Enfoque práctico de la informática en un entorno industrial

Pau Micó[1] y Javier Silvestre[2]

Dept. Informàtica de Sistemes i Computadors. Escola Politècnica Superior d´Alcoi (U.P.V.). Alcoi. Spain. E-mail: [1] pabmitor@disca.upv.es, [2] jsilves@disca.upv.es.

Resumen: Mediante este trabajo se pretende difundir el enfoque práctico que, desde la Escuela Politécnica Superior de Alcoi (EPSA), se le está dando a la asignatura de Informática industrial. Debido a las especiales características del entorno socio-económico de la comarca en la que se enclava la EPSA, la oferta de una informática aplicada al campo industrial supone un importante aliciente para el alumnado. Afortunadamente, en una asignatura cuya base es el aprendizaje de un lenguaje de programación nunca faltan los buenos ejemplos y, tal y como se pretende mostrar a continuación, la preparación de unas prácticas de laboratorio mediante prototipos electro-mecánicos (lo más parecidos posible a lo que el alumno se encontrará en el ámbito industrial que le rodea) han resultado en una formación más equilibrada, en un interés creciente por la asignatura y, debido a la motivación generada por las prácticas, en una demanda de conocimientos extraordinaria. Todo ello se intentará sintetizar a través de una presentación de los objetivos pretendidos, un detalle del plan docente aplicado, un desarrollo y descripción del trabajo realizado en el laboratorio, y un apartado final de conclusiones.

Palabras clave: Informática industrial, aprendizaje práctico.

Title: Informatics practical approach in an industrial environment

Abstract: From the School of Alcoi (EPSA) and by means of this work, we want to contribute presenting the practical approach given to the subject of Industrial Informatics. Due to the special socio-economic background of the district in which the EPSA is located, the learning of a computer science-based subject oriented towards the industrial environment is a major incentive for the students. Fortunately, there are many good examples of industrial applications, that allow to learn a programming language and, as we demonstrate next, the preparation of some laboratory practices with electro-mechanical prototypes (similar to the real devices that the student will deal with in a furture), have resulted in a more balanced formation, a growing interest in the subject and finally, in an special demand for knowledge. The paper is organized as follows: first, the goals of the practical learning are presented, secondly, the teaching schedule is detailed, next, we develop and describe the preparation of the laboratory and, finally, a section of conclusions is discussed.

Key words: Industrial informatics, practical learning.

Objetivos

En primer lugar, hay que hablar de la formación académica. Evidentemente, en el ánimo de los profesores que han participado en el proyecto se encuentra

como principal motivación la de mejorar la formación académica de los alumnos que cursan la asignatura de Informática industrial I en la EPSA. Esta mejora se ha pretendido a través de una innovación en las enseñanzas prácticas. En este caso se ha intentado no sólo el reforzar los conocimientos teóricos adquiridos en seminarios y presentaciones (tal y como venía siendo tradicional), sino que, además, se ha pretendido ofrecer una visión de futuro al ingeniero en ciernes que, en poco tiempo, deberá valerse por sí mismo en un mundo bastante alejado de la realidad académica. Se ha intentado aproximar el ámbito docente en el que el alumno se está formando, a la realidad industrial que habitualmente le rodea.

El objetivo general de la mejora propuesta consiste en una reestructuración global de la asignatura, aumentando su cariz experimental y ofreciendo a los alumnos la posibilidad de aplicar los conocimientos adquiridos a situaciones reales. De esta forma, además de reforzar los conceptos teóricos adquiridos, se utilizará un nuevo método práctico desarrollado por los profesores responsables de la asignatura cuya principal característica es la de reflejar una realidad industrial que incremente la motivación del alumno por la asignatura.

De esta forma los alumnos asimilarán mejor los contenidos desarrollados en clases de teoría, ya que actualmente, debido a las limitaciones en cuanto al material y a las necesidades intrínsecas de las asignaturas relacionadas, las actividades que se aplican no son suficientes, ya que están formadas mayoritariamente por simulaciones o enfocadas a tratar elementos individuales de un sistema de control, perdiendo de vista el desarrollo global.

