

PROBLEMÁTICA DE LA ENSEÑANZA PRÁCTICA DEL DISEÑO DE SISTEMAS BASADOS EN MICROPROCESADOR. ALGUNAS SOLUCIONES.

César Sanz

Universidad Politécnica de Madrid
E.U.I.T. Telecomunicación
Dpto. Sistemas Electrónicos y de Control
Ctra. Valencia Km 7, 28031 Madrid

Tfno: (91) 336 78 06

Fax: (91) 336 78 01

e-mail: cesar@sec.upm.es

RESUMEN

En este trabajo se plantea la problemática que rodea la enseñanza práctica del diseño de sistemas basados en microprocesador en general y el enfoque actualmente empleado en la Escuela Universitaria de Ingeniería Técnica de Telecomunicación de Madrid.

1. INTRODUCCIÓN

La enseñanza práctica del diseño de sistemas basados en microprocesador presenta inconvenientes de tipo económico y técnico, si se pretende que los estudiantes realicen en el laboratorio, el ciclo de diseño completo. Estos inconvenientes se intentan soslayar mediante diferentes estrategias que, en mayor o menor medida, reducen el grado de aproximación al objetivo deseado.

En este artículo se presenta un breve esbozo de la organización de esta docencia en la Escuela Universitaria de Ingeniería Técnica de Telecomunicación de la Universidad Politécnica de Madrid (EUITTM). A continuación, se indican las características, que idealmente, debe reunir un laboratorio de este tipo y la problemática asociada. Posteriormente, se plantean y discuten las soluciones generalmente adoptadas para la materialización de estos laboratorios. Por último, se concluye con la exposición de las soluciones concretas adoptadas en la EUITTM para este laboratorio y la presentación de dos sistemas desarrollados en nuestro departamento que sirven de soporte al mismo.

2. MARCO DE REFERENCIA

Aunque las ideas y reflexiones presentadas en este trabajo, respecto a la organización de las enseñanzas prácticas en el área de los microprocesadores, pretenden ser generales y por tanto, extrapolables a otros Centros, en última instancia se hará referencia al entorno de la EUITTM. Por ello, considero oportuno explicar brevemente cómo se organiza la enseñanza de los microprocesadores en esta Escuela.

Por tratarse de una ingeniería de primer ciclo, la ordenación y secuenciamiento temporal de las materias debe optimizarse al máximo para contrarrestar en lo posible la limitación temporal [1]. En el caso concreto de estas enseñanzas, en nuestra Escuela tradicionalmente se han repartido en dos asignaturas (que son responsabilidad de dos Departamentos diferentes):

a) La primera de ellas, se ocupa de los aspectos relacionados con la arquitectura del microprocesador y del sistema, e introduce los conceptos fundamentales como son, lenguaje ensamblador, programación, interfaces de entrada/salida, interrupciones, acceso directo a memoria, etc. Previamente, los estudiantes han cursado otra asignatura de Programación que les capacita en el paso del problema al algoritmo para resolverlo y la posterior codificación de éste en un lenguaje de alto nivel.

b) La segunda, se centra en la resolución de aplicaciones mediante el empleo de sistemas basados en microprocesador y está más orientada al diseño *hardware*. En ella, se estudia el microprocesador (haciendo énfasis en la temporización) y sus diversos periféricos, vistos como circuitos integrados *discretos*, o integrados en un único *chip* (microcontroladores). Esta asignatura es impartida por nuestro Departamento y la problemática de su enseñanza práctica es la que se aborda en este artículo.

En los nuevos Planes de Estudios de la EUITTM [2] puestos en funcionamiento el curso 1992/93, esta estructura organizativa se ha mantenido para las titulaciones de Ingeniero Técnico en Sistemas Electrónicos e I. T. en Telemática; mientras que, para las de I. T. en Sistemas de Telecomunicación e I. T. en Sonido e Imagen, los contenidos aparecen concentrados, y por tanto algo reducidos, en una única asignatura cuatrimestral.

Por último, conviene resaltar otro parámetro importante que repercute directamente sobre la organización de las enseñanzas prácticas: el número de alumnos. En nuestra Escuela, supera los 200 para la asignatura del Plan 71 y los 300 para las de los Planes 92.

