

MAXHC11. TARJETA DE BAJO COSTE PARA EL DISEÑO MIXTO *HARDWARE-SOFTWARE*

Sadot Alexandres F.¹, José D. Muñoz F.², Pedro Pérez de A.³

Departamento de Electrónica y Automática.
Escuela Técnica Superior de Ingeniería
Universidad Pontificia Comillas de Madrid.

¹sadot@upco.es

²daniel@dea.icaei.upco.es

³payala@ya.com

RESUMEN

Se presenta en este artículo una tarjeta basada en un microcontrolador de la familia Motorola y en dispositivos programables para el diseño de sistemas digitales. La tarjeta se ha adaptado perfectamente en la integración de robots móviles autónomos, en la automatización de sistemas y en general al desarrollo docente de las titulaciones de ingeniería de la Escuela Técnica Superior de Ingeniería ICAI de la Universidad Pontificia Comillas. Además, la disponibilidad de un *software* de desarrollo de alto nivel basado en software libre ha facilitado el uso de la tarjeta por parte de los alumnos sin incurrir en elevados gastos en licencias.

1. INTRODUCCIÓN

La necesidad de disponer de plataformas de desarrollo de bajo coste que permitan facilitar el ciclo de diseño *hardware-software* en las tareas docentes, como son prácticas de laboratorio, diseño digital y programación, es una tarea fundamental en las carreras de ingeniería que incluyen los sistemas digitales en sus planes de estudio. Muchas de estas plataformas son diseños propios de los Departamentos de las Facultades y Escuelas, bien por facilidad, bien por motivación propia o bien por un mejor control, flexibilidad y organización docentes [1][2][3][4][5]. La mayoría de este tipo de sistemas está asociado a un microcontrolador o bien a un microprocesador de fácil acceso donde se proyectan diseños propios de automatización, comunicaciones, control o robótica [6][7]

El sistema que aquí se presenta, la tarjeta MAXHC11, ha sido desarrollado basándose en un microcontrolador de la familia HC11 de Motorola como uno de los ejes. Debido a que la memoria interna disponible es muy pequeña, se ha añadido memoria externa para poder desarrollar cómodamente aplicaciones complejas escritas en lenguaje C, ya que el ensamblador, aunque en muchas situaciones es el lenguaje idóneo, no lo es para la programación de tareas complejas, tales como la programación de robots. Además se han añadido muchas facilidades *hardware* que si bien parece que a simple vista aumentan el tamaño y el coste de la tarjeta, simplifican y minimizan el ciclo de diseño de forma significativa. Tal es el caso de circuitos programables del tipo PLD. Además, esto último permite añadir el complemento docente del diseño digital, a través de la planificación e integración de *hardware* a medida usando lenguajes de descripción de *hardware* (HDLs) como Verilog o VHDL.

Además de la memoria externa y los circuitos programables, se han añadido una serie de periféricos, los cuales se han escogido basándose en las necesidades observadas durante el

desarrollo de muchos proyectos final de carrera [8][9][10][11][12]. Una necesidad muy importante es la de disponer de capacidades de depuración y de interfaz con el usuario. Para esta tarea se han añadido un visor LCD que permite presentar suficiente información sobre el flujo de programa y el estado de las variables del sistema. También se han añadido una serie de diodos led y de visualizadores de 7 segmentos que tienen como principal ventaja sobre el display LCD su facilidad de manejo.

