

iFOTON. Herramienta didáctica de bajo coste para el desarrollo de sistemas basados en microcontrolador

Norberto Cañas de Paz
Dpto. de Informática Aplicada
Universidad Politécnica de Madrid
28031 Madrid
e-mail: norberto@eu1.upm.es

Gracián Triviño Barros
Dpto. Tecnología Fotónica
Universidad Politécnica de Madrid
28660 Madrid
e-mail gtrivino@fi.upm.es:

Resumen

Con este documento presentamos un entorno de desarrollo, con interés didáctico, para sistemas con microcontroladores (que hemos llamado iFOTON). Ideado y construido para preparar prototipos con el microcontrolador PIC16F873, está constituido por dos componentes fundamentales: El primero es una tarjeta PCB de propósito general y de bajo coste, en la que se ha incluido una zona pretaladrada para incluir nuevos elementos. El segundo es un programa que permite grabar la memoria no volátil del microcontrolador, sin utilizar un programador hardware externo, valiéndose para ello del puerto paralelo (IEEE-1284) del computador en el que se ejecuta. Dicho programa suministra también un conjunto de utilidades básicas para la organización de proyectos sencillos.

1. Antecedentes

Existen alternativas comerciales muy variadas para el desarrollo de sistemas con microcontroladores, muchas de las cuales tienen distintos grados de interés para la enseñanza en razón de su coste, complejidad, orientación del producto, etc. En este sentido es interesante indicar que la mayoría de los fabricantes de microcontroladores suministran placas entrenadoras, con el soporte software necesario para utilizarlas de una manera adecuada. Las referencias de web [1] [5] [7] [10] [13] [16] constituyen una pequeña revisión de los sistemas de

este tipo disponibles en el mercado en este momento. Con un planteamiento específicamente educativo podemos encontrar algunos productos de Lego [8] que tienen la ventaja añadida de proporcionar las conocidas facilidades para la construcción de prototipos de estructuras mecánicas y acoplamiento de sensores.

Dentro de los sistemas construidos, para cubrir exclusivamente objetivos docentes, vale la pena destacar la controladora Enconor [17], desarrollada para alcanzar metas en el ámbito de la enseñanza secundaria, por profesores que imparten docencia en dicho nivel.

Fuera del ámbito comercial, existen antecedentes importantes en el terreno de los sistemas que permiten la programación de microcontroladores de coste reducido. Destacan posiblemente aquellos que pueden reprogramar el dispositivo sin necesidad de extraerlo del circuito en el que se encuentra alojado, como por ejemplo los planteados en las referencias [2] [14]. Sin embargo, en los casos indicados, para leer y enviar datos se utiliza el puerto paralelo de un computador personal según la especificación *Centronics*. Ello obliga a introducir componentes para contar con líneas bidireccionales de datos, distintos de los intrínsecamente necesarios para el cometido original del circuito.

Una alternativa distinta, que también permite eliminar la necesidad de adquirir un dispositivo grabador, consiste en introducir dentro del microcontrolador un programa cargador [15]. Dicho programa, al empezar a funcionar, explora si lo que se quiere hacer es cargar un nuevo programa

en la zona libre de memoria o ejecutar el programa actualmente cargado. Esta alternativa, que también es muy interesante, tiene como contrapartidas la reducción del espacio de memoria disponible para la aplicación, al tiempo que persiste el problema de grabar la memoria del componente la primera vez.

En relación con el software de grabación de la memoria de los microcontroladores, se pueden hacer las siguientes consideraciones: Por un lado los fabricantes de programadores comerciales incluyen habitualmente el software de grabación en sus productos. Dichos programas suelen tener una dependencia muy fuerte con el dispositivo grabador, por lo que habitualmente no tiene utilidad sin el mismo. Por otro lado, las alternativas que tienen como objetivo reducir el precio del grabador o que eliminan la necesidad del mismo, generalmente presentan un software de grabación de características muy limitadas, eliminando en ocasiones la posibilidad de aprovechar todas las características del microcontrolador afectado.