3. PROBLEMÁTICA DE LOS LABORATORIOS DE MICROPROCESADORES

El objetivo principal de un curso sobre microprocesadores, teoría y laboratorio, en el marco de una Ingeniería Técnica, se resume a mi juicio en: *capacitar al estudiante para afrontar y resolver problemas de diseño, aplicando, para ello, soluciones basadas en microprocesador además de la periferia necesaria y el software de control preciso*. No conviene olvidar un segundo objetivo ligado al anterior, como es, *empezar a desarrollar en el alumno el criterio de decisión entre la opción microcomputador u otras posibles, manejando para ello parámetros de ingeniería tales como coste, velocidad de proceso o*

tiempo de desarrollo.

A partir de esta premisa, las líneas maestras de la organización del laboratorio podrían establecerse de la siguiente forma: Primero, dedicar las primeras clases a familiarizar al estudiante con el manejo de la nueva instrumentación (analizador lógico, emulador). A continuación, otro grupo de clases en las cuales se experimente con la programación de los distintos periféricos de la familia del microprocesador elegido (puertos, temporizadores, interrupciones, etc). Y por último, abordar la realización de un diseño final que permita alcanzar el objetivo del laboratorio.

El puesto de un laboratorio de estas características debería estar equipado con: analizador lógico (de no menos de 64 canales), emulador del microprocesador elegido, grabador y borrador de memorias, además de la instrumentación tradicional (PC, osciloscopio digital y fuente de alimentación). La sustitución del emulador por un analizador lógico con sonda de personalización del microprocesador es una opción más económica pero que resta algo de flexibilidad al sistema. En cualquier caso, pensando en un laboratorio con 8 puestos de trabajo, el presupuesto necesario excede la capacidad de nuestro departamento y posiblemente se eleva también por encima de la capacidad media de la Universidad española.

El laboratorio *ideal* propuesto en el párrafo anterior permitiría, teóricamente, cubrir el objetivo deseado: que los estudiantes a lo largo del mismo diseñasen un sistema completo basado en microprocesador para resolver una aplicación concreta. Sin embargo, en la práctica, aún sería necesario resolver algunos otros problemas de tipo técnico relacionados con la realización de los prototipos por parte de los alumnos, el mantenimiento de los equipos y el tiempo disponible para el Laboratorio (número de créditos de la asignatura).

En cuanto al montaje de los prototipos las opciones disponibles son:

a) Placa de circuito impreso: Es el más fiable de todos, aunque plantea los inconvenientes del tiempo de diseño y el coste/disponibilidad de la fabricación (en ocasiones, cuando la complejidad del diseño lo permite, algunos alumnos recurren a la fabricación artesanal de las mismas). La gran ventaja es que redondea los objetivos del laboratorio puesto que el acabado se acerca más a un producto final.

b) Montaje con *wire-wrapping*: Esta técnica es muy flexible y bastante fiable si se ejecuta bien, pero tremendamente insegura si no realiza adecuadamente. Teniendo en cuenta que esta sería la primera experiencia de los estudiantes con dicha técnica, es probable que la balanza se incline hacia el segundo extremo, con lo cual el tiempo dedicado a la depuración del cableado podría restar tiempo al objetivo real del laboratorio, deteriorando así los resultados del mismo.

c) Placa de montaje de inserción: Estas placas resultan muy poco aconsejables para este tipo de realizaciones debido a lo complejo del cableado (buses de alto número de bits), a la frecuencia de las señales manejadas y a la poca fiabilidad de las mismas.

En cuanto a la instrumentación, en el caso del analizador lógico y del emulador, se trata de equipos cuyas sondas de conexión al circuito suelen ser bastante frágiles, a la vez que caras. Por lo que si el número de alumnos que cursan el laboratorio es elevado, y éste se

organiza, durante la realización del diseño final, en régimen abierto con tutorías, la vida media de dichas sondas puede verse notablemente reducida.

Por último, un diseño de estas características puede precisar más tiempo del habitualmente asignado a este laboratorio en los planes de estudios (concepción + montaje + pruebas), lo que provoca que el alumno deba emplear un elevado número de horas adicionales para la realización del mismo. Es por ello que este tipo de diseños quedan a veces relegados al entorno del Proyecto Fin de Carrera, donde el estudiante dispone de una dedicación horaria mayor.

