

LABORATORIO REMOTO DE MICROPROCESADORES Y MICROCONTROLADORES

Antoni Grau

Dept. de Ingeniería de Sistemas, Automática e Informática Industrial (ESAI)
Universitat Politècnica de Catalunya (UPC)
c/ Pau Gargallo, 5 08028-Barcelona
e-mail: agrau@esaii.upc.es

Resumen. El laboratorio remoto de microprocesadores y microcontroladores es un entorno orientado al aprendizaje de la programación de sistemas microprocesadores y microcontroladores, con la particularidad de que puede ser accedido a través de Internet desde cualquier ordenador. Está integrado por un conjunto de componentes que permitirán a un usuario acceder a él de manera remota utilizando un navegador de Internet y conectándose a una Web. Esto le suministrará remotamente el control de un conjunto de sistemas microcomputadores que podrá programar de manera independiente y ejecutar tareas y aplicaciones reales sobre el laboratorio.

1. Introducción y descripción del sistema

Cada día más la sociedad actual tiende a hacer un uso más extenso de las nuevas tecnologías para sus actividades cotidianas. En el caso de la población universitaria, este uso se ha de canalizar hacia un buen aprovechamiento del tiempo invertido y de los recursos que la Universidad pone a disposición de los alumnos. En esta línea creemos que es positivo ofrecer a los alumnos la posibilidad de realizar las prácticas de asignaturas referentes a diseño, implementación y desarrollo de *hardware* de manera no presencial.

En concreto, el objetivo del sistema que se propone es desarrollar una aplicación que permita al usuario disponer de una plataforma remota de microprocesadores y microcontroladores sobre los cuales pueda aprender a diseñar y programar aplicaciones, así como ayudarle a adquirir unos conocimientos completos sobre el diseño de sistemas basados en microprocesadores y microcontroladores.

En este proyecto se desarrolla sobre diferentes plataformas, lo que permite conseguir que los conocimientos sobre microcomputadores que adquiera el usuario de la aplicación sean globales y no estén supeditados a la plataforma que utilice. Se han diseñado tres plataformas, dos microcontroladores y un microprocesador. Esta variedad permitirá que aparezcan ventajas e inconvenientes del uso de las diferentes arquitecturas, sus diferencias y sus puntos en común y consolidará aún más los conocimientos del usuario.

Las arquitecturas implementadas son Intel 8088 y Philips 80552, no sólo son compatibles vía *software* con familias tan importantes como las del

8086 y 8051 respectivamente, sino que además tienen una gran aceptación e implantación comercial e industrial. El tercer microcontrolador implementado es el 87C196KD de Intel, que ofrece altas prestaciones y bus de 16 bits.

Este sistema se ha orientado para la asignatura "Diseño de Sistemas basados en Microcomputadores", la cual se imparte en la Facultad de Informática de Barcelona en la Universidad Politècnica de Catalunya. Actualmente se está realizando la programación de placas con alguno de los microcontroladores citados anteriormente (en concreto, el 87C196KD) y se realizan varias prácticas sobre él. La idea propuesta en este trabajo es acceder a una de las placas ya implementadas y ejecutar los programas que forman las prácticas que se ofrecen durante el curso. Como se ha dicho ya, se han implementado varias placas con diferentes microcomputadores.

La figura 1 representa los componentes básicos del sistema y el recorrido seguido por los datos dentro de él, durante la ejecución de un programa sobre alguno de los microcomputadores.

El sistema se activa cuando el usuario selecciona uno de los tres sistemas microcomputador implementados. Sobre cada uno de los microcomputadores se permite programar un conjunto de prácticas que tocan áreas diversas de los sistemas (comunicación serie, *timers*, etc). Es necesario la introducción del programa de usuario correspondiente a la práctica. Finalmente se ejecuta el programa, acción que realiza el usuario pulsando sobre un botón. De esta manera el usuario interacciona con una especie de caja negra que engloba a Internet, el servidor de Internet, la interficie de interconexión entre servidor y placas, y a los sistemas microcomputadores.

