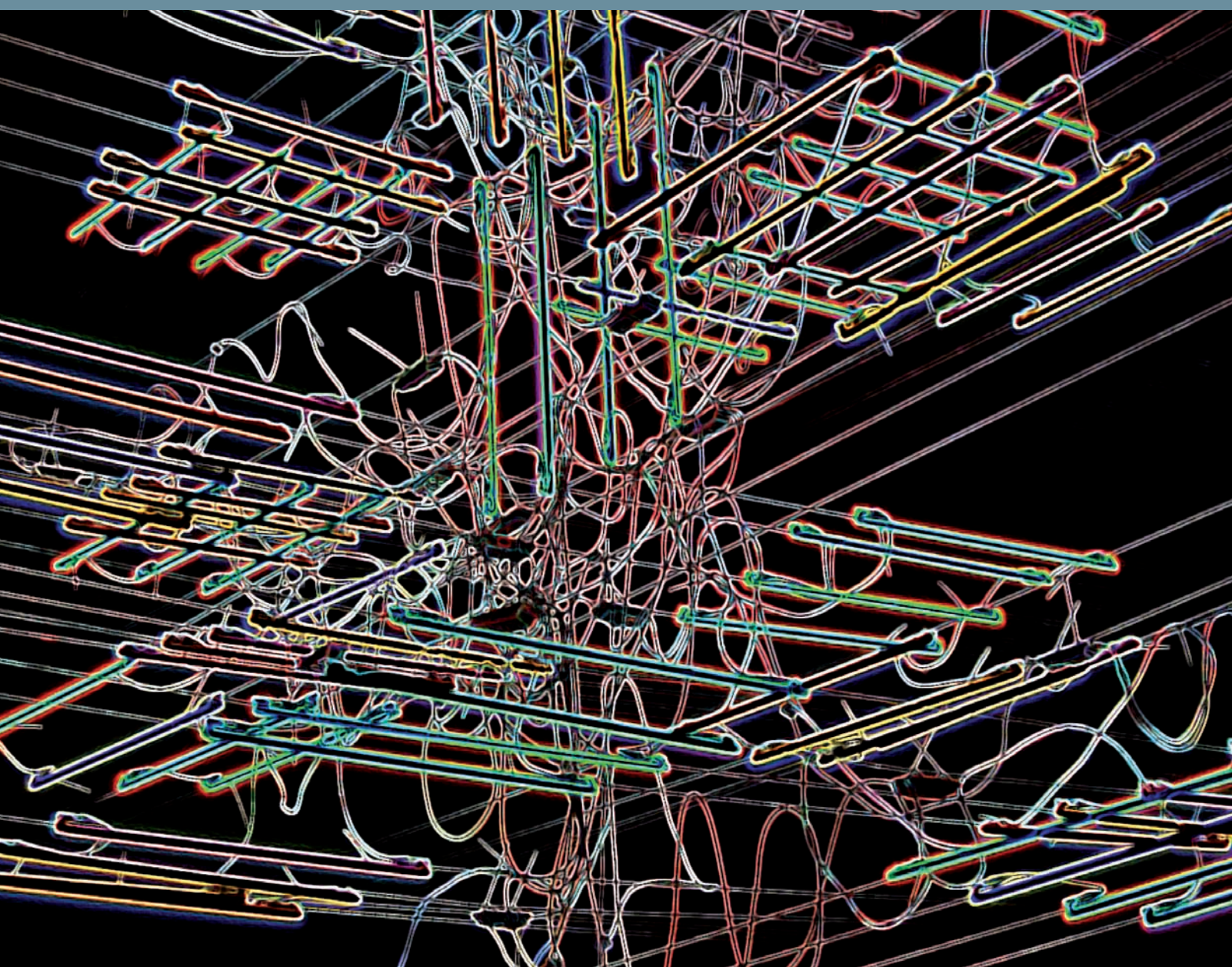


# Optimización del tráfico de información con buses de campo

FRANCISCO J. ORCAJO CAMPILLO



La respuesta en el intercambio de datos de un sistema automatizado mediante bus de campo depende de las características intrínsecas del bus implantado, pero también, de la gestión que realicemos en ese bus. La instalación de buses “lentos” es válida cuando no se requieren tiempos de respuesta reducidos. Sin embargo, si realizamos una gestión adecuada, podemos implementar este tipo de buses –abiertos y de coste reducido– en aplicaciones más exigentes como las del comando de equipos.