Descripción del trabajo

El conocimiento por parte del profesorado de la realidad industrial de la comarca en la que, como ya se ha comentado, se encuentra inmersa la EPSA, hizo desde un principio que la selección de las prácticas de laboratorio fuese una tarea ardua y harto complicada. Finalmente se decidió dividir el conjunto en dos bloques claramente diferenciados. El primero de ellos en los que el alumno aprovecharía para recordar y reforzar conceptos teóricos al respecto del lenguaje de programación utilizado (en nuestro caso, lenguaje C) (Cuesta, Sastre; 1999) (Zimmerman y Zimmerman, 1990). En el segundo bloque (que es el que resulta de interés en nuestro caso), se planteó la posibilidad de desarrollar una serie de prototipos electro-mecánicos (Rosh, 1992) a través de los cuales el alumno pudiera intuir (si no ver claramente) el ambiente industrial en el que, muy probablemente, tendía que interactuar en breve. Al primer bloque se le denomina *bloque de programación* y al segundo bloque se le llama *bloque industrial*. Por cuestiones didácticas y prácticas, el entorno de programación se desarrolla en Windows 98 (que sí deja acceder a los recursos hardware a bajo nivel y permite todo el tratamiento de interrupciones), aunque la programación en C se implementa mediante TURBO-C bajo MS-DOS.

La preparación de las prácticas englobadas en el *bloque industrial* resultó laboriosa y el diseño y fabricación de los diferentes prototipos que el alumno debería aprender a programar y controlar nos llevó todo un curso académico.

Plan docente

La asignatura de *Informática industrial I* para la que se preparan las prácticas mencionadas, es una asignatura cuatrimestral con una carga docente de 3,5

créditos de teoría y 2 de prácticas. Las 20 horas de prácticas se imparten a un único grupo durante 10 semanas a razón de 2 horas por sesión (sincronizando los contenidos teóricos con el progresivo desarrollo de las sesiones prácticas). Además, hay que tener en cuenta que no todos los alumnos implementan todas las prácticas presentadas aquí. La idea es que se adquieran los conocimientos básicos a partir de unas prácticas comunes y que, a continuación, sean ellos mismos los que elijan una práctica de aplicación real de entre las propuestas para ello, en la presentación de un trabajo práctico final.

En la parte de prácticas comunes y a razón de una práctica por sesión (7 en total de las cuales las dos primeras pertenecen al *bloque de programación* y el resto son del *bloque industrial*), se implementan: (p1) Introducción a la programación en C (dedicada a la implementación de estructuras de selección, repetición, control, gestión de distintos tipos de datos, etc.); (p2) Introducción a la programación en C (donde se introduce en manejo de ficheros y la gestión del entorno en modo gráfico); (p3) Medida del tiempo en el PC; (p4) Manejo de interrupciones; (p5) Gestión del puerto paralelo (donde el alumno cuenta con dos tipos de maqueta: la del control de motores o la del control de iluminación); (p6) Comunicación a través del puerto serie y (p7) Programación de tarjetas de adquisición de datos.

En el bloque de prácticas de aplicación real, para la que se proponen las 3 sesiones restantes (p8, p9 y p10), el alumno puede optar por la implementación de uno de los siguientes proyectos: (i) Simulador de control de acceso a un edificio; (ii) Diseño de un sistema de control semafórico; (iii) Simulador de un brazo robot; (iv) Simulador de un control de ascensor o (v) Simulador del control de una vivienda.

A continuación se citan algunas de las prácticas contenidas en el *bloque industrial* incluyendo una breve reseña descriptiva de los prototipos desarrollados.

Medida del tiempo

Una de las tareas más importantes a desarrollar en un sistema de control por computador es la medida, más o menos exacta, dependiendo de la aplicación, del tiempo. En esta práctica se pretende que el alumno tome contacto con distintas posibilidades de medida del tiempo usando un PC y el Lenguaje C.

Las técnicas de medida del tiempo presentadas son dos: la programación a bajo nivel del chip contador/temporizador PIT8254, y la programación mediante funciones de librería del reloj de tiempo real del sistema. Ambas técnicas se complementan y sirven de introducción para las prácticas aplicadas sobre prototipos de laboratorio ya que, por una parte, la programación a bajo nivel del PIT8254 ayuda al alumno a entender el concepto de programación sobre los puertos de un circuito integrado en su propio PC y, por otra, la utilización de funciones de librería sobre el reloj de tiempo real le proporciona una importante herramienta de temporización de muy fácil manejo y que les resultará de evidente utilidad para su aplicación en la programación de algunos de los prototipos presentados en prácticas posteriores (por ejemplo, los de control semafórico).