En el entorno que presentamos hemos eliminado la necesidad de añadir nuevo hardware al circuito de nuestro controlador, por apoyarnos en el estándar IEEE 1284 para puertos paralelos (que actualmente respetan los fabricantes de computadores de forma casi generalizada), en el que se incluye, entre otros, un modo de explotación bidireccional del bus de datos (Byte mode [6]). No hemos optado tampoco por introducir un cargador en el microcontrolador, por lo que disponemos de toda la memoria para albergar programas.

2. Introducción

Es sabido que además de enseñar conceptos es igualmente importante enseñar procedimientos que los utilicen. La aplicación mediante experimentación práctica de estos procedimientos es un aspecto clave para afianzar el aprendizaje del tipo de conocimientos manejados en nuestro campo.

Una de las dificultades a las que se enfrenta el profesor que imparte docencia en el área del desarrollo de sistemas basados en microcontrolador, es el problema de disponer de recursos eficaces y de bajo coste, que permitan realizar prácticas de diseño y desarrollo de sistemas con un planteamiento abierto y cuyos resultados hardware y software, al quedar en propiedad del alumno,

dejen en sus manos la posibilidad y motivación para nuevos desarrollos personales.

Con el propósito de realizar una aportación hacia la solución de este problema, nos hemos planteado la construcción de la herramienta que presentamos, siendo sus requisitos generales los siguientes:

1. Proporcionar un circuito adecuado para la construcción de prototipos con microcontroladores.
2. Proporcionar la posibilidad de programar la memoria del microcontrolador sin la necesidad de adquirir un sistema adicional.
3. Proporcionar facilidades para la organización de la documentación del proyecto y por lo tanto para la creación de una librería de proyectos reutilizables.

La presente comunicación está organizada de la siguiente manera: en primer lugar presentamos, el circuito que proponemos para desarrollar prototipos. En segundo lugar, para facilitar la grabación de la memoria del microcontrolador y la organización del proyecto, exponemos las características más destacadas del programa construido.

3. Facilidades para el diseño y construcción del hardware del prototipo

En los desarrollos con microcontroladores existe una inversión fija, difícil de eliminar, y es la asociada con la placa que debemos utilizar para hacer prototipos. Es cierto que cuando un desarrollo se encuentra en una etapa avanzada es generalmente adecuado diseñar un circuito impreso específico, pero mientras no se alcanza dicha situación tiene interés contar con placas que tengan los componentes que consideramos fijos, así como algunas zonas pretaladradas (en la misma placa o en placas separadas), de forma que se facilite la inserción de nuevos componentes y nuevas conexiones entre los mismos. Por este motivo proponemos una placa de propósito general, en la que hemos incluido el espacio y las conexiones para los elementos que con más frecuencia se utilizan, así como también una zona pretaladrada.

En relación con el segundo requisito tiene interés reducir el precio del entorno, lo cual se puede conseguir por diferentes caminos. Nosotros

hemos valorado especialmente la posibilidad de grabar la memoria del microcontrolador utilizando niveles TTL. Esta alternativa permite utilizar tensiones fácilmente generables desde un computador personal, eliminando con ello la exigencia de adquirir o construir un dispositivo grabador.

Queremos destacar que el programa de grabación y la placa de prototipos propuesta son elementos separados que pueden utilizarse juntos o de forma independiente. El programa de grabación se puede utilizar con otras placas, siempre que se suministre la interfaz adecuada con el computador y es evidente que no es necesario grabar el microcontrolador con el programa que aquí proponemos.

Para facilitar el desarrollo de prototipos, proponemos un circuito basado en el microcontrolador PIC16F873. Con él introducimos un conjunto de componentes reducido, bien conocidos y fácilmente localizables en el mercado, al tiempo que reservamos una zona pretaladrada para insertar nuevos elementos según sean las necesidades de cada proyecto concreto.