Un bus de campo es un sistema de transmisión de información (datos) que simplifica enormemente 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 tradicional bucle de corriente 4-20 mA.

Típicamente son redes digitales, bidireccionales, multipunto, montadas sobre un bus serie, que conectan dispositivos de campo como PLC, 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 será capaz de ejecutar funciones simples de diagnóstico, control o mantenimiento, así como de comunicarse bidireccionalmente a través del bus<sup>1</sup>.

Esta definición nos resulta útil como punto de partida para el problema que pretendemos abordar: ¿Cómo podemos optimizar los intercambios de información entre el gestor del bus y los diferentes participantes en el mismo? Nuestro problema se agudiza cuando pretendemos obtener respuestas rápidas –p.e. el arranque y parada de diferentes dispositivos– sobre un bus de baja velocidad.

### La arquitectura maestro-esclavo

Una gran parte de los buses de campo industriales responden a la arquitectura maestro-esclavo (del inglés Master-Slave). En este caso, el maestro es el equipo que realiza las peticiones (lectura y escritura de variables) y los diferentes esclavos integrados en el bus atienden las peticiones que reciben del maestro (figura 1).

### Buses lentos

El problema de la optimización del tráfico de la información es especialmente

crítico cuando nuestra instalación está dotada de un “bus lento”. Este tipo de buses disponen de velocidades de transmisión menores a los 100 kbits/s. El más extendido de todos ellos es MODBUS, desarrollado por Modicon<sup>2</sup> y que se ha convertido en un estándar de facto, puesto que está abierto a quien quiera utilizarlo. En una instalación típica, nos encontraremos con un MODBUS configurado a una velocidad de 19,2 kbits/s. En este caso, la gestión de una trama de 252 bytes ocupará un tiempo mayor de 100 milisegundos. Este dato es de una magnitud tal, que de no realizarse una gestión adecuada, se provocará el colapso del bus por la acumulación de peticiones por parte del maestro que no pueden ser atendidas o, como mínimo, obtendremos tiempos de respuesta inaceptables para el comando de equipos.

Como vemos en la figura 1, existe un intercambio de datos entre la aplicación de usuario –el programa del PLC– y el módulo de comunicación<sup>3</sup> que es quien sirve las peticiones de la primera. Cuando se recibe una petición –ya sea de lectura o de escritura– el módulo de comunicación debe gestionarla: lanzarla al bus y esperar una respuesta. En este tipo de buses, no se pueden simultanear diferentes peticiones sobre el medio físico, por lo que debemos garantizar que siempre que nuestro programa realice una petición el comunicador está libre para recibirla. Ciertamente, el propio módulo de comunicación suele disponer de una memoria en la que es capaz de almacenar diferentes peticiones y “gestionarlas”. Pero ¿qué ocurrirá si de forma permanente enviamos más peticiones de las que es capaz de atender? La respuesta es obvia: colapsaremos el módulo.

Existe un dato fundamental para que podamos diseñar una aplicación que no sobrecargue la capacidad de nuestro sistema de comunicaciones: es el tiempo de gestión del módulo de comunicaciones,  $T_{gmc}$ , para dar respuesta a cada una de las peticiones que recibe, y que deberemos consultar en la documentación técnica del fabricante (figura 2).

De lo anterior, la frecuencia máxima de las peticiones,  $F_{mp}$  en Hz, que realicemos al módulo será:

$$F_{mp} \leq \frac{1}{T_{gmc}}$$

Si no conocemos el dato del tiempo de gestión de comunicaciones, podemos realizar la aproximación siguiente:

$$F_{mp} \leq \frac{C_t}{K_s * 8(L_{trama} + 12)}$$

En donde

$C_t$  es el caudal de transmisión en bits/s; 19.200 bits/s, por ejemplo.

$L_{trama}$  es la longitud de los registros que queremos leer o escribir, en bytes. Le sumamos el valor 12 (bytes) que es el tamaño mínimo de la trama de petición de comunicaciones en el caso más desfavorable.