4. POSIBLES ORGANIZACIONES

La realidad de los laboratorios de este área es algo más reducida y habitualmente no se dispone de instrumentación tan sofisticada (por ejemplo, emulador).

La situación lógicamente varía de unos Centros a otros y de unas Universidades a otras. Por ejemplo, es de envidiar (al menos al autor sí le produce esta sana sensación) la situación del *Swiss Federal Institute of Technology* de Lausanne (Suiza), donde la relación estudiante/profesor es de 10 en este tipo de laboratorio, y los alumnos disponen de un servicio de diseño y fabricación de placas de circuito impreso para la realización de los prototipos de sus diseños *hardware* [3]. Por citar algún otro ejemplo menos ventajoso, en nuestra Escuela dicha relación es del orden de 40.

En la práctica, lo habitual en este tipo de laboratorios suele ser utilizar tarjetas, comerciales o diseñadas en el propio departamento, que materializan un sistema microcomputador sencillo, con más o menos periféricos, un bus de expansión al que pueden conectarse otros elementos y un pequeño *software* monitor que permite conectarlos a un *host* para el volcado de programas, depuración, etc. Como *host* suele usarse un PC, sobre el que se realiza la edición, el ensamblado/compilación de los programas de usuario y la depuración de los mismos. Este tipo de placas de evaluación/aprendizaje solían ser desarrolladas inicialmente por el fabricante de microprocesadores para facilitar la introducción de su producto en el mercado (por ejemplo los sistemas SDK-85 o SDK-86 de Intel) [4]. Cumplían el doble objetivo de presentar un sistema abierto diseñado con la CPU en cuestión, y de permitir al diseñador evaluar las posibilidades del microprocesador así como facilitar su aprendizaje. Esta filosofía de sistema con capacidad de ampliación, se ha mantenido en equipos comercializados por compañías, a veces ajenas a los fabricantes de procesadores, pero con una concepción orientada a la docencia de forma más clara. Así, junto con el *hardware*, el fabricante suministra además las herramientas *software* de ayuda al desarrollo citadas anteriormente (ensambladores cruzados, simuladores, depuradores, etc) [5] [6].

Este tipo de tarjetas permite desarrollar dos modelos de laboratorio:

a) *Laboratorios sin diseño hardware*: Es el modelo que más se aleja del objetivo descrito en el apartado 3. En este caso, el *hardware* viene fijado por la tarjeta de evaluación/aprendizaje del microprocesador elegido, a la que pueden conectarse diversos módulos, proporcionados por el propio fabricante de la tarjeta o diseñados en el mismo departamento, para la realización de diferentes prácticas (control de motores, conversión A/D

y D/A, adquisición de datos, generación de sonidos, interfaces a teclado/display y un largo etc). La labor del estudiante consiste en desarrollar el *software* de control de la aplicación e interactuar con el *hardware* disponible dentro del sistema. Como es obvio, la labor de diseño *hardware* queda sustituida, en el mejor de los casos, por la de análisis. Si se dispone de analizador lógico, es posible también la visualización de cronogramas y consideración de cuestiones relativas a la temporización. La ventaja de este tipo de laboratorio es su mantenimiento, que resulta bastante simple. [7] [8]

b) *Laboratorios con diseño hardware*: La infraestructura precisa para este segundo modelo es similar a la del anterior. La diferencia radica en que, en este caso, después de completar algunas prácticas del tipo de las descritas en el apartado a), el estudiante debe realizar un (habitualmente) último diseño que incluye el desarrollo de una circuitería adicional que se conecta a la placa de evaluación y del programa de control de la misma. No cabe duda que esta aproximación continúa reduciendo, en cierta medida, las expectativas de un Laboratorio de esta temática. Sin embargo, orientado adecuadamente, obliga al alumno a tener que adquirir un profundo conocimiento del sistema microcomputador que utiliza aunque no lo construya en su totalidad.