Dicho circuito contiene los siguientes bloques principales: microcontrolador (16F873), módulo generador de niveles RS232 (MAX232), componentes de alimentación y conectores. Describimos brevemente a continuación los mismos, después de introducir las características principales del conjunto.

3.1 Características principales

Las características más importantes de la placa que presentamos son las siguientes (ver figura 1):

1. Hemos intentado promover un ahorro de tiempo a la hora de preparar los montajes que se pretendan evaluar, sin que con ello se merme excesivamente la calidad del prototipo construido.
2. Se han ubicado los conectores de los puertos paralelos cerca de la zona pretaladrada para evitar, en la medida de lo posible, la existencia de cables sobre los componentes fijos.
3. Se incluye un conector para poder efectuar la grabación del microcontrolador desde un PC (o cualquier otro dispositivo que se adapte al protocolo de grabación del PIC).
4. Para minimizar el precio de la placa y facilitar la posibilidad de establecer conexiones por la técnica de "wrapping", se han escogi-

do tiras de *pin*s como conectores, salvo para el caso del RS-232.

5. Son accesibles todas las líneas de los puertos paralelos desde sus conectores, con independencia de que algunas de ellas lleguen a otros elementos del circuito (ej. conector de grabación), por tener un cometido especial dependiendo de las circunstancias o de las operaciones que se pretendan hacer con el microcontrolador.
6. Hay conectores dentro de la placa que permiten acceder a GND.
7. Hay un conector dentro de la placa que permite acceder a las tensiones de alimentación existentes antes y después del regulador (7805).

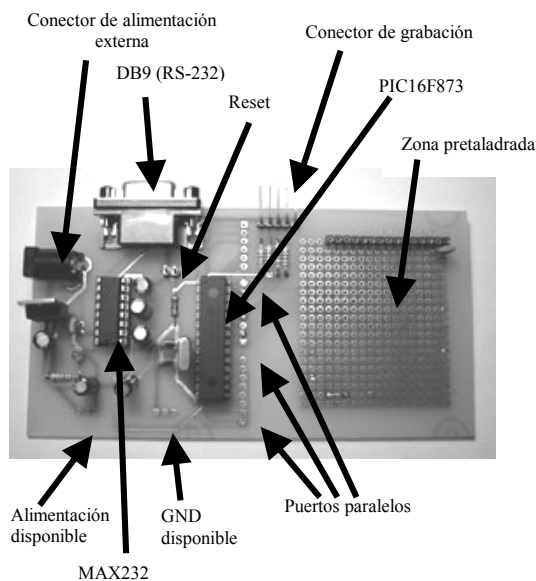


Figura 1. Placa iFOTON con componentes.

3.2 Microcontrolador

El microcontrolador seleccionado es el PIC16F873 de la empresa Microchip [10], [12]. Es difícil defender con detalle, hoy en día, una elección de este tipo frente a otras alternativas, dada la gran variedad de microcontroladores que hay en el mercado con posibilidades extraordinarias.

rias. Nos limitaremos a comentar algunos aspectos destacados del mismo. Su arquitectura es RISC y tiene integrados los siguientes periféricos: convertidor analógico digital de 10 bits, generador de señales PWM, módulo comparador y capturador, módulos de comunicaciones (USART, I2C), 3 puertos paralelos y 3 timers.

Tiene 3 bancos de memoria separados. Un banco de 4K palabras de 14 bits de memoria FLASH para código. Un banco SRAM de 192 bytes de datos y un banco EEPROM de 128 bytes de datos (regrabable en tiempo de ejecución y en tiempo de programación).

Como característica notable del PIC16F873 (que también tienen otros microcontroladores del mismo fabricante) hay que destacar la posibilidad de grabar su memoria con niveles bajos de tensión (Low Voltage in Circuit Serial Programming [9]), lo que permite programar el microcontrolador, incluso en el circuito en el que se encuentra insertado, sin recurrir a los habituales niveles de 13 o 14 Voltios. En concreto, podemos generar las señales necesarias para la lectura y escritura desde el puerto paralelo de un computador, como más adelante comentaremos al hablar del programa de grabación.