$K_s$  es un factor de seguridad que nos evita el solapamiento de peticiones,  $K_s > 1$ ; se aconseja tomar  $K_s = 1,5$

Como vemos, una vez definido el caudal de transmisión,  $C_t$  –que será el máximo que nos acepten el conjunto de participantes en el bus–, la frecuencia máxima de las peticiones,  $F_{mp}$ , es inversamente proporcional a la longitud de la trama, por lo que aconsejamos realizar peticiones para tramas menores o iguales a los 40 octetos.

Una vez que tenemos el valor de la frecuencia máxima de las peticiones, podemos generar en nuestra aplicación PLC un reloj interno que sirva para gestionar las peticiones lanzadas por el Maestro. Aquí tendremos, obviamente, que:

$$F_{reloj} \leq F_{mp}$$

Este reloj interno gestionará nuestras peticiones, pero ¿cuáles son estas peticiones? Distinguiremos, como mínimo, tres tipos.

**Lectura del estado** de los componentes del bus. Como ya hemos dicho antes, diseñaremos estas lecturas con tramas cortas, aunque sobre algún equipo fuese preciso realizar diferentes demandas de lectura para completar los datos que precisemos de ese dispositivo.

**Escrituras de parámetros.** Que en la mayoría de los casos consideraremos “no prioritarias” en el sentido de que no precisan de un tiempo de respuesta crítico.

**Comandos.** Son órdenes prioritarias sobre los Esclavos que precisan ser ejecutadas en el mínimo tiempo posible, independientemente del número de componentes del sistema y con un tiempo máximo de respuesta garantizado.

Ahora que tenemos un reloj para el control de las comunicaciones ( $F_{reloj} < F_{mp}$ ) disponemos de un patrón que nos permite gestionar el lanzamiento de peticiones con la certeza de que no saturaremos nuestro módulo de comunicaciones.

En nuestro caso, proponemos ejecutar las peticiones de lectura de forma

## RESUMEN

Las soluciones de automatismo que utilizan buses de campo para el intercambio de datos y el comando de equipos están experimentando un notable crecimiento. Este artículo presenta algunas técnicas que permiten optimizar la respuesta del sistema, fundamentalmente en los casos en los que la velocidad de intercambio de datos es reducida y queremos garantizar los tiempos de respuesta independientemente del número de participantes en el bus.

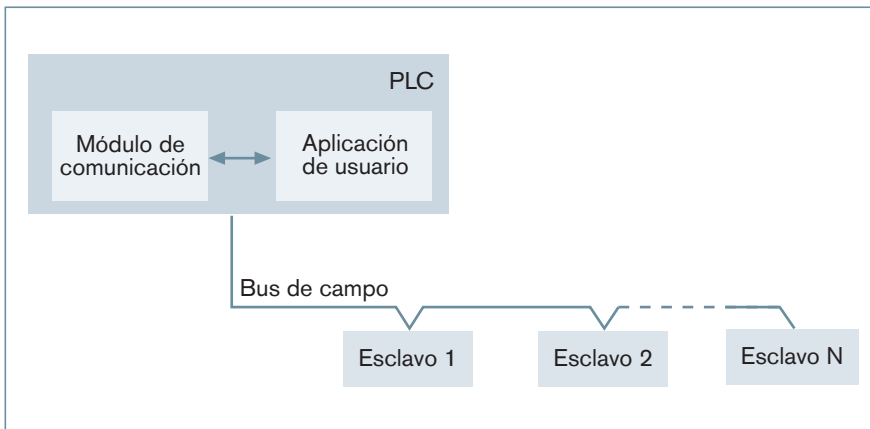


Figura 1. Arquitectura de un bus de campo.