2. DESCRIPCIÓN DE LA TARJETA MAXHC11

La tarjeta MAXHC11 es una tarjeta de desarrollo para sistemas digitales de propósito general, aunque sus aplicaciones están dirigidas a la automatización y control. Se ha elegido la serie Motorola por su facilidad de montaje, por poseer una arquitectura muy amigable para el programador y por disponer de *software* de desarrollo de alto nivel libre [17]. Un diagrama de bloques de la tarjeta se muestra en la figura 1. Como se puede apreciar, están disponibles 3 puertos de E/S (B, D y E). Los otros dos puertos (A y C) se usan como buses de datos y direcciones para acceder al mapa de memoria externa. No obstante se han dispuesto dos conectores para poder usar estos dos puertos para E/S si la aplicación sólo usa memoria interna. El mapa de memoria externa está controlado por la FPGA1. Como se puede apreciar, se dispone de una memoria RAM de 32 kB y de una memoria EPROM de 32 kB. Además existe un display LCD que ocupa dos posiciones de memoria. La FPGA1 también incluye dos puertos de 8 bits (puertos 1 y 2) para sustituir a los dos puertos que se pierden al usar memoria externa. El resto de periféricos, tres visualizadores de siete segmentos y cuatro pulsadores también se controlan por medio de la FPGA1. La segunda FPGA está totalmente libre para aplicaciones del usuario. Por último, conviene destacar que la tarjeta se ha diseñado para que sea compatible con la CT6811 de microbótica [15], de forma que se puedan usar el *software* y los periféricos diseñados para ésta.

Para el desarrollo *software*, la placa incorpora un puerto serie a través del cual se pueden descargar los programas usando el modo *bootstrap* del HC11. Durante la fase de desarrollo los programas se descargan en RAM, lo cual es obviamente lo más flexible. Para ello la RAM se sitúa en los 32 kB más altos de la memoria. No obstante, para permitir un funcionamiento autónomo una vez finalizada la aplicación, se ha incluido una EPROM para almacenar el programa final. Para minimizar los cambios al *software*, es posible intercambiar el mapa de memoria de forma que la EPROM ocupe la zona alta de la memoria en lugar de la RAM.

Se ha tratado también de facilitar la tarea de montaje por parte del alumno, por lo que aparte de haber cuidado la serigrafía y la disposición de los componentes se ha desarrollado una sencilla guía de montaje. Además, dado que la finalidad principal de la tarjeta es la de servir de plataforma de desarrollo para proyectos fin de carrera, se han dispuesto en la tarjeta una serie de puntos de acceso para pruebas y expansión. Por ejemplo, están disponibles todos los pines de las FPGAs para que sea fácil añadir circuitería externa controlada por éstas. En la figura 2 se muestra la serigrafía y en la figura 3 se muestra una fotografía de la tarjeta.

