

Apuntes de: Sistemas embebidos (2009)

Tema 1

© Benito Úbeda Miñarro

ABSTRACT

Introducción y generalidades acerca del diseño de sistema embebidos, empotrados o incrustados.

Tema 1

INTRODUCCIÓN A LOS SISTEMAS EMBEBIDOS (SSEE)

1.1.- Concepto y aplicaciones

1.1.1.- Concepto

Un sistema embebido (SE) o sistema empotrado lo vamos a definir como un sistema electrónico diseñado específicamente para realizar unas determinadas funciones, habitualmente formando parte de un sistema de mayor entidad. La característica principal es que emplea para ello uno o varios procesadores digitales (CPUs) en formato microprocesador, microcontrolador o DSP lo que le permite aportar 'inteligencia' al sistema anfitrión al que ayuda a gobernar y del que forma parte.

En el diseño de un sistema embebido se suelen implicar ingenieros y técnicos especializados tanto en el diseño electrónico hardware como el diseño del software. A su vez también se requerirá la colaboración de los especialistas en el segmento de usuarios de tales dispositivos, si hubiese lugar a ello.

Hardware

Normalmente un sistema embebido se trata de un módulo electrónico alojado dentro de un sistema de mayor entidad ('host' o anfitrión) al que ayuda en la realización tareas tales como el procesamiento de información generada por sensores, el control de determinados actuadores, etc.. El núcleo de dicho módulo lo forma al menos una CPU en cualquiera de los formatos conocidos:

- Microprocesador.
- Microcontrolador de 4, 8, 16 o 32 bits.
- DSP de punto fijo o punto flotante.
- Diseño a medida 'custom' tales como los dispositivos FPGA

El módulo o tarjeta, además puede haber sido desarrollado para satisfacer una serie de requisitos específicos de la aplicación a la que está dirigido. Entre éstos, podemos citar:

- Tamaño: por lo general deberá ser reducido, aunque también es posible que se desee que adopte un formato estándar: PC-104, Eurocard, etc.
- Margen de temperatura específico del ámbito de aplicación:
 - o Gran consumo (0°C hasta 70°C)
 - o Industrial y automoción. Márgenes de temperatura hasta 125°C
 - o Aeroespacial
 - o Militar
 - o Electromedicina
- Consumo de energía: En aplicaciones en las que es necesario el empleo de baterías, se buscará minimizar éste.
- Robustez mecánica: Existen aplicaciones donde los dispositivos sufren un alto nivel de vibraciones, golpes bruscos, etc. En el diseño se deberá tener en cuenta dicha posibilidad.
- Coste: No es lo mismo diseñar un producto a medida con pocas unidades que diseñar un producto para el competitivo mercado del gran consumo. La calibración de los costes es esencial y es tarea de los ingenieros de diseño.
- Etc.

Software

En lo que se refiere al software, se tendrán requisitos específicos según la aplicación. En general para el diseño de un SE no se dispone de recursos ilimitados sino que la cantidad de memoria será escasa, la capacidad de cálculo y dispositivos externos será limitada, etc. . Podemos hablar de las siguientes necesidades:

- Trabajo en tiempo real.
- Optimizar al máximo los recursos disponibles.
- Disponer de un sistema de desarrollo específico para cada familia de microprocesadores empleados.

- Programación en ensamblador, aunque en los últimos años, los fabricantes o empresas externas han mejorado la oferta de compiladores que nos permiten trabajar en lenguajes de alto nivel, tales como C.
- etc.

El empleo de un sistema operativo determinado o el no empleo de éste dependerá del sistema a desarrollar y es una de las principales decisiones que habrá que tomar en la fase de diseño del SE. Así, en el caso de decidirse por el empleo de microcontroladores y DSP, por lo general no se usará sistema operativo mientras que si se emplea algún micro del tipo ARM, PowerPC, Intel X86, etc. si que lo llevará. La decisión dependerá de los requisitos del sistema, tanto técnicos como económicos.

Resumen

Podemos concluir finalmente que un SE consiste en un sistema basado en microprocesador cuyo hardware y software están específicamente diseñados y optimizados para resolver un problema concreto de forma eficiente. Normalmente un SE interactúa continuamente con el entorno para vigilar o controlar algún proceso mediante una serie de sensores. Su hardware se diseña normalmente a nivel de chips (SoC, System on Chip) o de tarjeta PCB, buscando minimizar el tamaño, el coste y maximizar el rendimiento y la fiabilidad para una aplicación particular.