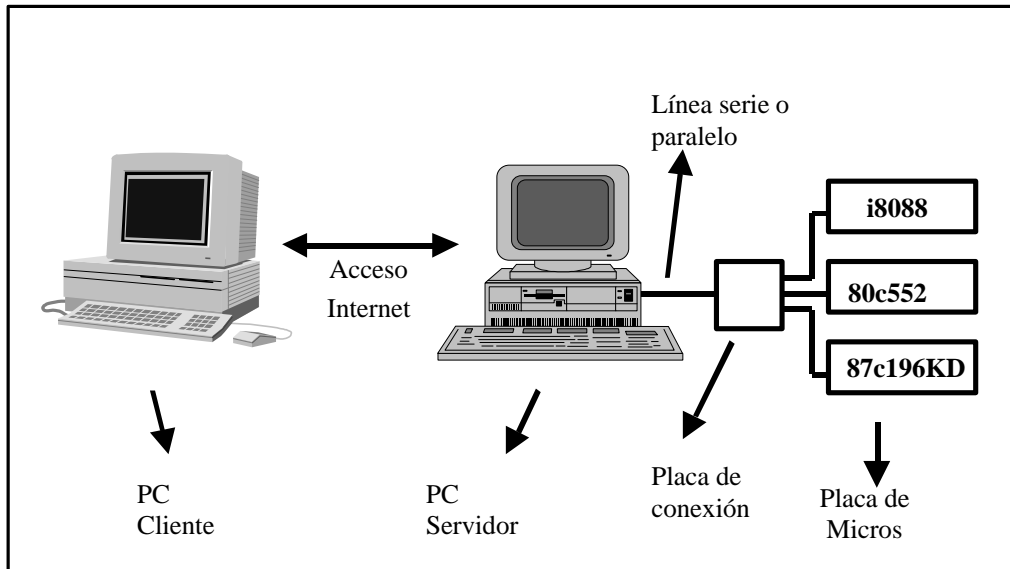


Figura 1. Configuración del sistema de laboratorio remoto.

Dicha caja negra recibe unos parámetros (programa, microcomputador y práctica) y retorna unos resultados (validez e incorrección del programa, resultados).

El PC servidor de Internet, a la llegada del paquete con la especificación del microcomputador, práctica y programa realiza la compilación de este último devolviendo al cliente los mensajes de error encontrados de existir alguno. Si no existen errores, se selecciona el microcomputador especificado, se carga y ejecuta el programa. A su finalización, la placa activa retorna al PC servidor los resultados de ejecución de la práctica y esta información viaja de nuevo al cliente al que se le presentará en la página de su explorador de Internet.

2. Descripción de las prácticas

Dado que cada microcomputador dispone de características propias y únicas, no es posible homogeneizar las prácticas a desarrollar sobre ellos, por lo que se diferencian cada una de ellas. En concreto, veamos a continuación las peculiaridades de cada computador (a grandes rasgos) y cuales son las prácticas que se han propuesto.

Sistema 8088: Se ha elegido este microprocesador por ser la arquitectura que inició la era PC, y que aun es compatible su programación con los sistemas PC actuales. Permite una gran capacidad de interconexión en el bus de cualquier placa de desarrollo adicional y, finalmente, es de gran validez didáctica. La placa de desarrollo basada en el microprocesador Intel 8088 dispone de los siguientes bloques básicos:

- * Sistema de alimentación
- * Sistema MicroProcesador:
 - Generador de Reloj
 - Microprocesador Intel 8088
 - Memoria de datos (RAM)
 - Memoria de programa (EPROM)
 - Decodificación de direcciones (PAL)
- * Interficie de Entrada / Salida:
 - Comunicación serie (USART)
 - Puerto de entrada / salida (PIO)
 - Convertor analógico/digital (ADC)
 - Controlador de interrupciones (PIC)
 - Interval Timer (IT)

* Práctica

1. Comunicación serie: se transmite y se devuelve una cadena de caracteres.
2. Generación de frecuencia: cálculo de frecuencia de señal mediante contadores.
3. Adquisición de datos con el convertor ADC.