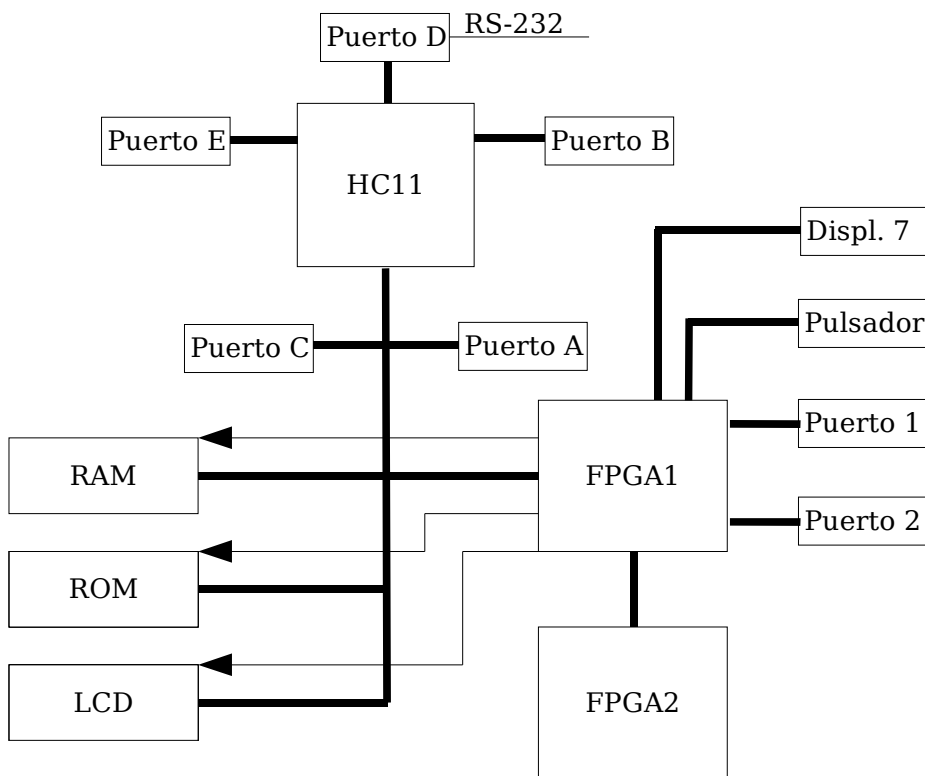


Figura 1. Diagrama de bloques de la tarjeta

A modo de resumen se detallan las características principales de la tarjeta:

- Microcontrolador MCHC11 a 8 MHz .
 - Incluye RAM y EPROM externa de 64Kb
 - Cuatro puertos de 8 bits.
 - Puerto A de E/S disponible al exterior.
 - Puerto B de Salida disponible al exterior y al conversor A/D.
 - Puerto C de E/S disponible al exterior.
 - Puerto E de Entrada disponible al exterior.
 - Un puerto de 6 bits.
 - Puerto D de E/S disponible al exterior.
- Dos módulos FPGAs (CPLDs concretamente) del tipo MAX7128SLC84-15 de 1 fabricante Altera.
- Una conexión RS232 para interfaces serie.
- Conexiones y puertos de volcado para PLDs tipo byte-blaster.
- Entradas de pulsadores, interruptores, leds y un zumbador.
- Fuente de alimentación regulada.
- Interfases para LCD y visualizadores de 7 segmentos.

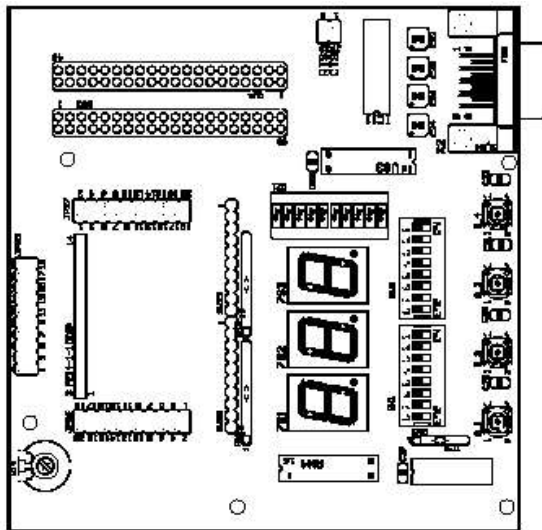
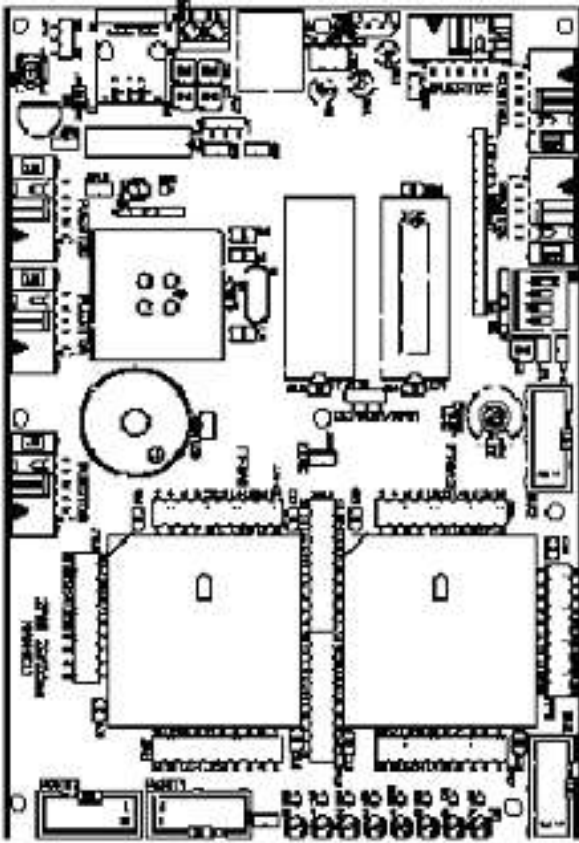