También comentar que bajo el concepto amplio de sistemas embebidos se da cabida a toda una serie de técnicas y metodologías de diseño tanto hardware como software. Tratarlas todas ellas con un mínimo de profundidad en una única asignatura cuatrimestral es una tarea inabordable. En esta primera fase se ha optado por dar un enfoque volcado hacia el mundo de los microcontroladores, dado el amplio peso que éstos tienen en las aplicaciones de carácter industrial y consumo, la relativa facilidad con que es posible manejar un sistema de desarrollo y la posibilidad de abordar proyectos prácticos no muy complejos en poco tiempo.

1.1.2.- Aplicaciones

Las aplicaciones mas numerosas y habituales de los SSEE suelen ser del tipo industrial y gran consumo. Existen en el mercado de semiconductores una amplia

variedad de familias de microprocesadores, microcontroladores y DSPs dirigidos a este sector.

En la práctica totalidad de las áreas de nuestra vida nos encontramos con sistemas embebidos que prácticamente nos pasan desapercibidos. Sirva como ejemplo el sector del automóvil, que en pocos años ha introducido notables avances en lo referente a la seguridad, confort, infomovilidad, etc.

Pero, en general, podemos enumerar los siguientes campos de aplicación:

- Equipos industriales de instrumentación, automatización, producción, etc.
- Equipos de comunicaciones.
- En vehículos para transporte terrestre, marítimo y aéreo
- En dispositivos dedicados al sector de consumo tales como electrodomésticos, equipamiento multimedia, juguetes, etc.
- En bioingeniería y electromedicina.
- Sector espacial y de defensa.
- Equipos para domótica.
- Etc.

En la actualidad, todos los fabricantes de semiconductores ofrecen su gama de productos relacionándolos con el amplio rango de aplicaciones a los que van dirigidos. A modo de ejemplo, se reproduce la clasificación que hace Texas Instrument, uno de los líderes mundiales en la fabricación de semiconductores:

- Audio
- Automotive
- Broadband
- Communications & Telecom
- Computers & Peripherals
- Consumer Electronics
- Industrial
- Medical
- Security
- Space, Avionics, & Defense
- Video and Imaging
- Wireless

http://focus.ti.com/apps/docs/appshomepage.tsp?DCMP=TIHeaderTracking&HQS=Other+OT+hdr_a_apps

Un usuario no técnico de un sistema embebido puede no ser consciente de que está usando un sistema computador. En algunos hogares las personas, que no tienen por qué ser usuarias de un ordenador personal estándar (PC), utilizan del orden de diez o más sistemas embebidos cada día: TV, móvil, cámara de fotos, frigorífico, lavadora, coche, etc.

1.2.- Arquitecturas de computadores mas empleadas en los sistemas embebidos.

1.2.1.- Introducción

En el diseño de SSEE basados en microcontroladores, en general no se requiere una gran potencia de procesamiento, ni dispositivos de presentación con gran resolución gráfica ni sistema operativo y si en cambio el trabajo en tiempo real. Tampoco se suelen contemplar las posibilidades de ampliación hardware con nuevos módulos ya que el sistema anfitrión se diseña en su totalidad para unos requisitos específicos, de forma tal que si el sistema anfitrión se queda obsoleto lo será no sólo por la CPU embebida sino también por el resto de los elementos que lo integran, con lo que la única alternativa consistirá en el rediseño del sistema completo, en la mayoría de los casos. Requisitos tales como tamaño, margen de temperatura, consumo e inmunidad ante interferencias electromagnéticas suelen ser de gran importancia.

Sin ánimo de ser exhaustivos, en la figura 1.1 se muestra un diagrama de bloques de lo que puede ser un modelo general de un sistema embebido, de los aquí considerados.