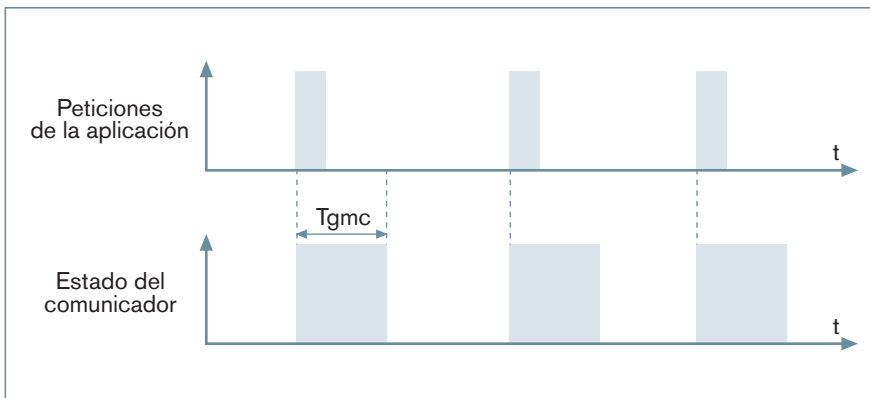
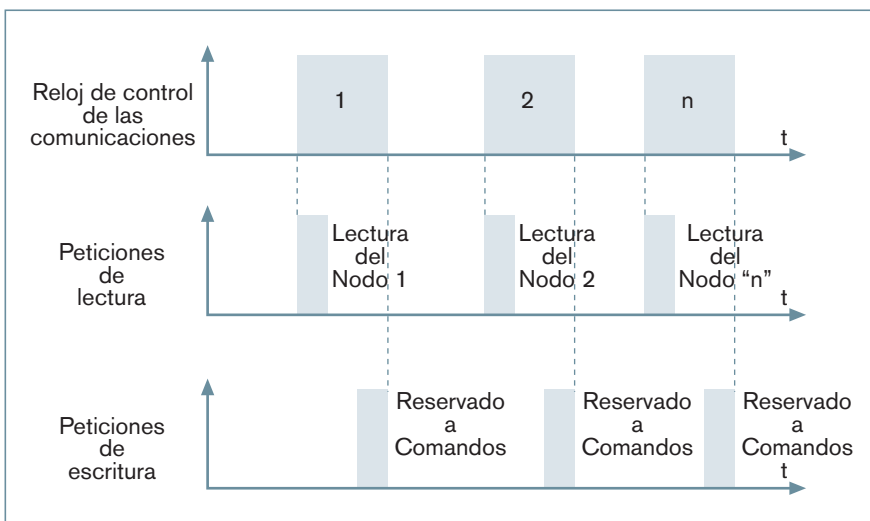


Figura 2. Cronograma peticiones-estado del módulo de comunicaciones.

Figura 3. Cronograma de peticiones gestionadas por un reloj de control.



cíclica (figura 3) de manera que (en el caso más simple que será cuando sólo lancemos una sola trama por cada esclavo) tendremos que el tiempo de refresco del estado de cada esclavo,  $Trslv$ , será:

$$Trslv = \frac{1}{n(Freloj)}$$

En donde:

$n$  es el número de esclavos en el bus.

$Freloj$  es la frecuencia del reloj de control de las peticiones que habíamos determinado previamente.

Con esta solución garantizamos el tiempo de lectura –tiempo de refresco del estado de los esclavos– pero debemos gestionar nuestras peticiones de escritura de manera que no se produzcan de manera simultánea. Si ello no fuese posible, deberíamos realizar la siguiente variante:

Lecturas cíclicas gestionadas por el reloj de control de las comunicaciones.

Escrituras por excepción. En este caso, ante la petición de una solicitud de escritura se interrumpirá el funcionamiento del reloj.

Cualquiera de estas soluciones nos garantiza que las peticiones de escritura se ejecutan con el mínimo retardo dependiendo únicamente de las características intrínsecas del *hardware* utilizado. Por otro lado, el refresco del estado de los equipos estará en función del *hardware* y del número de componentes del bus; como no puede ser de otro modo.

### Prioridad en las peticiones

Otro de los elementos que debemos jerarquizar es la prioridad en las peticiones. Lo entenderemos más fácilmente con un ejemplo. Si queremos comandar un equipo en bus, necesitaremos saber su estado (marcha, paro o defecto) que estará almacenado en la tabla “Estado”. Los posibles defectos de ese equipo estarán en la tabla “Defectos”, la intensidad consumida por este equipo, en “Intensidad”, etc. En nuestro caso, la tabla “Estado” es la información prioritaria y deberemos leerla periódicamente. Sin embargo, la tabla “Defectos” la leeremos sólo si “Estado” nos indica que es preciso hacerlo e “Intensidad” únicamente cuando el equipo esté en marcha. De esta manera, reduciremos la longitud de las tramas y realizaremos las peticiones, únicamente, cuando nos aporten información relevante.

De lo anterior, obtenemos una serie de conclusiones que nos pueden resultar muy útiles en el diseño de nuestras arquitecturas de automatización.