Figura 2 Serigrafía de los componentes

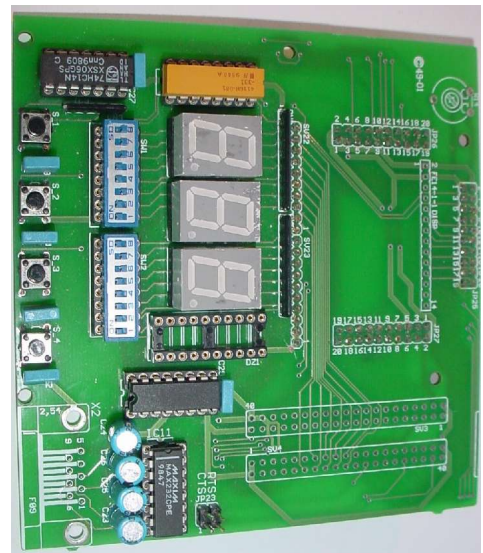


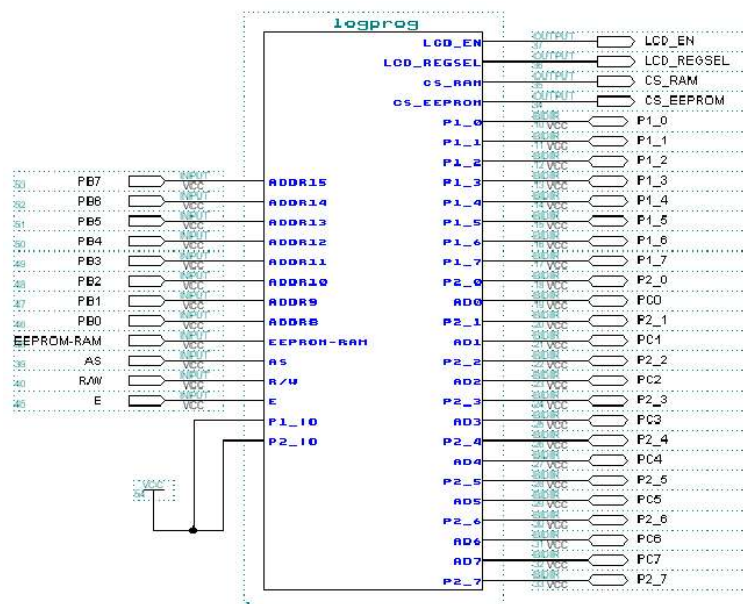
Figura 3 Fotografía de la tarjeta

3. LÓGICA PROGRAMABLE

El módulo de lógica programable consiste en dos FPGAs del fabricante ALTERA [14], concretamente el dispositivo, MAX7128SLC84-15. Estos, aún siendo de los más sencillos, disponen de capacidad suficiente para implementar en ellas circuitos digitales de apoyo al microcontrolador lo suficientemente potentes para aplicaciones y diseños de tipo docentes en la automatización y robótica.