Sistema 80C552: La elección de este microcontrolador la encontramos en el hecho que hoy en día la industria usa ampliamente este microcontrolador o alguno compatible de su familia. Ha sido durante muchos años un referente y seguirá siéndolo, por lo que es necesario que los alumnos conozcan este sistema, su arquitectura, programación y utilidades. La placa de desarrollo basada en el microcontrolador Philips 80C552 dispone de los siguientes bloques básicos:

- * Sistema de alimentación
- * Sistema Microcontrolador:
 - Microcontrolador Philips 80552
 - Sistema de memoria de programa
 - Sistema de memoria de datos
- * Interficie de Entrada/Salida

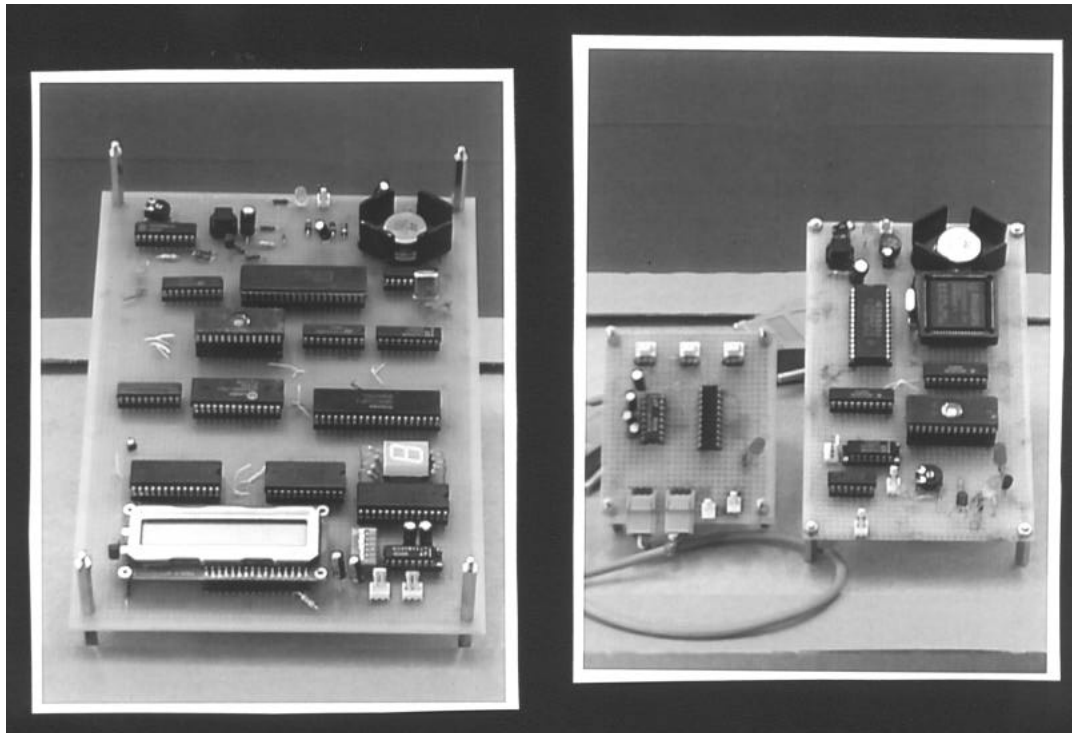


Figura 2. Vista del sistema microprocesador i8088 y del 80552.

* Práctica:

1. Comunicación serie.
2. Generación de frecuencia: Cálculo de frecuencia de señal mediante contadores.
3. Adquisición de datos con el convertor ADC.
4. Control PWM.

Sistema 196KD: Mediante esta placa se dará soporte a los usuarios para realizar prácticas con el microcontrolador de Intel® 87C196KD. Se ha elegido este microcontrolador por la gran cantidad de alternativas que ofrece y principalmente porque se basa en una arquitectura de 16 bits y pertenece a una familia de micros que tiene un amplio conjunto de instrucciones muy diferentes al resto de sistemas del laboratorio remoto.

Entre sus principales características podemos destacar un espacio de direccionamiento de 64 Kb (bus de direcciones de 16 bits) y la posibilidad de manipular datos tanto de 8 como de 16 bits. Para aprovechar esta característica se ha implementado un sistema dual con dos memorias externas, una RAM y otra ROM, de manera que con la segunda que contiene la preprogramación se trabajará en modo 8 bits y se pasará a modo 16 bits cuando se trabaje con la memoria RAM que contendrá los programas que envíe el usuario del laboratorio remoto vía Internet.

Como dispositivos más destacables este microcontrolador ofrece un convertor A/D de 10 bits con 8 entradas analógicas, dos contadores de 16 bits, tres salidas de modulación por pulsos

(PWM), cinco puertos multiplexados de E/S de 8 bits cada uno, una UART full-duplex, un *watch dog timer* (WDT) y un módulo de E/S de alta velocidad con cuatro entradas y seis salidas.