Para terminar este apartado queremos destacar que el microcontrolador seleccionado ofrece una excelente relación calidad precio como demuestra la gran difusión que tiene actualmente en el mercado.

3.3 Módulo generador de niveles RS-232

Como se ha indicado el microcontrolador proporciona una USART que respeta la norma RS-232, pero genera señales con niveles TTL. Para adaptar la amplitud de las mismas a los niveles que permite la norma, se incluye el componente MAX232. Sobre decir que si no es necesario utilizar estas facilidades de comunicaciones, en el prototipo que vamos a desarrollar, podemos utilizar la placa que proponemos prescindiendo del MAX232 y de los condensadores asociados, sin ningún perjuicio para el funcionamiento del resto del circuito.

3.4 Componentes de alimentación

Para alimentar la placa hemos considerado interesante poder utilizar transformadores de los que habitualmente se suministran con los altavoces de

los PC's o aparatos similares. Por ello hemos incluido un conector tipo *jack*, seguido de un regulador de tensión 7805 (Fairchild). Se ha incluido también un *led* que nos permite comprobar si la placa está alimentada o no.

3.5 Conectores

Para poder acceder a los diferentes servicios, que suministran los elementos anteriormente mencionados, ha sido necesario dotar al circuito de un conjunto de conectores que permita con facilidad explotar sus posibilidades.

Pueden dividirse de la siguiente manera:

1. Conectores de puertos paralelos. Permiten acceder a los puertos paralelos del microcontrolador.
2. Conector de comunicaciones RS-232. Es un conector del tipo D-Shell (DB9) por el que se puede acceder a las líneas de comunicación suministradas por el componente MAX232.
3. Conector de grabación serie. Por medio del cual podemos activar los servicios de lectura y grabación de la memoria no volátil del microcontrolador desde un programador externo (ej. un computador personal).
4. Conector de alimentación. Permite el acceso a las tensiones disponibles en la placa de circuito impreso: La tensión regulada suministrada por el componente 7805 o la que directamente suministre la fuente de alimentación externa.
5. Conector de tierra. Permite acceder a GND para cubrir las necesidades de tensión de referencia de los nuevos componentes que necesitamos añadir.

4. Facilidades para grabar la memoria y para la organización del proyecto.

El programa de grabación permite borrar, leer y grabar los bancos de memoria no volátil del microcontrolador (PIC16F873). Para ello es necesario conectar el puerto paralelo del computador con el conector de grabación de la placa para prototipos anteriormente presentada y utilizar las diferentes opciones que brinda el programa desarrollado.

eguidamente exponemos los aspectos más destacados relacionados con la utilización de dicho programa.

4.1 Grabación con niveles bajos de tensión

El microcontrolador PIC16F873 permite grabar su memoria con niveles bajos de tensión. En su palabra de configuración existe un bit que determina si está activada esta opción (LVP), o por el contrario hay que utilizar el procedimiento ordinario que exige tensiones del orden de 13.5 Voltios. El componente viene configurado de fábrica con el bit LVP activado.

Hay una pequeña contrapartida cuando se activa el bit LVP y es que la patilla RB3 del microcontrolador deja de ser un *pin* de propósito general en el puerto paralelo B, dado que interviene en el protocolo de entrada en modo grabación. Conectando dicha patilla a GND garantizamos que el microcontrolador no entra en modo grabación, ejecutando al arrancar el programa cargado en memoria. Por ello, el conector de grabación ha sido diseñado de tal manera que cuando no está acoplado el cable se pueda insertar un puente que precisamente establece esta conexión (Figura 2).