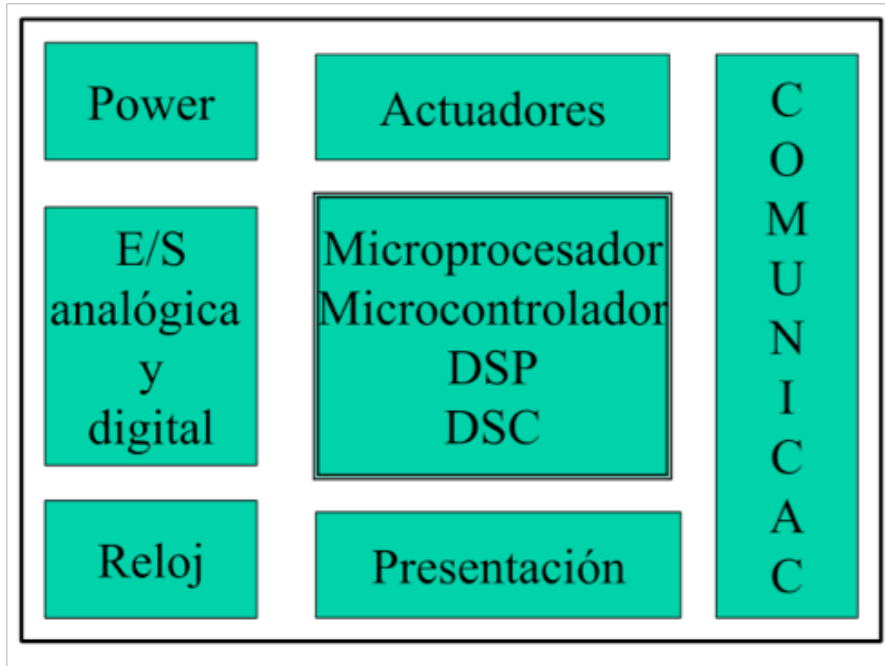


Figura 1.1.- Diagrama de bloques simplificado de un módulo típico de un SE.

Comentamos cada uno de sus módulos a continuación:

Microprocesador, microcontrolador, DSP o DSC



Se entiende que en nuestra definición de SE, éste siempre alberga una o más CPUs ya que son el elemento encargado de aportar la 'inteligencia' al sistema. El formato en el que la CPU se encuentra puede ser el de microprocesador, microcontrolador (μC), DSP, etc. Según este formato, la memoria necesaria puede ir integrada dentro del chip que contiene la CPU, de forma externa a éste o un bajo ambas posibilidades. La oferta en el mercado de semiconductores tanto de microprocesadores como microcontroladores y DSP es elevada y se requiere de una cuidada fase de estudio inicial para seleccionar el más adecuado a cada aplicación.

Básicamente, en el diseño de SSEE haremos uso de cualquiera de los siguientes conceptos:

Microprocesador: Es un chip que incluye básicamente la CPU y circuitería relacionadas con los buses de datos y memoria. Para poder realizar su tarea se necesitan otros chips adicionales (Sistema mínimo) tales como memoria, circuitos de entrada salida E/S (I/O) y reloj.

Microcontrolador (MCU): Es un dispositivo que alberga el sistema mínimo dentro de un único chip, esto es, incluye CPU, buses, reloj, memoria ROM, memoria RAM, E/S, otros periféricos tales como convertidores A/D, temporizadores (timers), etc.

Procesador Digital de Señal (DSP): Son microcontroladores o microprocesadores diseñados específicamente, tanto en arquitectura hardware como conjunto de instrucciones, para realizar tareas típicas de procesamiento digital de señales en tiempo real.

DSC: Dispositivos mixtos microcontrolador/DSP que algunos fabricantes ofrecen dentro de su catálogo de productos.

Probablemente, el microcontrolador 8051 (8 bits) desarrollado por Intel en 1980, marca el inicio en la carrera hacia el desarrollo de productos específicos para aplicaciones embebidas. Éste es probablemente el microcontrolador más popular, pues aunque se lleva hablando mucho tiempo de que estaba condenado a la desaparición, sus continuas mejoras le auguran una larga vida. En este aspecto, comentar que los núcleos 8051 se usan en más de 100 microcontroladores de más de 20 fabricantes independientes como [Atmel](#), [Dallas Semiconductor](#), [Philips](#), [Winbond](#), entre otros. La denominación oficial de Intel para familia de μ Cs 8051 es MCS 51.

Pero a lo largo de estos años, la gran mayoría de empresas fabricantes de semiconductores, han ido lanzando productos que han invadido el mercado y que hace extremadamente difícil para los ingenieros la labor de seleccionar el μ C más adecuado para cada aplicación, pues se dispone de una tremenda oferta de micros de 8 bits, de 16 bits y en la actualidad de 32 bits. A su vez con arquitecturas más completas, capacidades de cálculo más elevadas y menores consumos de energía.