Para que la tarjeta MAXHC11 arranque en el modo básico y funcione correctamente la FPGA-1 necesita tener cargada un pequeño circuito (denominado logprog) que, como se ha dicho antes, implanta la decodificación del mapa de memoria y la interfaz con algunos periféricos. Este circuito se carga a través del cable byte-blaster usando el *software* de desarrollo MAX-Plus II de Altera y puede ser alterado para adecuarse a las necesidades de cada usuario. Si se desea añadir otros circuitos a la primera FPGA, se tendrá que modificar el circuito logprog. No obstante, en este caso ha de respetarse el circuito original y la asignación de patillas de la FPGA que usa éste ya que estos pines no solo están conectados a los terminales de acceso, sino también al resto de la tarjeta. Por tanto es imprescindible respetar esta asignación de patillas para evitar dañar los buses del microcontrolador y las memorias. A continuación se muestra el esquema de más alto nivel del circuito, en el que se pueden apreciar las señales que entran y salen del módulo.

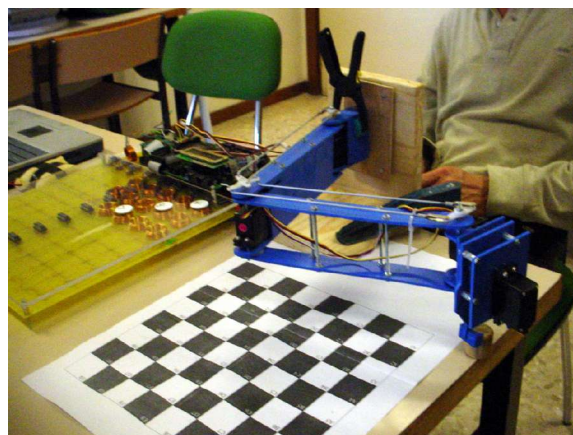
Se utiliza aproximadamente un 28% de la FPGA para incorporar los sistemas necesarios. El resto del espacio de esta FPGA queda libre para que el usuario desarrolle sus propios dispositivos. La segunda FPGA esta íntegramente disponible para el usuario.



4. APLICACIONES E INTEGRACIÓN

Durante estos últimos tres años, se han desarrollado varias aplicaciones docentes sobre la tarjeta MAXHC11, principalmente en el campo de la robótica móvil, aunque no ha sido la única. Otras aplicaciones se han venido implantando con éxito [11]. El sistema permite

desarrollar *software* a través de una plataforma Linux o MS-DOS usando lenguaje ensamblador o lenguaje C de una forma fácil, pudiéndose en ambos casos acceder a todos los recursos del microcontrolador. A modo de ejemplo, se puede ver la facilidad de uso de la tarjeta a nivel de alumnos de las titulaciones de ingeniería en algunos de los diseños creados para competiciones de micro robots [16].



5. DESARROLLO Y PROGRAMACIÓN

La tarjeta MAXHC11 permite desarrollos *hardware* usando las FPGAS incluidas en la placa y desarrollos *software* en el microcontrolador HC11. En ambos casos está disponible *software* gratuito de forma que el alumno pueda usar la tarjeta sin incurrir en costosos gastos de licencias.

Para el desarrollo *hardware* se usa la versión de estudiante del *software* MAX+Plus II de Altera, disponible a través de su página web [18].

La programación del microcontrolador se puede realizar en lenguaje ensamblador utilizando el ensamblador AS11 de Motorola, el cual es *freeware*. Si se desea programar en lenguaje C se puede usar una versión para HC11 del compilador de código libre gcc del proyecto GNU [17]. Ambas herramientas están disponibles para plataformas Linux y MS-DOS.

La carga de programas en la tarjeta se realiza mediante las herramientas de Microbótica [15]. Estas herramientas utilizan el modo *bootstrap* del HC11 para cargar un programa en la memoria interna del HC11. Existen dos modos de funcionamiento:

- Si se desea cargar un programa en memoria y ejecutarlo sin más, se puede usar la herramienta `downmcpu`
- Si se desea depurar el programa o cargarlo en la memoria EEPROM interna del HC11 es necesario utilizar un programa monitor. Este programa está formado por dos componentes: Un programa servidor que se ejecuta en el HC11 (`ctserver`) y que es necesario descargarlo previamente mediante `downmcpu`, y un programa cliente (`ctdialog`) que se ejecuta en el PC.