El problema del mantenimiento en este último tipo de laboratorio no resulta en absoluto trivial cuando el número de alumnos es elevado y el personal laboral del departamento asignado al mismo es escaso, o bien, de nivel técnico insuficiente. La experiencia en la EUITTM nos demuestra que el *acceso directo al sistema* por parte de los estudiantes provoca frecuentes averías, siendo, tal como cabría esperar, el número de alumnos y la tasa de mortalidad de las tarjetas directamente proporcionales. Una opción utilizada en algunas Escuelas de Ingeniería [9] en este tipo de situaciones es proteger al sistema mediante optoacopladores, en aquellas líneas a las que los estudiantes tienen acceso. Esta solución restringe la versatilidad del sistema, ya que la programación de determinados elementos (por ejemplo, puertos paralelo que puedan configurarse como entrada o salida), queda fijada por la circuitería de protección, y conseguir que ésta se configure de acuerdo a la programación del dispositivo protegido resulta, en general, costoso.

5. SOLUCIONES ADOPTADAS EN LA EUITTM PARA ESTE LABORATORIO

En este apartado, se presentan dos sistemas desarrollados en el Departamento de Sistemas Electrónicos y de Control de la UPM y que son utilizados en este Laboratorio. El primero, denominado SM86, es lo que podría denominarse un sistema de evaluación/aprendizaje aunque con algunas modificaciones interesantes [10]. El segundo, es una tarjeta de entrada/salida de propósito general, GenIO, conectable al bus del PC, cuya característica más notable es su extremadamente simple mantenimiento y la rapidez de su reparación en caso de avería [11].

Sistema SM86

Se trata de un sistema modular en tarjetas formato *eurocard* simple conectadas en un bastidor a un *back-panel*. El sistema tiene una arquitectura compatible PC, de tal modo que, todos los recursos disponibles en el mismo (puertos, temporizador, controlador de interrupciones, UART, etc), están ubicados en la mismas direcciones que en el PC. Además,

el *software* monitor del sistema es una versión reducida del BIOS del PC, por lo que los mismos ejecutables (*.COM o *.EXE) que corren en el PC, pueden ejecutarse sobre el SM86 (siempre que no utilicen algún recurso no disponible, como por ejemplo, unidades de disco o pantalla). Su CPU es un 8086 en modo máximo y está disponible el zócalo para el coprocesador numérico. El sistema está organizado, en la versión actual, en cuatro tarjetas:

- a) Placa de CPU (8086, 8087, 64 Kbytes de EPROM y lógica asociada al DMA)
- b) Placa de RAM (128 Kbytes de SRAM y puertos programables 8255)
- c) Placa de periféricos (controlador de interrupciones 8259, temporizador 8254, UART 8250 e interfaz RS-232C)
- d) Fuente de alimentación (5 V, 1.5 A)

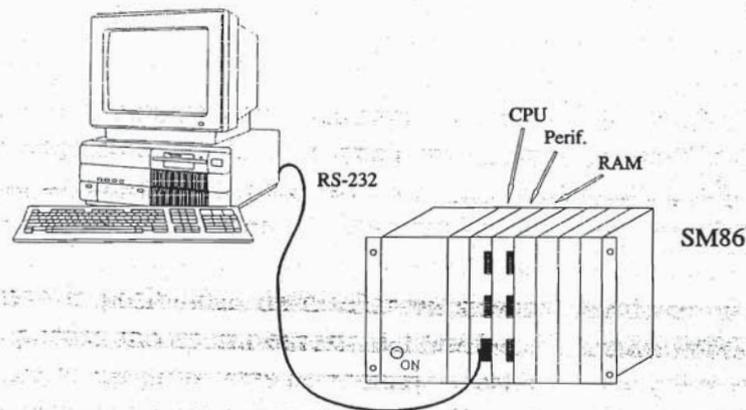


Figura 1. Sistema SM86

El sistema se completa con un entorno *software*, desarrollado sobre MS Windows 3.1, que permite la comunicación con el SM86 para el volcado de programas ejecutables, visualización/modificación de memoria y registros en el SM86, ejecución del programa (paso a paso, continuo o con puntos de ruptura).