Comunicaciones

Los sistemas de comunicaciones adquieren, en el diseño de sistemas embebidos, cada vez mayor importancia. Lo normal es que el SE pueda comunicarse mediante

interfaces estándar de comunicaciones por cable o inalámbricas. Así un SE normalmente incorpora puertos de comunicaciones bajo los estándares mas extendidos, bien aquellos que necesitan de un cableado físico o se trate de comunicaciones inalámbricas. Podemos citar:

- RS-232
- RS485
- SPI
- CAN
- USB
- Ethernet
- Fibra óptica.
- Comunicaciones inalámbricas (WiFi, WiMax, Bluetooth, GSM, GPRS, UMTS, DSRC, RFID, etc.)

Presentación

El subsistema presentación típico suele ser una pantalla gráfica, táctil, LCD alfanumérico, diodos LED, etc. Por lo general forma parte del interfaz hombre máquina del sistema, si es que lo lleva. El uso de pantallas gráficas del tipo táctil suele ser una solución muy aceptada, aunque conlleva mayor complejidad en el software a desarrollar y mayor potencia de calculo de la CPU seleccionada. En la figura 1.3 se muestra un típico display LCD alfanumérico de dos lineas por 20 caracteres por línea.



Figura 1.2.- Display LCD alfanumérico.

Actuadores

Denominamos actuadores a los posibles elementos encargados de llevar a cabo las acciones indicadas por la CPU. Entre éstos disponemos de drivers de corriente, controladores de motores eléctricos, conmutadores, relés, etc.

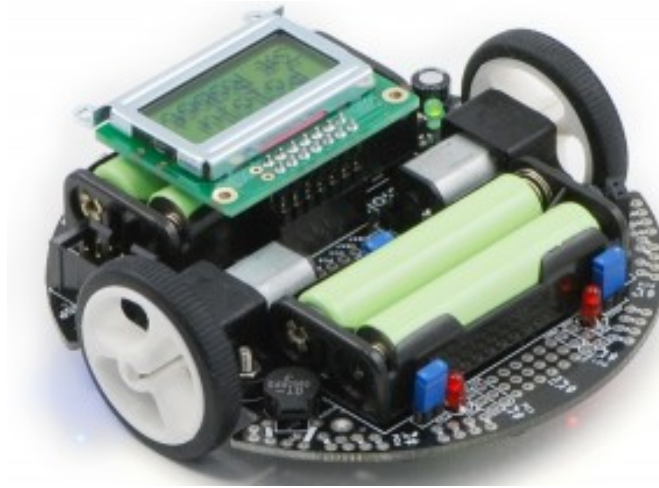


Figura 1.3.- Actuadores de un robot.

Pines de E/S analógicos y digitales

El módulo de Entrada/Salida (I/O) se encarga de hacer llegar o enviar las señales analógicas y digitales a los diferentes circuitos encargados de su generación y procesamiento. Tal es el caso de la conversión A/D para el procesamiento digital de señales analógicas procedentes de sensores, activación de actuadores mediante circuitos 'driver', reconocimiento del estado abierto cerrado de un conmutador o pulsador, encendido de diodos LED, etc.

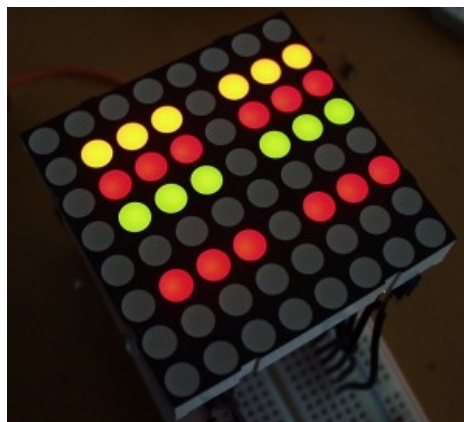


Figura 1.4.- Matriz de diodos LED para iluminación.

Reloj

El módulo de reloj es el encargado de generar las diferentes señales de reloj necesarias para la temporización de los circuitos digitales. Habitualmente se parte de un único oscilador principal, cuyas características son de vital importancia en determinadas aplicaciones. Aspectos a tener en cuenta en la selección del tipo de oscilador son:

- Frecuencia necesaria y la posible selección de ésta de forma automática.
- Estabilidad y precisión de la frecuencia con la temperatura, envejecimiento, vibraciones, etc.
- Consumo de corriente requerido y su complejidad hardware.
- El coste del resonador empleado para construirlo.



Figura 1.5.- Módulos osciladores de cristal de cuarzo.