Por otro lado y además, tras la realización de esta práctica también se pretende la aplicación de los conceptos adquiridos a los protocolos necesarios

para la comunicación a través de los puertos paralelo y serie del PC. Se propone programar el envío a la impresora de cierto fichero de texto a través del puerto paralelo, para lo que se utilizarán las rutinas de medida y gestión del temporizador del PC para implementar adecuadamente el protocolo de comunicaciones.

Manejo de interrupciones

En esta práctica el alumno aprende a gestionar interrupciones de tipo hardware. Para ello se ha diseñado un prototipo electrónico que se conecta al PC a través del puerto paralelo y que consiste en una serie de interruptores y pulsadores a través de los cuales se genera la interrupción. Esta práctica resulta especialmente interesante por la clara aplicación práctica que este tipo de gestión supone en un ambiente industrial. En concreto, el alumno deberá entender el circuito propuesto para la interrupción (Figura 1), así como programar el puerto paralelo para que sea capaz de detectar una interrupción hardware a través de alguna de sus entradas, además de aprender a programar los Controladores de Interrupciones Programables (PICs) del PC para habilitar las interrupciones hardware a través del puerto paralelo.

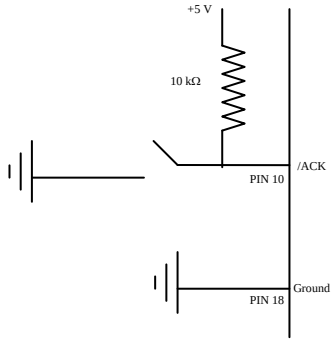


Figura 1. Interruptor para el puerto paralelo.

Puerto paralelo (i): sistema de control de un motor eléctrico

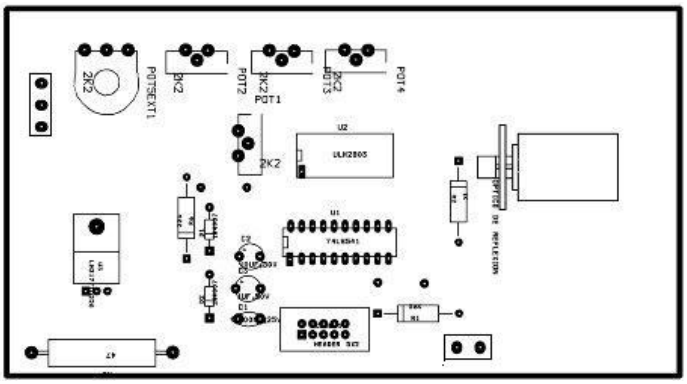


Figura 2. Sistema de control de un motor eléctrico.

Una vez desarrollado y presentado el prototipo compuesto por un motor eléctrico (Figura 2) y que se comunica con el PC a través del puerto paralelo, el alumno deberá utilizar el lenguaje de programación correspondiente (en nuestro

caso C) para, programando el puerto paralelo, ser capaz de controlar la velocidad de rotación del motor (mediante un menú desde donde el usuario pueda seleccionar distintas velocidades de rotación, e incluso programar manualmente una velocidad de rotación concreta), además de mostrar por pantalla la velocidad a la que el motor gira en cada momento.

Puerto paralelo (ii): sistema de control para iluminación

En este caso, las tareas a llevar a cabo en el laboratorio consistirán en la construcción de un hipotético sistema de control para el prototipo del panel de luces presentado en la Figura 3 (izquierda). Para ello se establecerán unas entradas, dadas por el estado de ciertos indicadores del panel, y unas salidas (otra serie de indicadores). El alumno deberá diseñar e implementar el correspondiente protocolo para el envío y recepción a y desde el prototipo, de las señales digitales adecuadas.