Este sistema permite:

a) Visualizar los buses del sistema, con la ayuda de un analizador lógico, para observar situaciones tales como: respuesta a un RESET, introducción de estados de espera en un ciclo de máquina, ciclo de reconocimiento de una interrupción, búsqueda anticipada de instrucciones, protocolo de petición/concesión de buses entre el 8086 y el 8087, etc.

b) Ejecución controlada de programas editados y compilados en el PC.

c) Inserción de nuevas tarjetas en el sistema (en el bastidor quedan 3 *slots* disponibles). En este caso, nos inclinamos por el modelo a) del apartado 4; es decir, tarjetas ya diseñadas en el Departamento y no por los alumnos que tan sólo realizan el *software* necesario para controlarlas (p.e. un voltímetro con un convertidor A/D y presentación en cuatro dígitos multiplexados). La razón de esta decisión es lo costoso que resulta el mantenimiento de este sistema, lo que constituye el principal inconveniente para su uso en *Laboratorios con diseño hardware*.

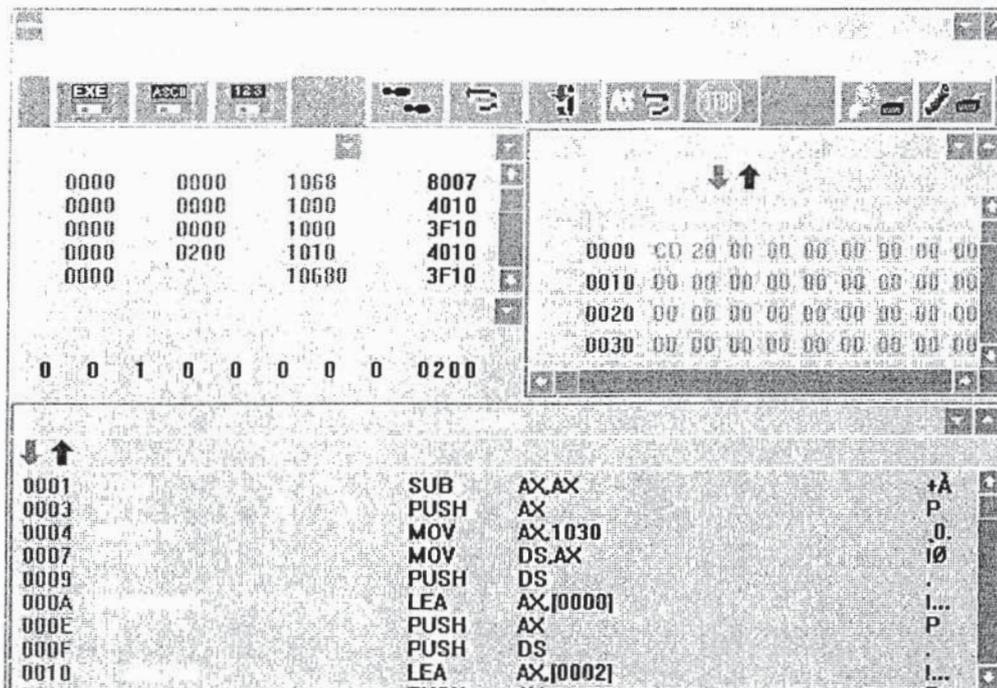


Figura 2. Entorno *software* del SM86

Tarjeta GenIO

Se trata de una tarjeta conectable al bus del PC, que dispone de un conjunto de periféricos (dos 8255 y un 8254) y proporciona acceso a recursos de la placa base del ordenador (dos interrupciones, IRQ2 e IRQ3 y al canal 1 de DMA). En este caso, el propio PC se convierte en el sistema microcomputador (placa de evaluación) objeto de estudio y utilización por parte de los alumnos. Con esta tarjeta, simplemente se le dota de una periferia adicional y versátil.

La tarjeta dispone de dos conectores de cable plano de 40 pines, a través de los cuales el alumno puede acceder a todos los recursos de la misma: puertos, temporizador, interrupciones y señales relacionadas con el DMA. Existe un tercer conector en la placa a través del cual se puede acceder a la parte baja del bus de datos del PC.

Esta tarjeta permite:

- Experimentar con la programación de los periféricos disponibles (8255 y 8254) en sus diversos modos de funcionamiento.
- Realizar prácticas relacionadas con los métodos de entrada/salida de datos: por programa, por interrupción (además de las dos disponibles, se puede usar también la int 1Ch asociada a la IRQ0 del PC), o por DMA (pueden realizarse transferencias entre memoria y uno de los puertos o viceversa, a través del canal 1 de DMA).
- Finalmente, abordar la realización de un diseño final que permite aproximarse, en cierta medida, al objetivo del laboratorio descrito en el apartado 3.