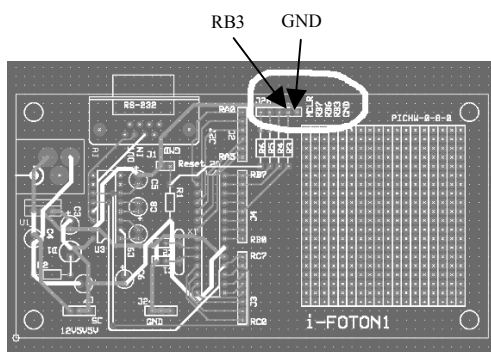


Figura 2. Diseño físico iFOTON.

4.2 Cable de conexión entre el computador y la placa de prototipos

Para entrar en modo programación es necesario que pasen a nivel bajo los pines RB6, RB7, RB3 y MCLR del microcontrolador, seguidamente debe pasar a nivel alto el pin RB3 y posteriormente MCLR (los detalles de temporización pueden

verse en [12]). En modo programación el pin RB6 pasa a desempeñar funciones de reloj de sincronización y el pin RB7 pasa a ser de datos. Este último (RB7) debe ser bidireccional, dado que se utilizará en el sentido Computador→Placa cuando se quieran enviar comandos y datos al microcontrolador, mientras que el sentido será Placa→Computador cuando se realicen operaciones de lectura de la memoria.

Cuando el microcontrolador funciona ejecutando el programa que tiene cargado en memoria, los pines RB6 y RB7 vuelven a ser de propósito general, por lo que pueden ser utilizados libremente en nuestros proyectos.

Por lo dicho anteriormente, las señales que debemos contemplar a la hora de construir el cable de conexión con el computador son cuatro, debiendo añadirse un quinto cable para GND. Para que las operaciones de lectura, borrado y grabación se realicen correctamente con nuestro programa es necesario establecer las conexiones que planteamos a continuación (Tabla 1), en el supuesto de que el conector del puerto paralelo del computador sea del tipo D-Shell 25 (el más habitual)⁽¹⁾.

PIN puerto paralelo	Conector grabación PIC
1	RB6 (Reloj)
2	RB7 (Datos)
16	RB3 (Control)
17	MCLR
18, 19, 20, 21, 22, 23, 24 y 25	GND

Tabla 1. Conexiones del cable de grabación.

4.3 Arranque del programa

Lo primero que realiza el programa al arrancar es una prueba de velocidad del computador. La información obtenida es utilizada para realizar una correcta temporización de las señales, y así respetar las restricciones que impone el fabricante del microcontrolador. Esta forma de actuar ha sido

⁽¹⁾ Para realizar el cable hemos utilizado par trenzado y no hemos tenido problemas con longitudes inferiores a 2 metros.

necesario implantarla por haber tenido problemas al utilizar los servicios suministrado por el nivel API del sistema operativo utilizado (la versión actual funciona sobre Windows 95-98 y con un sistema de desprotección de puertos también en NT y Windows 2000. Las siguientes versiones que se elaboren tendrán como soporte adicional plataformas LINUX).

Una vez realizada la prueba de velocidad, el sistema abre una ventana en la que puede accederse a dos menús desplegables. El primero de ellos permite administrar la estructura de información de los proyectos que desarrollemos (cuya utilización es opcional), mientras que el segundo permite acceder a los servicios de programación del microcontrolador propiamente dichos.

4.4 Menú de proyectos

Dado que uno de los cometidos del entorno de desarrollo precisamente es ser una herramienta de carácter didáctico que facilite la realización de prácticas con microcontroladores, consideramos adecuado fomentar ciertas costumbres metodológicas en los futuros ingenieros que formamos. El motivo del presente menú es obligar a los alumnos a completar y documentar una serie de etapas (requisitos, diseño, etc.) antes de abordar la programación del microcontrolador.

Las opciones disponibles son: “Crear o modificar proyecto”, “Abrir proyecto” o “Configurar aplicación”.