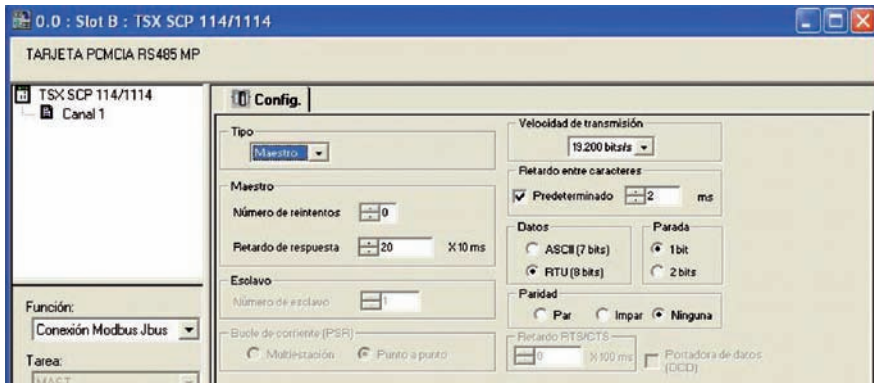


Figura 4. Configuración de una tarjeta PCMCIA que actúa como maestro MODBUS.

## Conclusiones

Cuando precisemos tiempos de refresco muy cortos, deberemos reducir el número de esclavos que componen el bus y reducir el tamaño de las tramas de lectura porque el tiempo de refresco depende de estos dos parámetros. Como el número de esclavos de la instalación no es una “variable”, habrá que estudiar, en determinadas ocasiones, la necesidad de instalar varios buses, por debajo del número máximo que esté determinado en su topología. Sólo así obtendremos tiempos de respuesta reducidos en los llamados buses lentos.

Otro de los elementos sobre los que podemos actuar es la longitud de la trama. En aplicaciones críticas, tendremos que lanzar tramas de lectura de la menor longitud posible –con los valores fundamentales únicamente– realizando una gestión específica para el resto de lecturas no prioritarias de manera que no sobrecarguemos la capacidad de nuestro bus.

Como vimos al inicio, el tiempo de gestión de una trama de 252 octetos en bus de 19,2 kbits/s es de unos 100 ms. Si somos capaces de reducir nuestras tramas a unos 20 octetos –siguiendo la estrategia de priorizarlas– obtendremos tiempos de res-

puesta de 10 ms (será necesario consultar en los catálogos del fabricante el dato del tiempo mínimo de gestión de una trama). Si trabajamos con tramas “priorizadas” gestionadas por un reloj, en el ejemplo que estamos analizando, tendremos que, en un bus con 20 esclavos, realizaremos una lectura cada 200 ms. Este orden de magnitud nos permitirá comandar equipos en bus siempre y cuando el número de participantes sea menor de unos 30 equipos.

## Bibliografía

Definición tomada de la Wikipedia en [http://es.wikipedia.org/wiki/Bus\\_de\\_campo](http://es.wikipedia.org/wiki/Bus_de_campo)  
 Modicom es posiblemente el primer fabricante de PLC y el responsable del desarrollo del bus Modbus.  
 El módulo de comunicación es un dispositivo *hardware* dedicado, integrado en el PLC y que dispone de una configuración específica.

## AUTOR

**Francisco Orcajo Campillo**  
 francisco@orcajocampillo.com  
 www.orcajocampillo.com

Ingeniero técnico industrial por la Escuela Politécnica de Burgos. Actualmente desempeña el ejercicio libre de la profesión diseñando e implementando soluciones programadas con PLC y SCADA. Anteriormente trabajó en el Grupo Schneider, en T.J. Bocanegra y en Telettra Española.

**UrbicAD architecture SL**  
 Software para el desarrollo de Planes de Autoprotección, Emergencia y Evacuación.

**UrbicAD DB-SI/ Planes de Autoprotección**

- Memoria de cumplimiento del **CTE DB-SI** o del **RD 2267/2004** para establecimientos Industriales.
- Desarrollo de Planes de Autoprotección conforme **NB-Autoprotección (RD 393/2007)**.
- Gestión de Emergencias y situaciones de crisis.