Existen varios tipos de osciladores basados en el empleo de resonadores de cristal de cuarzo, cerámicos, SAW, LC y RC.

Cada uno de ellos presenta características específicas de margen de frecuencia, estabilidad y consumo. De entre todos, el oscilador con mejores características en cuanto a estabilidad de frecuencia y coste son los basados en resonador de cristal de cuarzo, mientras que los que requieren menor consumo son los RC. Además éstos últimos es posible integrarlos dentro de un chip, lo que contribuye a minimizar el tamaño.

El uso de sintetizadores de frecuencia basados en el empleo de lazos enganchados en fase (PLL) permite disponer de un conjunto discreto de frecuencias con gran precisión y estabilidad, donde la selección de la frecuencia se realiza digitalmente.

NOTA: La frecuencia del oscilador normalmente no es la frecuencia a la que se ejecutan las instrucciones. Por ejemplo, en la familia PIC, cada instrucción necesita para ejecutarse 4 ciclos de reloj.

Modulo de alimentación (Power)

El módulo de energía (power) se encarga de generar las diferentes tensiones y corrientes necesarias para alimentar los componentes activos que forman el SE. Lo normal es el empleo de baterías para los dispositivos portátiles y fuentes de alimentación (convertor AC/DC) para los sistemas que disponen de acceso a la red de energía eléctrica.

Cuando son necesarias dos o mas tensiones de valor especifico, mediante el empleo de conversores DC/DC se pueden obtener ésta a partir de una única tensión de entrada generada por una fuente o batería.

Los valores típicos mas empleados para alimentar los sistemas embebidos son 5 Vdc, 9 Vdc, 12 Vdc y 24 Vdc.

Dado que la conexión de la energía es clave en muchas aplicaciones y ésta a su vez es fuente de ruido e interferencias, a veces es necesario dotarla de filtros, circuitos integrados supervisores de alimentación, protectores de sobretensión, etc.

El consumo de energía puede ser determinante en el desarrollo de algunos SSEE, que necesariamente se alimentan con baterías y a las que no es posible recargar de forma continua. En aplicaciones donde es imposible su sustitución, la vida del SE suele estar limitada por la de las baterías, tal es el caso de los satélites artificiales, dotados de paneles solares y baterías.

1.2.2.- Generalidades sobre dispositivos CPU

La arquitectura de los dispositivos empleados en los SSEE ha evolucionado tremendamente en los últimos años, con el objetivo de conseguir mayor capacidad de procesamiento, de manejo de dispositivos, menor consumo, etc.

Por otro lado, si bien hace unos años existía una diferencia nítida entre el concepto de microprocesador y de microcontrolador, en la actualidad, los sistemas fabricados en un solo chip han adquirido tal grado de complejidad que es muy difícil separar ambos conceptos.

Ya se ha comentado que un microprocesador lo definimos como la implementación en forma de circuito integrado (IC) de una Unidad Central de Proceso (CPU) junto con los buses de interconexión (el bus de control, el bus de direcciones y el bus de datos) mientras que un microcontrolador lo definimos como la implementación, dentro de un único chip, del microprocesador, la memoria y subsistemas de E/S.

También se han desarrollado los procesadores digitales de señal (DSP) que habitualmente se fabrican bajo un formato semejante al de un microcontrolador (CPU+Memoria+Periféricos de E/S en un solo chip) pero con una arquitectura especialmente diseñada para realizar las tareas más habituales en procesamiento digital de señales, de forma rápida.

Sin embargo existe una amplia gama de dispositivos pertenecientes a estos tres subconjuntos, cada vez con la inclusión de más funcionalidades, que hace a esta división cada vez más difusa.

La idea de los microcontroladores fue realizar dispositivos muy sencillos con muy poco hardware adicional, dedicados a tareas de control de dispositivos. Es por esto que los subsistemas de E/S debían ser muy completos mientras que las necesidades de memoria eran muy escasas. Se buscaba, a su vez, bajo coste y trabajo en tiempo real.

El diseño del SE consistirá en un módulo (PCB) que contiene varios circuitos integrados (chip's) interconectados entre sí y con el resto de componentes electrónicos pasivos y activos que se definen en el circuito eléctrico (esquemático) del SE. En general, un SE simple contará con un microprocesador, memoria, unos pocos periféricos de E/S y un programa dedicado a una aplicación concreta almacenado permanentemente en la memoria.

