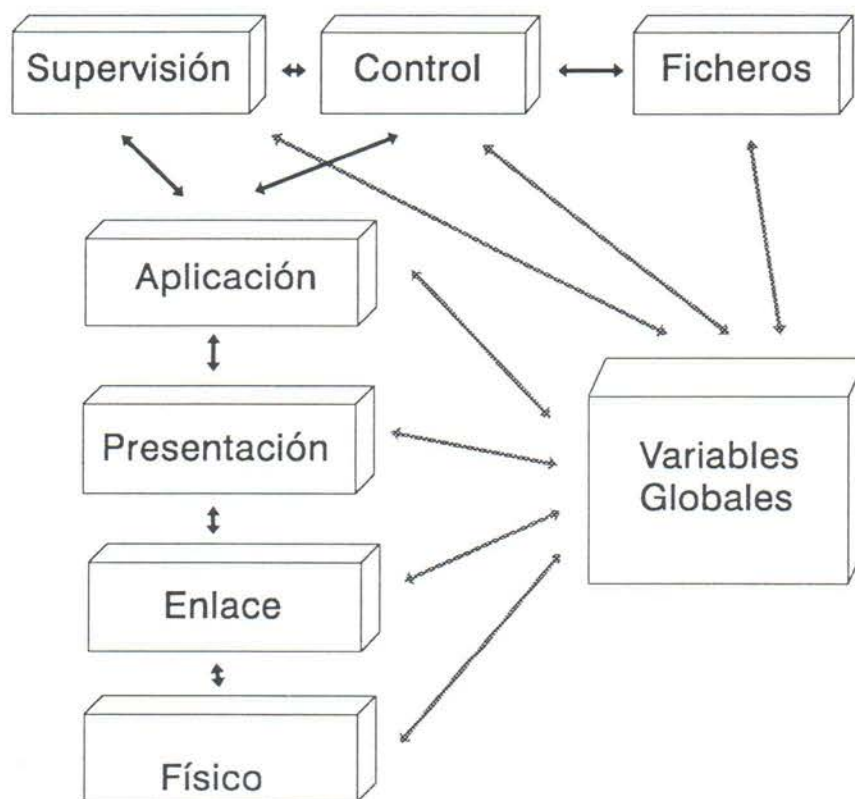


Fig. 6: Estructura Software del proyecto OCRE.

Los ficheros que componen el programa fuente son:

- Variables Globales: VARGLOBA
- Nivel Físico: NFISICO
- Nivel de Enlace: NENLACE
- Nivel de Presentación: NPRESENT
- Nivel de Aplicación: NAPLICA
- Interfase con el usuario: VENTANAS, AUXVENT1, AUXVENT2 y

## GRAFICAS

- Funciones para el manejo de ficheros: SHELL y AUXSHELL.
- Programa Principal: OCRE

### 3.1. MANUAL DE USUARIO.

Como parte de los objetivos del proyecto OCRE se ha escrito un manual completo de usuario en el que se incluye:

- Información sobre el conjunto de programas que lo componen.
- Requisitos "hardware" para su utilización.
- Procedimientos para la instalación.
- Descripción general de los modos de operación.
- Descripción pormenorizada de todas las funciones, clasificadas por ventanas.
- Información sobre las posibles causas de error.

### 3.2. VALIDACIÓN Y PRUEBAS.

A lo largo del desarrollo del proyecto se efectuaron un conjunto de pruebas encaminadas por una parte a comprobar que OCRE cumplía todos los objetivos iniciales y que pasaba un control de calidad. Estas pruebas fueron de dos tipos:

**Comunicaciones.** Se probó la comunicación con el controlador a las distintas velocidades, paridad, puertos, etc., observándose que a las velocidades de 300 y 1200 baudios nunca daba problemas, incluso en ordenadores tipo PC XT, sin embargo, a la velocidad de 9600 baudios sólo funcionaba en ordenadores con gran velocidad de proceso, PC AT, 386, etc. Por ello, la velocidad elegida por defecto ha sido 1200 baudios. Se observó que, para comunicaciones con parámetros distintos a los elegidos por defecto, resultaba bastante incómodo tener que ejecutar el programa y cambiar dichas opciones desde dentro una vez comprobado el error de comunicación. Para evitar ésto, se permite llamar a OCRE con una línea de comandos especificando la dirección del controlador elegido, puerto de comunicaciones serie ("COM1" ó "COM2"), la velocidad de transmisión, VEL300, VL1200 ó VEL9600 y la paridad, "ODD" o impar ó "EVEN" o par. También, se probó el acceso a todos los parámetros del controlador que OCRE permite, tanto en lectura como en escritura.

**Usuario.** Después de una serie de pruebas teniendo en cuenta los errores que puede cometer el usuario, se ha intentado hacer a OCRE lo más robusto posible a este tipo de fallos. Es difícil tener en cuenta todas las posibilidades por lo que el programa se deberá

ir mejorando conforme el usuario vaya encontrando problemas. Otra de las pruebas realizadas fue utilizar OCRE con monitores monocromo, observándose que no existía suficiente contraste debido a los colores elegidos. La solución fue hacer una nueva versión en blanco y negro.

#### 4. CONCLUSIONES.

Las posibles aplicaciones de OCRE, como ya se indicó en la introducción, se centran en pequeñas plantas industriales y laboratorios de investigación donde existan controladores del tipo EMAX. Gracias a OCRE se consigue agilizar la supervisión y control de estos equipos y por tanto de los procesos que controlan.

El proyecto OCRE está abierto a la inclusión de muchas más utilidades. Entre ellas: la posibilidad de comunicar con otro tipo de controladores, la monitorización de varios controladores a la vez, la aplicación de estrategias de control combinadas y la creación de ficheros de datos con la evolución de las variables más importantes.

#### BIBLIOGRAFÍA.

- "Operator's Manual for Electromax V". Leeds & Northrup Instruments 277650. 1982.
- "Electromax V. User's Manual. Serial Communications". Leeds & Northrup 277669 Rev-E. 1982.
- "Supplementary Operator's Manual for Electromax V SelfTune Option". Leeds & Northrup Instruments 277565. 1982.
- Kernighan - Ritchie. "The C Programming Language". 2ª Edición. Prentice Hall Software Series. 1988.
- "Turbo C. User's Guide". Versión 2.0. Borland International. 1988.
- Herbert Schildt. "Turbo C. Programación Avanzada". Borland - Osborne / Mc Graw Hill. 1989.
- "The C-scape Interface Management System Manual". Oakland Group, Inc. 1986 - 1989.
- "Look and Feel Screen Designer Manual". Oakland Group, Inc. 1989.
- CD 8482 - Information Technology - Telecommunications and Information Exchange Between System - Twisted Pair Multipoint Interconnections. ISO/IEC 1992.