* Prácticas:

1. Programación del puerto serie para la transmisión de datos.
2. Programación del PWM.
3. Programación del convertor A/D.

En la figura 2, se muestran algunas de las placas que se han implementado y han sido usadas en el laboratorio remoto. Los primeros prototipos (foto) han sido montadas con Wire-Wrap, técnica de prototipado que permite cualquier modificación en el circuito hasta que se obtiene la versión definitiva que se montará en circuito impreso. Además de las placas con los sistemas microcontroladores, se han montado varios diseños para poder comprobar el correcto funcionamiento de las prácticas y así cerrar el lazo de ejecución. Por ejemplo, para probar la programación del PWM se ha montado un motor con un disco *encoder* para leer la velocidad de giro y verificar así la correcta ejecución de la práctica.

3. Programación

El laboratorio remoto de microcontroladores y microprocesadores tiene una carga importante de *software* que podemos dividir en dos partes. La

primera y más importante es la que conforma la Web del laboratorio y la segunda es la que se encarga de compilar y cargar los programas en los diferentes microcontroladores.

3.1. Programación WEB

La Web del laboratorio remoto de micros está formada por un conjunto de páginas que en definitiva permiten al usuario enviar programas al servidor, seleccionando previamente un microcomputador con el que trabajar. La página de entrada de datos viene complementada por un conjunto de páginas de ayuda sobre el sistema y sobre las prácticas que se pueden programar.

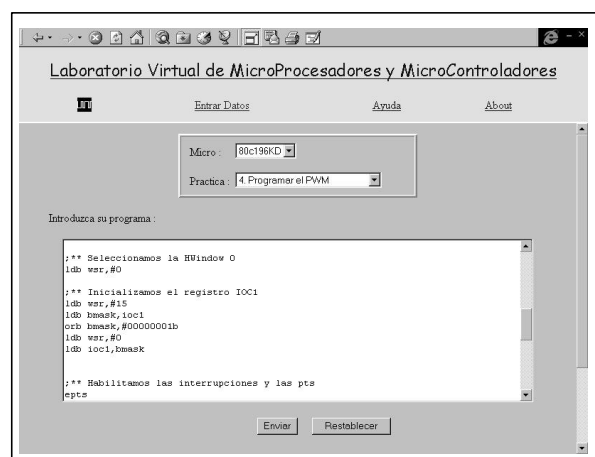
Se ha montado la Web del laboratorio sobre Personal Web Server 4.0 (PWS), una aplicación para Windows 95 que permite simular un servidor de Web de Windows NT pero con menos opciones

de configuración. Este simulador es adecuado para realizar pruebas y para entornos Intranet, por lo que la versión definitiva debería correr sobre Windows NT para ser accesible desde la totalidad del mundo. Para implementar la Web del laboratorio se han implementado páginas HTML convencionales y páginas ASP que se encargan de ejecutar *scripts* en la parte del servidor y devolver resultados a los clientes. La totalidad de los *scripts* han sido implementados en JavaScript por ser un lenguaje muy extendido interpretable por cualquier navegador, siempre que tenga las opciones adecuadas activadas.

Al entrar a la página de bienvenida del laboratorio se informa al usuario de los requerimientos que debe tener su máquina para una correcta utilización del laboratorio, a continuación se muestra la pantalla de bienvenida (figura 3).



a)



b)