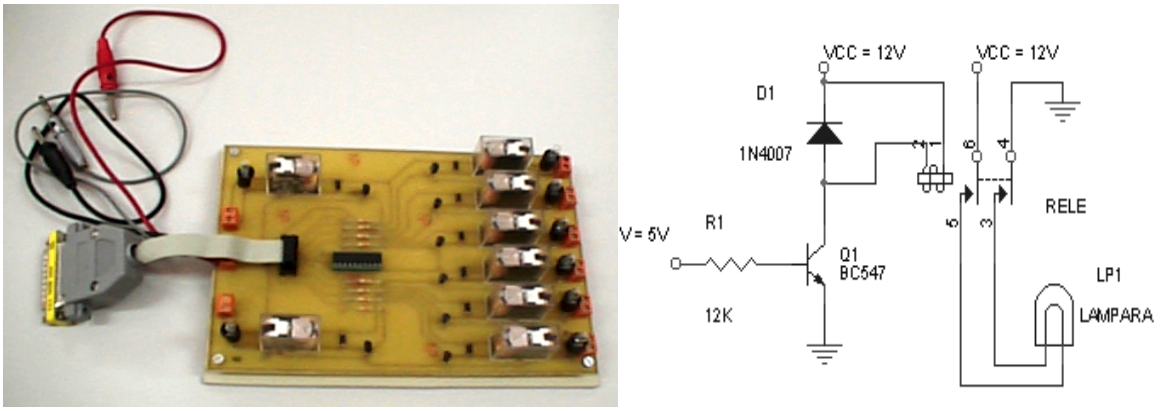


Figura 3. (Izquierda) Sistema de control de iluminación y (derecha) red de polarización.

La aplicación implementada deberá estar compuesta por un menú a través del cual el usuario sea capaz de encender o apagar las luces seleccionadas (presentando un mensaje de error en el caso en el que se trate de encender una luz ya encendida o apagar una que ya lo esté), encender todas las luces al mismo tiempo, apagar todas las luces al mismo tiempo y programar el prototipo mediante un encendido rotativo y temporizado (similar al de los anuncios luminosos). Todo esto a través del bus paralelo.

La red de polarización utilizada para la alimentación de las bombillas, y dado que la tensión que se recibe en el prototipo a través del puerto paralelo resulta insuficiente para su encendido, es la que se presenta en la Figura 3 (derecha). Además, mediante este montaje se aísla el puerto paralelo del resto del circuito de manera que queda protegido de posibles picos y subidas de tensión indeseados.

Comunicaciones a través del puerto serie

Aunque, en este caso, se trata de un puerto que se encuentra en desuso hay que tener en cuenta que todavía aparece como puerto de comunicaciones en mucha de la maquinaria industrial existente en las empresas de la zona (de la industria textil podemos citar, por ejemplo, los telares, tricotosas, urdidoras, flocadoras, estampadoras, etc.). Por este motivo se incluyó esta práctica en la

que el alumno debe programar un protocolo de comunicaciones a través del puerto serie, poniendo en contacto dos PCs y pudiendo realizar una comunicación entre ellos de tipo full-dúplex. Para ello deberá estudiar y entender la programación a bajo nivel de la UART (Universal Asynchronous Receiver & Transmitter) de su PC. Al finalizar la práctica el programa desarrollado debe permitir la comunicación (a través del puerto serie) entre los usuarios de dos PCs. El programa desarrollado deberá permitir a los usuarios el utilizar un menú de configuración de la UART (para cambiar tanto la velocidad de transmisión y recepción como la longitud de las palabras enviadas y la configuración de los protocolos. Además, el programa deberá presentar por pantalla el correspondiente mensaje de error, si las configuraciones de las UARTs de transmisión y recepción difieren en alguno de sus parámetros. Finalmente, el usuario podrá utilizar el programa a modo de chat mediante el menú correspondiente.

Por otra parte, los alumnos deberán diseñar las conexiones del puerto serie que permitan la comunicación entre los dos terminales en modo full-dúplex, tal y como se muestra en la Figura 4 (incluyendo los dos tipos de conector del puerto serie, el de 9 y el de 25 pines, con su correspondiente numeración y su correspondencia con las señales estándar para llevar a cabo el protocolo de comunicación full-dúplex a implementar por los alumnos).