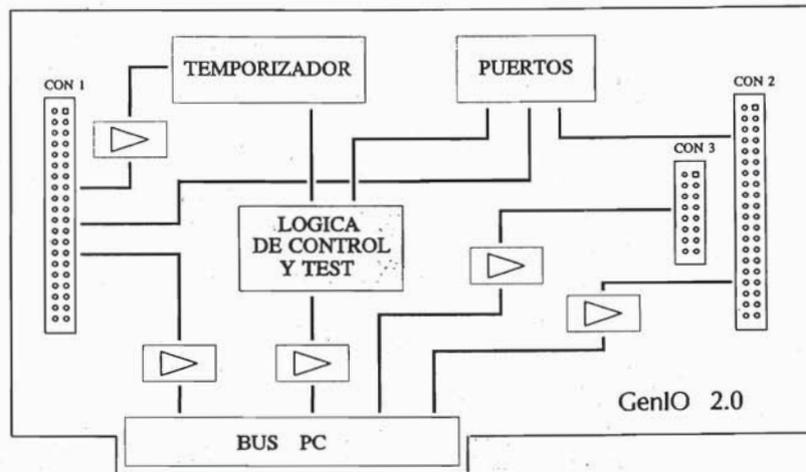


Figura 3. Tarjeta GenIO

Existen tarjetas comerciales de este tipo, algunas de las cuales hemos utilizado en cursos anteriores [12] [13]. Pero, en general, no son tan versátiles como ésta. Algunas no proporcionaban entradas de interrupción y menos aún, soporte para un canal de DMA, y además, nos plantearon graves problemas de mantenimiento durante la realización del diseño final.

Con la experiencia adquirida en el uso de estas tarjetas comerciales, nos planteamos el diseño de una propia que solventase el problema del mantenimiento. En principio, se evaluó la protección (*buffers* u optoacopladores) de los circuitos a los que iban a tener acceso los estudiantes: puertos y temporizador. Sin embargo, para el 8255 (puertos), la pérdida de flexibilidad en su uso es excesiva, e intentar reducir esa pérdida conduce a que el coste de la protección excede con mucho al coste del circuito protegido, por lo que no resulta rentable. Por ello, la protección se limita a colocar *buffers* sólo en aquellas señales cuyo sentido no es programable (entradas de reloj del 8254, salidas del mismo, entradas de interrupción, etc); mientras que, en el caso de los puertos (8255), no existe protección alguna.

Una vez desestimada la protección total del sistema, el esfuerzo de diseño se volcó en hacer que el proceso de identificación de la avería y su reparación fuese lo más simple y rápido posible.

La tarjeta GenIO está diseñada de tal modo que, cerrando entre sí los dos conectores DB25 (uno macho y otro hembra) en que se terminan los cables planos, y ejecutando un programa de auto-test, se identifican automáticamente los circuitos integrados averiados. Con ello, la labor de mantenimiento (que no requiere una especial cualificación) se reduce a ejecutar el programa de test, anotar la referencia del circuito averiado, apagar el ordenador, sustituir el integrado averiado en la tarjeta y, por último, volver a ejecutar el programa de test para verificar el proceso. El tiempo empleado no sobrepasa los cinco minutos.

Otra ventaja de esta tarjeta es su bajo coste, y la flexibilidad que ofrece la propiedad del diseño a la hora de hacer modificaciones y mejoras sobre la misma.

Además de las importantes ventajas indicadas, la utilización durante varios cursos de esta tarjeta y su programa de test, ha revelado una ventaja adicional no menos interesante como es proporcionar un clima de total confianza en el funcionamiento de la tarjeta. El alumno al iniciar su clase de laboratorio ejecuta él mismo el programa de test de la tarjeta y se asegura que se encuentra correctamente antes de conectar su *hardware* y comenzar a depurar su programa. De este modo, la típica frase "*¡Mi programa y mi circuito están bien! ¡Es la tarjeta, la que no funciona!*" ha dejado de escucharse.