“Crear o modificar proyecto” permite indicar qué ficheros contienen información relativa a un proyecto en concreto. Los proyectos creados en esta aplicación pueden estar constituidos por los ficheros: programa fuente, requisitos, diseño del proyecto, diagramas del circuito, fichero adicional y fichero ejecutable.

“Abrir proyecto” permite obtener el nombre de los ficheros que tienen información de un proyecto así como editarlos, invocando automáticamente a los programas de edición que se han especificado para tal efecto en la opción “Configurar aplicación”.

4.5 Menú programador

Este menú permite efectuar las operaciones de borrado, lectura y grabación de la memoria del microcontrolador.

Las opciones disponibles son: “Configurar puerto paralelo”, “Grabar microcontrolador”, “Leer microcontrolador” y “Borrar microcontrolador”.

“Configurar puerto paralelo” es una opción que posiblemente no sea necesario utilizar nunca, pero se suministra para dar cobertura a posibles situaciones extraordinarias. Por defecto el programa asume que los registros del controlador del puerto paralelo están ubicados a partir de la dirección \$378. Si este no es el caso, esta opción suministra otras direcciones habituales para poder elegir, así como la posibilidad de especificar una distinta a todas las ofertadas.

“Grabar microcontrolador” solicita al usuario el nombre del fichero que contiene el código ejecutable a grabar (sí existía un proyecto abierto toma como elección por defecto el fichero ejecutable del mismo) pudiendo especificarse también dentro de dicho fichero el contenido de la memoria de datos EEPROM⁽¹⁾. El código ejecutable debe suministrarse respetando el formato INHX8M de Intel [11], opción que normalmente suministran los ensambladores y compiladores que generan código para los microcontroladores PIC.

La opción “Leer microcontrolador” permite cargar en el microcontrolador el contenido de la memoria de código, de la memoria EEPROM, la palabra de identificación y la palabra de configuración del microcontrolador. Al activar esta utilidad la aplicación abre una nueva ventana en la que se muestran los datos recogidos, al tiempo que se da la opción de guardarlos en un fichero (con formato INHX8M) por si posteriormente los queremos utilizar para una nueva grabación.

Por último, “Borrar microcontrolador” es una opción que nos permite borrar de manera independiente la zona de código (FLASH), la de datos (EEPROM), o borrar un microcontrolador protegido. En este último caso es el propio microcontrolador quien garantiza la confidencialidad del código y los datos almacenados, dado que antes de desproteger los bancos de memoria los borra.

⁽¹⁾ Al grabar no se pueden especificar los 16 primeros bytes (de los 128 disponibles en la EEPROM), dado que en el proceso de escritura el micro utiliza un espacio de direccionamiento, imagen de la memoria EEPROM, que tiene reservados dichos bytes para otros cometidos.

5. Experiencia docente con el sistema presentado

El entorno que hemos presentado ha sido ya utilizado con éxito para la realización de prácticas en la Facultad de Informática de la Universidad Politécnica de Madrid. Concretamente en las asignaturas “Instrumentación y Adquisición de Datos” de 4º curso y “Dispositivos Optoelectrónicos Aplicados a la Informática” de 5º curso. En la primera los alumnos desarrollan un prototipo de estación meteorológica, utilizando el sistema presentado en combinación con la herramienta software LabView de National Instruments. En la asignatura de 5º curso los alumnos deben construir un analizador de la señal infrarroja enviada por distintos mandos a distancia.

Para construir la placa, el estudiante debe seguir un plan detallado, disponible en la página de web de la asignatura [3]. En dicho documento se especifican los pasos a seguir y comprobaciones a realizar, durante el proceso de montaje del circuito sobre la placa para prototipos.

Una descripción más detallada de las aplicaciones didácticas de iFOTON se puede encontrar en [4]

6. Conclusiones

La herramienta presentada ha demostrado ya su versatilidad en pequeños proyectos desarrollados por los autores (Contador de consumo de energía eléctrica, Control de posición del carro de una impresora, Control de velocidad en un motor D.C. con carga variable, etc.) así como en la realización de las prácticas citadas en asignaturas de nivel universitario.