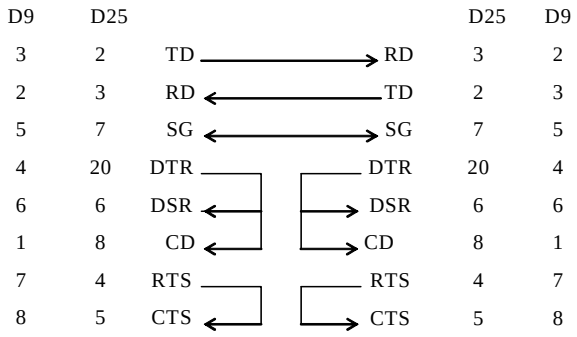


Figura 4. Conexionado full-dúplex del puerto serie.

Programación de tarjetas de adquisición de datos

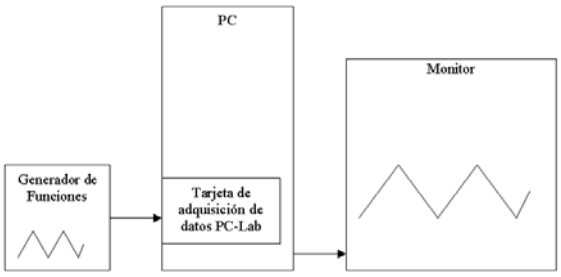


Figura 5. Montaje utilizado para la tarjeta de adquisición de datos.

Al igual que se ha hecho con los puertos, en esta práctica el alumno aprenderá a obtener y enviar información al exterior, usando en este caso una tarjeta de adquisición de datos comercial. La tarjeta conectada a los PCs del laboratorio es la PCL-812PG (también denominada PC-Lab) (Varios, 1990). En este caso se

pretende que el alumno aprenda a programar y controlar una tarjeta de adquisición de datos conectada a su PC para lo cual deberá implementar un osciloscopio con ella de la siguiente forma (Figura 5): se conecta un generador de funciones a la entrada analógica de la PC-Lab y se desarrolla el programa adecuado para la configuración de la tarjeta, captura de las muestras de entrada y presentación de la señal por el monitor del PC. La tarjeta muestreadora permite diferentes configuraciones (trigger por software, frecuencia de muestreo fija, etc.) por lo cual el alumno deberá implementar también (en el programa anterior) un menú con las diferentes opciones de configuración de la tarjeta.

Simulador de control de acceso a un edificio

Para el que se ha desarrollado una tarjeta de expansión que interacciona con los buses tanto paralelo como serie, a través de los cuales se conecta el sistema de control de acceso al edificio (Figura 6). El sistema de control está compuesto por un motor que actúa sobre los goznes y está encargado de abrir la puerta de acceso y por un sistema de bobinas y un segundo motor que acciona el cerrojo para abrir o cerrar la puerta. El trabajo del alumno consiste en controlar todo el sistema para accionar los mecanismos de apertura y control dependiendo de un código que el usuario introduce por el teclado del PC.

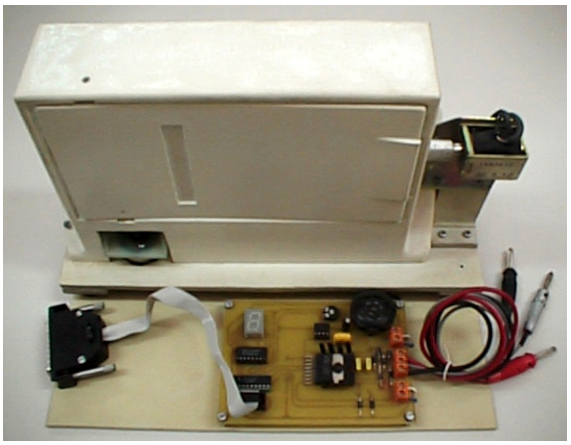


Figura 6. Control de acceso a un edificio.

Diseño de un sistema de control semafórico

En este caso el alumno deberá realizar un programa para controlar el sistema de semáforos mostrado en el prototipo de la Figura 7. La comunicación se realizará a través de tarjetas de expansión para los buses serie y paralelo, que han sido implementadas especialmente para manipular los prototipos de forma genérica. Para controlar el sistema de semáforos, el alumno tendrá que identificar inicialmente las líneas del puerto seleccionado y que le permitan establecer un protocolo de comunicaciones con el prototipo, además de identificar también las líneas de datos por las que enviar la información de encendido-apagado de las luces de cada uno de los semáforos. En cuanto al software se refiere, el alumno analizará la problemática asociada al control de los semáforos en un cruce múltiple, por lo que, antes de empezar a programar en código C, tiene que haber resuelto el problema planteado mediante diagramas de bloques y pseudo-código.