La arquitectura en la que se basan la inmensa mayoría de micros es del tipo Harvard con juegos de instrucciones reducidos (RISC), pero existen notables diferencias entre cada familia de éstos.

Podemos resumir como principales características de las actuales familias de microprocesadores las siguientes:

- Uso de CPUs con arquitecturas de 8, 16, 32 y 64 bits.
- Gran cantidad de periféricos de E/S integrados.
- Incluyen memoria dentro del chip y necesidad de manejo de gran cantidad de memoria externa.
- Empleo de sistema operativo.
- Sistemas de desarrollo basados en lenguajes de alto nivel tal como el C.
- Amplia gama de chips y gran variación en el coste.
- Se insertan en placas base, de tamaño cada vez mas compacto, que incorporan una amplia gama de chips dedicados a las distintas funciones previstas.

También, las principales características actuales de los microcontroladores las podemos resumir en:

- Uso de CPUs con arquitecturas harvard de 2, 4, 8, 16 o 32 bits.
- Gran cantidad de periféricos de E/S integrados.
- Necesidad de poca memoria y generalmente no posibilidad de manejar memoria externa.
- No empleo de sistema operativo.
- Sistemas de desarrollo típicos basados en ANSI C.
- Bajo coste del chip.
- La placa (PCB) en la que se insertan suele ser sencilla en comparación con un formato PC, pues a veces solo contiene este único chip.

En el caso de los DSPs, las podemos resumir en:

- Uso de CPUs con arquitecturas harvard modificadas de 16 y 32 bits. Versiones de coma fija y coma flotante.
- Gran cantidad de periféricos de E/S integrados, especializados en la transferencia en tiempo real de grandes volúmenes de datos.
- Incluyen gran cantidad de memoria dentro del chip.
- No empleo de sistema operativo.
- Sistemas de desarrollo típicos basados en ANSI C.
- Coste del chip medio.

- La placa base en la que se incorporan (PCB) suele tener una complejidad media y a veces forma parte de un sistema superior, como por ejemplo, la tarjeta de sonido de un PC.

Empleo de un PC embebido como alternativa

Las opciones de diseño para un SE cada vez son mayores debido a los imparable avances tecnológicos en el sector de la electrónica y microelectrónica.

Así, cada vez el grado de integración de los dispositivos semiconductores, pasivos y electromecánicos es mayor y lo que hasta no hace mucho tiempo era impensable hoy en día es una realidad: Es posible incorporar como SE dentro de un determinado host a todo un modulo tipo PC en formato compacto, tipo Single Board Computer (SBC) dentro de determinadas aplicaciones que necesitan de una mayor complejidad . Existen en el mercado una amplia oferta de SBCs basados en la familia de procesadores de Intel: i386, i486, Pentium, etc. a un coste razonable para determinadas aplicaciones. La familia de procesadores ARM también surge como una opción específica dirigida al sector de los SSEE portables, tales como las cámaras de fotos, móviles, PDA's, etc.