Con estas primeras experiencias se verifican la consecución de los objetivos planteados al comprobarse una mayor facilidad para desarrollar prototipos con el microcontrolador seleccionado, al tiempo que el entorno demuestra tener interés didáctico por permitir realizar prácticas reales con microcontroladores a un precio realmente reducido. Por otro lado, ha sido satisfactorio comprobar como los mismos alumnos han propuesto la reutilización de la placa montada en un curso, para realizar las prácticas del curso siguiente.

Hemos comprobado también el efecto motivador que produce la actividad de montaje del

circuito, en la que se incluye la soldadura de componentes y el proceso de verificación.

Como líneas de evolución planteadas está en estudio incorporar al entorno la posibilidad de utilizar otros microcontroladores de la misma casa, así como desarrollar la correspondiente versión para plataformas LINUX.

A partir de los desarrollos de los primeros usuarios de la herramienta, estamos creando una base de datos compartida de proyectos reutilizables hardware-software y acumulando experiencia para la mejora de la sencilla herramienta de gestión de documentación disponible.

Referencias

- [1] AMD. *Embedded Processors*. <http://www3pub.amd.com/products/epd/>. AMD, 28-1-2002.
- [2] Antonio Serna y José Vicente García. *Desarrollo y Construcción de Prototipos Electrónicos*. (pag 59-77). Ed. Paraninfo, 2000.
- [3] Gracián Triviño. *iFOTON*. <http://www.dtf.fi.upm.es/~gtrivino/IFOTON>, 28-1-2002.
- [4] Gracián Triviño y Norberto Cañas. *Recursos didácticos para el diseño y construcción de instrumentación digital a coste reducido*. (Enviado para su aceptación a: SAAEI2002. Seminario Anual de Automática, Electrónica industrial e Instrumentación. Alcalá de Henares, Septiembre 2002)
- [5] IBM. *IBM Microelectronics*. <http://www-3.ibm.com/chips/products/>. IBM, 28-1-2002.
- [6] IEEE. *IEEE Standard Signaling Method for a Bidirectional Parallel Peripheral Interface for Personal Computers*. Estándar IEEE 1284-2000 (Revision of IEEE Std 1284-1994), 2000.
- [7] Intel. *Embedded Products*. <http://www.intel.com/design/embprod.htm>. Intel Corporation, 31-12-2002.

- [8] Lego. *MindStorm*. <http://mindstorms.lego.com/>. Lego Group, 28-1-2002.
- [9] Microchip. *EEPROM Memory Programming Specification*. Documento DS39025E. Microchip Technology Inc. 2000.
- [10] Microchip. *Microchip Product Lines*. <http://www.microchip.com>. Microchip Technology Inc. 28-1-2002.
- [11] Microchip. *MPASM USER'S GUIDE with MPLINK and MPLIB*. Microchip Technology Inc. 1999.
- [12] Microchip. *PIC16F87X. 28/40-pin 8-Bit CMOS FLASH Microcontrollers*. Documento DS30292B Microchip Technology Inc. 1999.
- [13] Motorola. *Design Resources*. <http://e-www.motorola.com/>. Motorola, 28-1-2002.
- [14] Robert Spur. *A PC-Based Development Programmer for the PIC16C84*. Application Note AN589. Microchip Technology Inc. 1999.
- [15] Rodger Richey. *Downloading HEX files to PIC16F87X PIC Microcontrollers*. Documento DS91025a, Microchip Technology Inc. 1998.
- [16] Texas Instruments. *DSP Developers' Village*. <http://dspvillage.ti.com/docs/dspvillagehome.jhtm>. Texas Instruments, 28-1-2002.
- [17] *Controladora Educativa ENCONOR*. <http://www.enconor.com/> 31-12-2001.