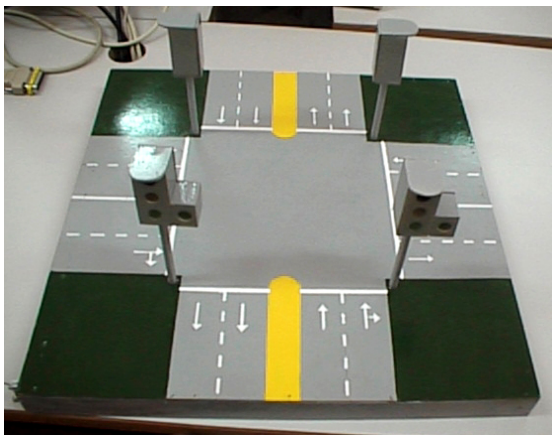


Figura 7. Sistema de control semafórico.

Simulador de un brazo robot

En este caso el alumno deberá controlar un brazo robot a través del bus del PC (Figura 8). La comunicación se realizará a través de tarjetas de expansión para los buses serie y paralelo, que han sido implementadas especialmente para manipular los prototipos de forma genérica. El motivo de esta práctica será tanto el de programar el microcontrolador del brazo como el de implementar el protocolo de comunicaciones adecuado para controlar su movimiento. El brazo robot deberá poder moverse tanto en horizontal como en vertical así como el realizar tareas sencillas de posicionamiento e interacción con pequeñas piezas geométricas (coger y depositar elementos sobre determinadas ubicaciones). La implementación de este sistema implica que el alumno deba conocer las características del microcontrolador del brazo robot así como el pensar en un protocolo compatible entre el PC y el robot. Por otra parte, toda la documentación técnica facilitada a los alumnos estará en inglés, con lo que se les obliga y conciencia de la importancia del conocimiento de este idioma.

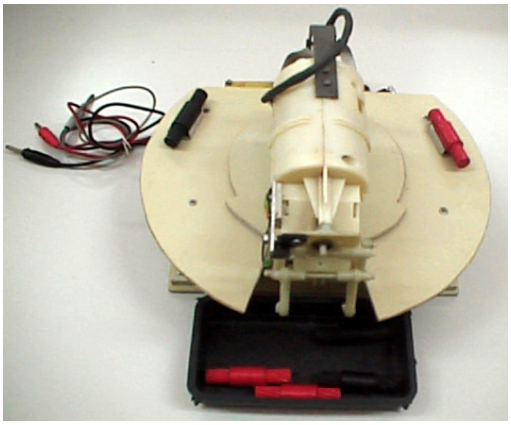


Figura 8. Simulador de un brazo robot.

Simulador de un control de ascensor

Al igual que sucede en los prototipos presentados anteriormente, en este caso el alumno aprenderá a controlar un ascensor (prototipo mostrado en la Figura 9), utilizando para ello los puertos paralelo y serie del PC. Para ello el usuario tiene

que interactuar a través del PC con un panel de control similar al que se encuentra en un ascensor convencional. De esta manera el alumno deberá pensar en toda la problemática asociada a la programación del ascensor. Se plantea resolver el problema para un ascensor sin memoria (problema de control del dispositivo hardware del ascensor) para, posteriormente, hacerlo en el caso de un ascensor con memoria (problema hardware + el software requerido para hacer el control de la memoria).

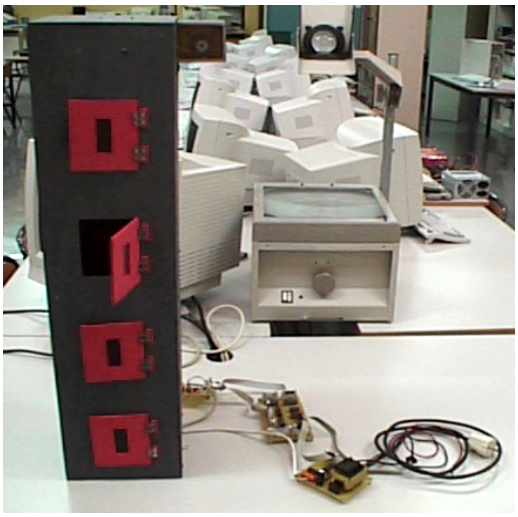


Figura 9. Simulador de un control de ascensor.