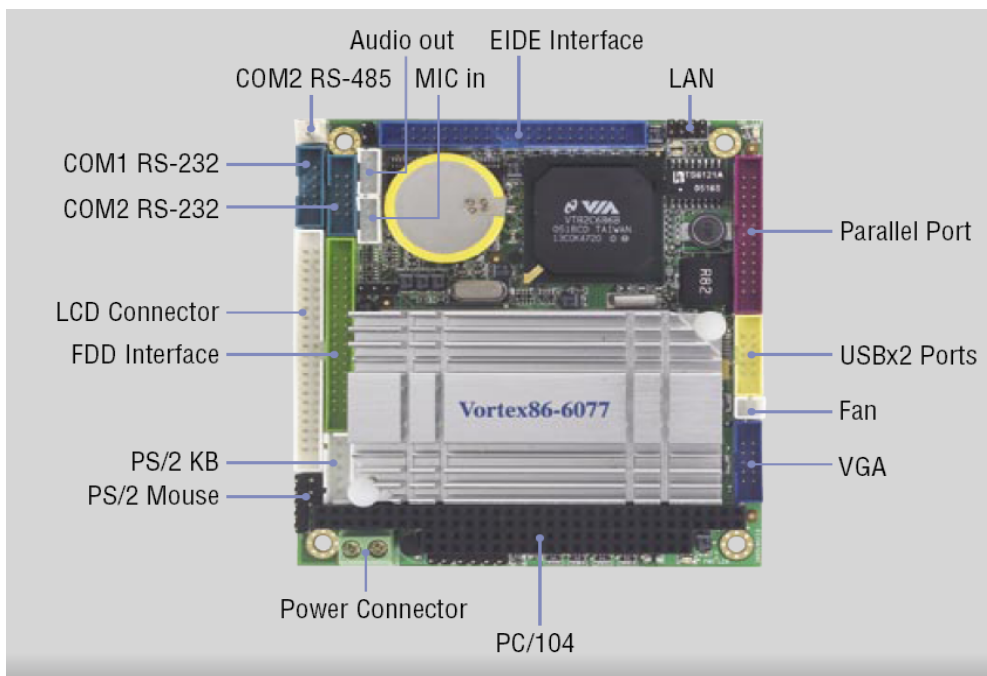


Figura 1.6.- Aspecto de un PC embebido bajo formato estándar PC-104.

Un PC embebido posee una arquitectura basada en éstos elementos básicos:

- **Microprocesador:** Encargado de realizar las operaciones de cálculo principales del sistema. Ejecuta código para realizar una determinada tarea y dirige el funcionamiento de los demás elementos que le rodean.

- **Memoria RAM:** Almacena el código de los programas que el sistema puede ejecutar así como los datos. Su característica principal es que debe tener un acceso de lectura y escritura lo más rápido posible para que el microprocesador no pierda tiempo en tareas que no son meramente de cálculo. Al ser volátil el sistema requiere de un soporte donde se almacenen los datos incluso sin disponer de alimentación o energía.

- **Memoria Caché:** Más rápida que la principal en la que se almacenan los datos y el código accedido últimamente. Dado que el sistema realiza microtareas, muchas veces repetitivas, la caché consigue ahorrar tiempo ya que no hará falta ir a memoria principal si el dato o la instrucción ya se encuentra en la caché.

- **Memoria No volátil.** Habitualmente conocida como 'Disco duro', en él la información no es volátil y además puede conseguir capacidades muy elevadas. A diferencia de la memoria RAM que es de estado sólido éste suele ser magnético en aplicaciones tipo PC pero su excesivo tamaño y falta de robustez mecánica lo suele hacer inviable para PCs embebidos. Los avances tecnológicos, una vez más, han conseguido resolver el problema desarrollando discos de estado sólido. Existen en el mercado varias soluciones de esta clase (DiskOnChip, CompactFlash, IDE Flash Drive, etc.) con capacidades suficientes para la mayoría de sistemas embebidos (hasta más de 20 GB).

- **BIOS-ROM:** Basic Input & Output System, sistema básico de entrada y salida) es código que es necesario para inicializar el ordenador y para poner en comunicación los distintos elementos de la placa madre. La ROM (Read Only Memory, memoria de sólo lectura no volátil) suele ser un chip donde se encuentra el código BIOS.

- **CMOS-RAM:** Es un chip de memoria de lectura y escritura alimentado con una pila donde se almacena el tipo y ubicación de los dispositivos conectados a la placa madre (disco duro, puertos de entrada y salida, etc.). Además contiene un reloj en permanente funcionamiento que ofrece al sistema la fecha y la hora.

- **Chip Set:** Es un chip que se encarga de controlar las interrupciones dirigidas al microprocesador, el acceso directo a memoria (DMA) y al bus ISA, además de ofrecer temporizadores, etc. Es frecuente encontrar la CMOS-RAM y el reloj de tiempo real en el interior del Chip Set.
- **Entradas** al sistema: pueden existir puertos para ratón, teclado, vídeo en formato digital, comunicaciones serie o paralelo, etc.
- **Salidas** del sistema: puertos de vídeo para monitor , pantallas de cristal líquido, altavoces, comunicaciones serie o paralelo, etc.
- **Ranuras de expansión** para tarjetas de tareas específicas que pueden no venir incorporadas en la placa madre, como pueden ser más puertos de comunicaciones, acceso a red de ordenadores vía LAN (Local Area Network, red de área local) o vía red telefónica: básica, RDSI (Red Digital de Servicios Integrados), ADSL (Asynchronous Digital Subscriber Loop, Lazo Digital Asíncrono del Abonado), etc. Un PC estándar suele tener muchas más ranuras de expansión que un PC embebido. Las ranuras de expansión están asociadas a distintos tipos de bus: VESA, ISA, PCI, NLX (ISA + PCI), etc.