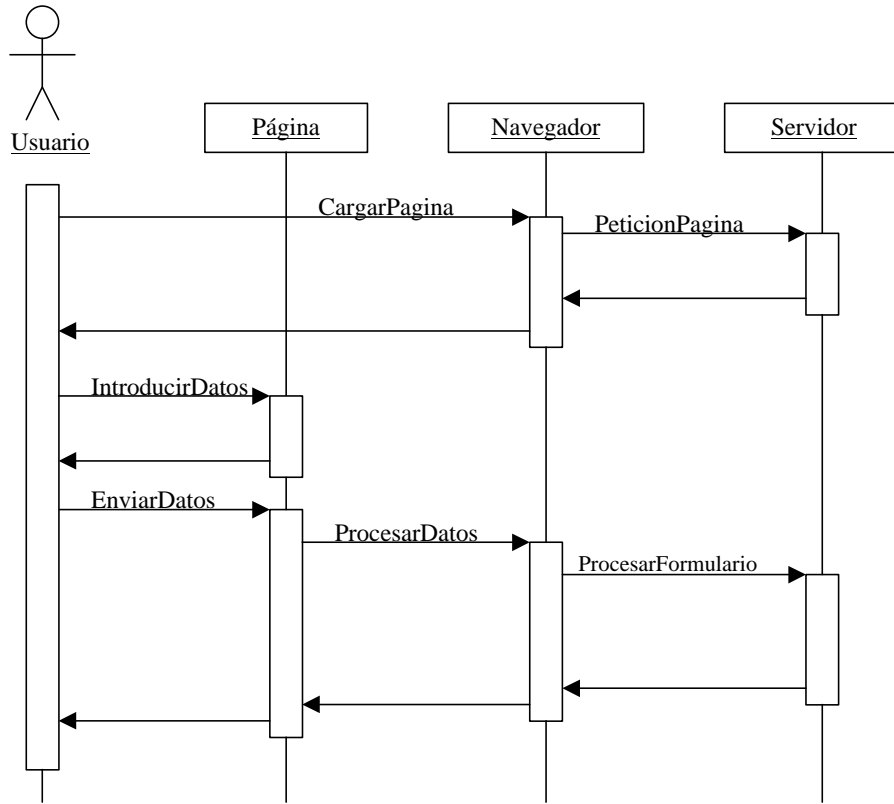
Figura 3. Pantalla de inicio y ventana de programación de las prácticas.

Una vez el usuario ha entrado en el laboratorio, se encuentra con una página de entrada de datos que está compuesta de dos listas desplegables que muestran micros disponibles y prácticas respectivamente, y una caja de texto sobre la cual puede editar sus programas. Por último hay dos botones, uno de envío de prácticas y otro para inicializar la página. En la parte superior se encuentra el menú para acceder a la ayuda y a información sobre el laboratorio y este proyecto en general. Esta pantalla se muestra en la figura 3 b.

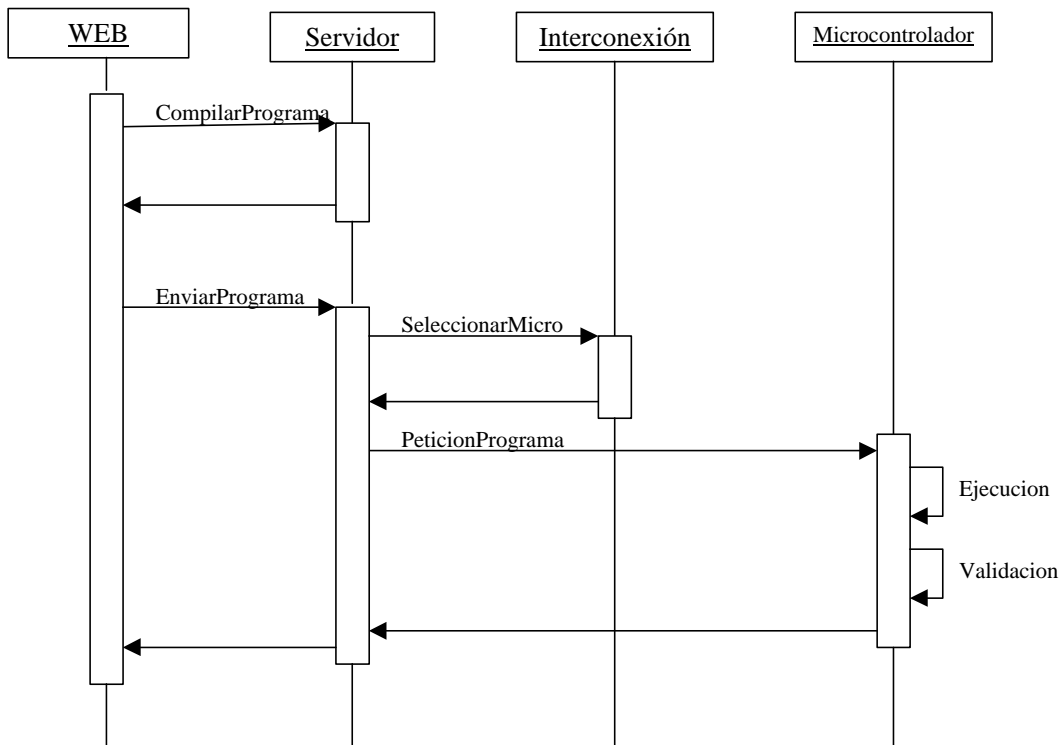
Cuando el usuario ha entrado su práctica y ha pulsado el botón de envío, se inicia la carga de una página ASP que procesará la práctica mediante una secuencia de comandos del servidor. En resumen se trata de la escritura de la práctica en un fichero