Simulador del control de una vivienda



Figura 10. Simulador de un control de una vivienda.

En este caso, aplicaremos los algoritmos de control sobre el prototipo de una vivienda de varias plantas interactuando, no sólo sobre el ascensor, sino también sobre su sistema de iluminación. Este sería un caso claro de aplicación domótica sobre proyectos reales. De esta manera el alumno tendrá que analizar globalmente el problema, no sólo a nivel hardware (tarjetas de expansión de los puertos paralelo y serie para la comunicación con el PC), sino también a nivel

software (para diseñar un panel de control accesible al usuario final), así como la necesidad de plantear y diseñar todas aquellas canalizaciones por las que han de discurrir las líneas eléctricas que posibiliten la implementación del sistema en una vivienda real. Normalmente esta práctica se desarrolla implicando a varios equipos de alumnos, que son coordinados por uno de ellos. Con todo ello se pretende simular de la manera más real posible una de las situaciones que se podría dar en su vida profesional: la interacción y la coordinación con varios equipos de trabajo. Tanto el proyecto final, como el planteamiento y presentación de los problemas planteados durante la práctica así como las soluciones por las que ha optado para resolverlos, son objeto de puesta en común para debate entre los diferentes grupos de trabajo. Así, además de concienciar a los alumnos de la importancia y dificultades intrínsecas del trabajo en equipo, conseguimos comunicarles lo que les depara su futuro inmediato como ingenieros, planteándoles una posible línea de trabajo como lo es la de la realización de proyectos de ingeniería (en este caso la domótica).

Conclusiones

El desarrollo de las prácticas presentadas nos ha permitido certificar la mejora del nivel académico del alumnado que supera el nivel mínimo del establecido en años anteriores para el aprobado de la asignatura.

Además, el amplio rango de prototipos implementados permite que los alumnos sean capaces de ver un gran número de aplicaciones reales diferentes de las que, de otra manera, no serían conscientes. Su motivación aumenta en la medida en la que van siendo capaces de realizar y resolver los distintos problemas de informática industrial planteados en las prácticas.

Por otra parte, su nivel de satisfacción también se ha visto acrecentado simplemente por el hecho de que, sin un gran esfuerzo de imaginación, son capaces de extrapolar al mundo real la gran mayoría de las aplicaciones prácticas desarrolladas sobre los prototipos. Además, el mero hecho de interactuar con elementos electro-mecánicos reales a través de su operación y manipulación, incrementa su seguridad y autoestima a la hora de desarrollar futuros trabajos para el mundo industrial.

Otra de las ventajas que se desprenden de trabajar con los prototipos implementados es que, debido a su diseño versátil, permiten un fácil desarrollo y aplicación de multitud de nuevas prácticas y conceptos que puedan resultar de interés docente en un futuro próximo.

Como conclusión final cabe destacar que el método empleado ha supuesto una inversión rentable y de innegable interés tanto en capital humano (mediante el desarrollo y afianzamiento de unas aptitudes curriculares que, de otra forma, quedaban desligadas de la realidad) como en capital docente (por las dosis de motivación extraordinaria que este método ha supuesto tanto para profesores como para alumnos).

Referencias bibliográficas

An, P. (1998). *P C Interfacing*. Newnes.

Campelo, J. C., Rodríguez, F., Torres, V. (1997). *Periféricos e interfaces industriales*. Valencia: SPUPV.

Charte, F. (1997). *Programación con C++ Builder*. Ediciones Anaya Multimedia, S.A.

Cuesta, D. y Sastre, C. (1999). *El PC: Estructura y Programación*. Valencia: SPUPV.

Gil, J. A. (1999). *Estructura de computadores, Volumen 2*. Valencia: SPUPV.

Leblanc, G. (1997). *Borland C ++ Builder*. Ediciones Gestión 200, S.A.

Rosh, W. (1992). *La biblia del Hardware*. Ed. Jackson.

Varios, (1990). *P C-Lab User's manual*. Advantech Co.

Zimmerman, S. y Zimmerman, B. (1990). *La biblia del Turbo C*. Anaya Multimedia.