5. CONCLUSIONES

Se ha desarrollado un sistema de bajo coste basado en microprocesador y lógica programable que permite realizar aplicaciones docentes tanto *software* como *hardware*. Además, los sistemas de desarrollo utilizados son de código abierto o de libre distribución, lo que permite su uso por los estudiantes sin incurrir en elevados costes de licencias.

El sistema desarrollado ha cumplido sus objetivos docentes en los últimos tres años con una excelente aceptación en aplicaciones de sistemas digitales y de sistemas autónomos embarcados. Se destacan:

- Versatilidad de un sistema propio para desarrollar proyectos sencillos con herramientas de libre distribución.
- Ciclo de diseño *hardware-software* y depuración en un mismo sistema.
- Interfaces que permiten la conexión a diferentes dispositivos.

Por último, los autores agradecen el esfuerzo para el desarrollo del MAXHC11 al Departamento de Electrónica y Automática y a los alumnos que han intervenido en este desarrollo, tanto en su diseño y depuración como desarrollando aplicaciones a través de Proyectos Fin de Carrera.

6. BIBLIOGRAFÍA

- [1] J. Ramón y otros. "Plataforma docente de robots móviles, cooperantes y autónomos". TAAE'98. 1998..
- [2] J. de la Rosa y otros. "Laboratorio docente de robots móviles, cooperantes y autónomos". TAAE'00. 2000.
- [3] G.González de Rivera, y otros ."GP_Bot. Plataforma hardware para la enseñanza de robótica en la titulación de Ingeniería Informática." TAAE'02. 2002. pp. 67-70.
- [4] J.García Zubía, y otros. "Prácticas de lógica programable y microbótica: GALBOT". TAAE'02. 2002 pp. 455-458.
- [5] C.Sanz. y otros "Recursos para la electrónica digital" TAAE'02. 2002. pp.137-140
- [6] M.Martínez y otros. "Metodología docente en la evaluación de sistemas digitales mediante software de síntesis lógica". TAAE'02. 2002. pp153.156.
- [7] A.Palacio y otros. "Sistema integrado microcontrolador-FPGA para uso en prácticas de sistemas electrónicos digitales". TAAE'02. 2002 pp. 229-232.
- [8] I.Suaréz. Proyecto Fin de Carrera. ETSI-UPCO. "Sistema de control de baterías de un robot autónomo: detección de niveles y control de movimiento" 2002
- [9] F.Salanueva. Proyecto Fin de Carrera. ETSI-UPCO. "Robot seguidor de línea". 2001.
- [10] P. Pérez. Proyecto Fin de Carrera ETSI-UPCO. "Robot de sumo, módulo de sensores". 2001.
- [11] P. Pérez. Proyecto Fin de Carrera ETSI-UPCO. "Tablero de ajedrez Inteligente". 2003.
- [12] B. Navarro. Proyecto de Fin de Carrera ETSI-UPCO. "Robot de sumo, modulo de potencia". 2001.
- [13] MC68HC11. Motorola.
- [14] [MAX 7000 Programmable Logic Device Family Data Sheet](http://www.altera.com/literature/ds/m7000.pdf). Altera. Disponible on-line en: <http://www.altera.com/literature/ds/m7000.pdf>. Última visita 2/03/2004.
- [15] Micróbotica. <http://www.microbotica.es>. Última visita: 12/02/2004.
- [16] Congreso y concurso de robótica Hispabot. <http://www.hispabot.org>. Última visita: 12/02/2004.
- [17] Free Software Foundation. Using the GNU Development Tools for 68HC11 and 68HC12. 2003. Disponible on-line en: <http://m68hc11.serveftp.org/doc/guide.html>. Última visita: 12/02/2004.
- [18] Altera. <http://www.altera.com>. Última visita: 1/03/2004.