de texto (program.a96) y de la generación de un fichero de control (control.txt) para que un programa consumidor de prácticas que corre continuamente en el servidor, se percate de la llegada de una nueva petición de algún usuario.

El programa consumidor se trata de un bucle que detecta la llegada de prácticas para el laboratorio y lanza la compilación, en caso de errores el compilador genera un fichero de extensión *lst* (program.lst) que la página ASP se encarga de devolver al usuario. Si la compilación se lleva a cabo sin errores, el mismo programa consumidor se encarga de lanzar el programa cargador para trasladar el programa a la memoria del microcontrolador y que sea ejecutada.



a)



b)

Figura 4. Diagrama de ejecución de la aplicación WEB: a) desde el cliente hasta el servidor (con Internet); b) desde el servidor hasta los microcontroladores (en local, sin Internet).

Cuando la ejecución ha terminado, el consumidor de prácticas le proporciona a la página ASP los resultados de la ejecución por medio dos ficheros; uno de control (control.res), que indica si la ejecución se ha realizado con éxito o si por el contrario se ha producido un error de cualquier naturaleza. El otro fichero es un fichero de extensión *res* (salida.res) que contiene los resultados de la ejecución en el caso de que ésta haya sido positiva, la página ASP recoge los resultados, los interpreta y los muestra devolviendo al usuario una página HTML convencional con los resultados de la ejecución.

A partir de este momento el usuario es capaz de volver a enviar la misma práctica si el resultado no ha sido satisfactorio o probar con la siguiente práctica.

En la figura 4, se muestra el diagrama de ejecución de la aplicación Web en dos fases: a) desde el cliente hasta el servidor, esta parte usa como canal de comunicaciones la red Internet; b) mientras que la comunicación entre el servidor y las placas de los sistemas microcontroladores es local mediante la línea serie.

4. Conclusiones

Creemos que la educación, como todas las áreas, ha de seguir la modernización que este mundo conlleva, y con ella el uso de las nuevas tecnologías. La presencia del alumno en las aulas y en los laboratorios es imprescindible, pero plantear nuevas alternativas no está en confrontación con lo anterior. Se han de complementar y no sustituir los nuevos hábitos educativos. Por este motivo, nos atrevemos a proponer laboratorios remotos, porque seguro que el alumno asistirá en otras asignaturas (o incluso en la misma) a las sesiones que se propongan de carácter presencial.

En este laboratorio se ha propuesto la programación de varios microcomputadores de uso muy general mediante la red Internet y como única herramienta para el cliente un navegador convencional.

No se descarta ampliar este tipo de laboratorio a otras asignaturas donde sea posible la programación de dispositivos no accesibles por parte del alumno de forma presencial.

Referencias

1. Manual de usuario del microcontrolador 8XC196KC/8XC196KD, INTEL Literature.
2. MSDN Library de Microsoft, Tutorial sobre páginas HTML, ASP y JavaScript.
3. Low Power Schottky TTL ICs DataBook, 1st Edition, SGS-THOMSON MicroElectronics.
4. Unified Modeling Language For Real-Time Systems Design v2.0, <http://cs.chungnam.ac.kr/~cmsong/loo/realtime/umlrt/umlrt.htm>
5. "The Indispensable PC Hardware Book", Hans Peter Messmer, Adison Wesley, 1997.
6. "The PC Handbook", Stanley Shell, QED Technical Publishing Group, 1998.
7. "Lenguaje C y estructuras de datos", García de Sola, Mc Graw Hill, 1996.
8. "Circuitos integrados digitales", Muiderkring B. V. Bussum, Ed. Paraninfo, 1995.