[www.urbicad.com](http://www.urbicad.com) // Tel. 963 492 144

**PREPARACIÓN A DISTANCIA Y PRESENCIAL**

**CATEDRÁTICOS Y PROFESORES DE ENSEÑANZA SECUNDARIA**

~ FILOSOFÍA ~ LATÍN Y CULTURA CLÁSICA ~ GRIEGO Y CULTURA CLÁSICA ~ LENGUA CASTELLANA Y LITERATURA ~ GEOGRAFÍA E HISTORIA ~ MATEMÁTICAS ~ FÍSICA Y QUÍMICA ~ BIOLOGÍA Y GEOLOGÍA ~ DIBUJO ~ INGLÉS ~ FRANCÉS ~ ALEMÁN	~ MÚSICA ~ EDUCACIÓN FÍSICA ~ PSICOLOGÍA Y PEDAGOGÍA ~ TECNOLOGÍA ~ ECONOMÍA ~ FORMACIÓN Y ORIENTACIÓN LABORAL ~ ADMINISTRACIÓN DE EMPRESAS ~ ORGANIZACIÓN Y GESTIÓN COMERCIAL ~ INFORMÁTICA ~ ORGANIZAC. Y PROYECTOS	DE FABRICACIÓN MECÁNICA ~ ORGANIZAC. Y PROCESOS DE VEHÍCULOS ~ ORGANIZAC. Y PROYECTOS DE SISTEMAS ENERGÉTICOS ~ SISTEMAS ELECTROTÉCNICOS Y AUTOMÁTICOS ~ SISTEMAS ELECTRONICOS ~ CONSTRUCCIONES CIVILES Y EDIFICACIÓN ~ PROCESOS DIAGNÓSTICOS CLÍNICOS Y ORTOPÉDICOS ~ PROCESOS SANITARIOS	~ PROCESOS EN LA INDUSTRIA ALIMENTARIA ~ INTERVENCIÓN SOCIOCOMUNITARIA ~ HOSTELERÍA Y TURISMO ~ PROCESOS Y MEDIOS DE COMUNICACIÓN ~ ASESORIA Y PROCESOS DE IMAGEN PERSONAL ~ ANÁLISIS Y QUÍMICA INDUSTRIAL ~ PROCESOS DE PRODUCCIÓN AGROARIA
---	--	---	--

**PROFESORES TÉCNICOS DE FORMACIÓN PROFESIONAL**

~ PROCESOS DE GESTIÓN ADMINISTRATIVA ~ PROCESOS COMERCIALES ~ SISTEMAS Y APLICACIONES INFORMÁTICAS ~ MECANIZADO Y MANTENIMIENTO DE MÁQUINAS ~ SOLDADURA ~ INSTALAC. Y MANTENIM. DE EQUIPOS TÉRMICOS Y DE FLUIDOS ~ MANTENIMIENTO DE VEHÍCULOS ~ INSTALACIONES ELECTROTÉCNICAS ~ EQUIPOS ELECTRONICOS	~ OFICINA DE PROYECTOS DE CONSTRUCCIÓN ~ PROCEDIMIENTOS SANITARIOS Y ASISTENCIALES ~ PROCEDIMIENTOS DIAGNÓSTICO CLÍNICO Y ORTOPÉDICO ~ OPERACIONES Y EQUIPOS DE PRODUCTOS ALIMENTARIOS ~ SERVICIOS A LA COMUNIDAD ~ COCINA Y PASTELERÍA ~ SERVICIOS DE RESTAURACIÓN ~ TÉCNICAS Y PROCEDIMIENTOS DE IMAGEN Y SONIDO ~ OPERACIONES DE PRODUCCIÓN AGROARIA
--	---

**CATEDRÁTICOS Y PROFESORES DE ESCUELA OFICIAL DE IDIOMAS**

~ INGLÉS	~ ESPAÑOL	~ FRANCÉS	~ ALEMÁN
----------	-----------	-----------	----------

**MAESTROS DE ENSEÑANZA PRIMARIA**

~ EDUCACIÓN PRIMARIA ~ EDUCACIÓN INFANTIL	~ INGLÉS ~ FRANCÉS	~ EDUCACIÓN FÍSICA ~ AUDICIÓN Y LENGUAJE	~ EDUCACIÓN MUSICAL ~ PEDAGOGÍA TERAPÉUTICA
--	-----------------------	---	--

**CEDE**  
 C/ CARTAGENA, 129 - 28002 MADRID  
 TELS.: 91 564 42 94 - FAX: 91 563 60 54  
 www.cede.es - E-mail: oposiciones@cede.es