6. CONCLUSIONES

En este artículo se ha analizado la problemática de los laboratorios de diseño de sistemas basados en microprocesador, estableciendo una clasificación de las diferentes alternativas para su realización y la influencia sobre los objetivos alcanzables. Se han presentado dos sistemas reales, diseñados en el departamento de Sistemas Electrónicos y de Control de la EUITTM, con cuya utilización pueden alcanzarse los objetivos docentes deseados. En concreto, la tarjeta GenIO permite que los alumnos realicen diseños completos (*hardware + software*) con un impacto mínimo sobre el funcionamiento del laboratorio (mantenimiento), aunque el número de estos sea elevado.

Actualmente, estamos valorando la posibilidad de diseñar un sistema basado en un microcontrolador, al que los alumnos puedan incorporar el resto del *hardware* necesario para la aplicación. Esta opción reduce en parte los problemas de montaje y cableado enunciados en el apartado 3 y, debido a la existencia de herramientas de desarrollo *software* sobre PC a un coste aceptable, se presenta como una alternativa muy interesante para soportar la docencia en estos laboratorios, si bien el problema del diagnóstico y la reparación de averías de forma eficiente está aún por resolver.

7. AGRADECIMIENTOS

El diseño físico y fabricación del sistema SM86 se ha realizado dentro de un proyecto de Investigación de *Acciones Concertadas de la U.P.M.* concedido a nuestro Departamento [14]. El autor desea agradecer a Pedro Cobos y Matías J. Garrido, compañeros del Departamento las múltiples y enriquecedoras discusiones sobre este tema, así como la ayuda en la revisión de este trabajo. Por último, mi agradecimiento a toda la gente, profesores del Departamento y alumnos de Trabajo Fin de Carrera, que han colaborado en la realización de los sistemas SM86 y GenIO, cuyo número es demasiado alto para poder citarlos a todos.

8. BIBLIOGRAFÍA

- [1] J. Arriaga, C. Sanz, "The Three-Year Engineering Curriculum: A Difficult Balance", *Proceedings of the 23rd IEEE Frontiers in Education Conference*. Washington, DC, pp 297-300. Noviembre, 1993.
- [2] Resolución 26886 de 1 de Octubre de 1992 de la Universidad Politécnica de Madrid

publicada en el B.O.E. de 2 de Diciembre de 1992. pp 40942-40960

- [3] J.D. Nicaud, "Dedicated Tools for Microprocessor Education", *IEEE Micro*, Vol 11, nº 1, pp 14-17, 62-68. Febrero, 1991.
- [4] SDK-85 System Design Kit. User's Manual. Intel.
- [5] "TM-683. Sistema Microinstructor para el Microprocesador 68000. Manual Técnico", Departamento de Ingeniería Electrónica (UPC) e Instrumentación Electrónica PROMAX S.A., Septiembre, 1990.
- [6] R.W. Floyd, "The MC68000 Educational Computer Board", *Byte*, Vol 8, nº 10, pp 324-336. Octubre, 1983.
- [7] J.A. Fulcher, "Fun and Games and Microcomputer Interfacing", *IEEE Micro*, Vol 11, nº 1, pp 18-21, 75-78. Febrero, 1991.
- [8] T.W. Schultz, "Peripheral Hardware and a Hands-On Multitasking Lab", *IEEE Micro*, Vol 11, nº 1, pp 30-33, 80-82. Febrero, 1991.
- [9] E.I. Boemo, J. Meneses, G. González, F. Barbero, "FPGAs in Education: A Case Study", *3rd Workshop on FPGAs and Applications*, Oxford, U.K. Septiembre, 1993.
- [10] C. Sanz, M.J. Garrido, S. López, "SM86. Manual de Usuario", Dpto. Sistemas Electrónicos y de Control, E.U.I.T. de Telecomunicación de Madrid, U.P.M.
- [11] C. Sanz, M.J. Garrido, "Tarjeta GenIO. Manual de Usuario", Dpto. Sistemas Electrónicos y de Control, E.U.I.T. de Telecomunicación de Madrid, U.P.M.
- [12] "PIO-12 User's Manual", KEITHLEY, U.S.A.
- [13] "SI-8255 Digital I/O and Counter Card. User Manual", Singular Technology Corp. Taiwan R.O.C.
- [14] M.J. Garrido y otros, "Sistema Microcomputador modular compatible con la arquitectura PC para aplicaciones industriales y didácticas. Memoria Final", Acciones Concertadas 1991, Universidad Politécnica de Madrid.