Sistemas en un chip (System On Chip, SoC)

Hoy en día existen en el mercado fabricantes que integran un microprocesador y los elementos controladores de los dispositivos fundamentales de entrada y salida en un mismo chip, pensando en las necesidades de los sistemas embebidos (bajo coste, pequeño tamaño, entradas y salidas específicas,...). Su capacidad de proceso suele ser inferior a los procesadores de propósito general pero cumplen con su cometido ya que los sistemas donde se ubican no requieren tanta potencia. Los principales fabricantes son ST Microelectronics (familia de chips STPC), National (familia Geode), Motorola (familia ColdFire) e Intel.

En cuanto a los sistemas operativos necesarios para que un sistema basado en microprocesador pueda funcionar y ejecutar programas suelen ser específicos para los sistemas embebidos. Así nos encontramos con sistemas operativos de bajos requisitos de memoria, posibilidad de ejecución de aplicaciones de tiempo real,

modulares (inclusión sólo de los elementos necesarios del sistema operativo para el sistema embebido concreto), etc. Los más conocidos en la actualidad son Windows CE, QNX y VxWorks de WindRiver.

1.3.- Visión hardware/software de los sistemas embebidos

1.3.1.- Concepción global del sistema.

La concepción del sistema en su globalidad, parte de una idea de una persona, e un equipo de diseño o de un encargo a medida realizado para un 'cliente' que desea resolver una determinada necesidad. Podemos establecer un serie de tareas previas que culminarán en la elaboración de un anteproyecto, en base al cual se tomarán las decisiones de seguir adelante o descartar el proceso:

1. Determinación de los requisitos globales del sistema.
2. Selección del microprocesador, microcontrolador o DSP mas adecuado.
3. Selección de la tecnología de fabricación mas adecuada.
4. Elección de la memoria y del sistema operativo si procede.
5. Determinación de las entradas-salidas, comunicaciones, etc.
6. Determinación de las necesidades de homologación en función de la aplicación.
7. Seleccionar el equipo humano mas adecuado para su desarrollo, selección de proveedores hardware y software, etc.
8. Realización de un **anteproyecto**, lo mas detallado posible que nos permita realizar una evaluación sobre la viabilidad técnica y económica del sistema así como generar un presupuesto de costes lo mas veraz posible.

1.3.2.- Fases típicas de diseño.

Una vez tomada la decisión de proceder a la fabricación del SE, vamos a describir brevemente las fases típicas de diseño en cualquier sistema electrónico, sea éste embebido o no lo sea. Desde la primera fase, conocida como diseño previo del sistema, hasta la última, que consiste en la decisión del producto final, se debe de cuidar el orden de ejecución de las tareas y solapar todas las que sea posible de modo que se obtenga un producto fiable en el menor tiempo posible de puesta en el mercado (time-to-market).

En la figura 1.7 se muestra un diagrama simplificado con las fases a seguir y que comentamos a continuación:

1. Diseño inicial del sistema que incluye toda una serie de tareas que acabarán en la elaboración de un esquema eléctrico del mismo y en un diseño de necesidades software.
2. A partir del esquemático y de la forma física de cada uno de los componentes que intervienen, elaborar un diseño hardware del mismo. Esta tarea incluye el posicionamiento de cada uno de los componentes y el ruteado de las pistas de cobre que realizarán las necesarias interconexiones entre los pines de los componentes, generando un prototipo de PCB, sobre el que se realiza el montaje o ensamblado de todos y cada uno de los dispositivos mediante el procedimiento de soldadura mas adecuado. Termina en un prototipo hardware.
3. Desarrollo del prototipo de software con la programación inicial del micro o de los micros que formen parte del SE.
4. Integración hardware/software mediante el volcado o programación en el circuito de los micros. Se dispondrá así del primer prototipo listo para proceder a su testeo y depuración.
5. Pruebas y depuración del software y hardware mediante el empleo de prototipos hasta llegar a la versión final. Si se detectan errores en el hardware será necesario proceder a rediseñar la placa y volver a comenzar el proceso. Si los errores son de software, el proceso es similar, solo que menos costoso en cuanto a materiales que no en cuanto a horas de ingeniería.
6. Producto Final. Tras el resultado satisfactorio en todas las pruebas se conseguirá el producto final. En el caso de previsiones de fabricación masiva será necesario fabricar pre-series y probarlas para así minimizar los imprevistos de cara a la fabricación en serie de altas cantidades.

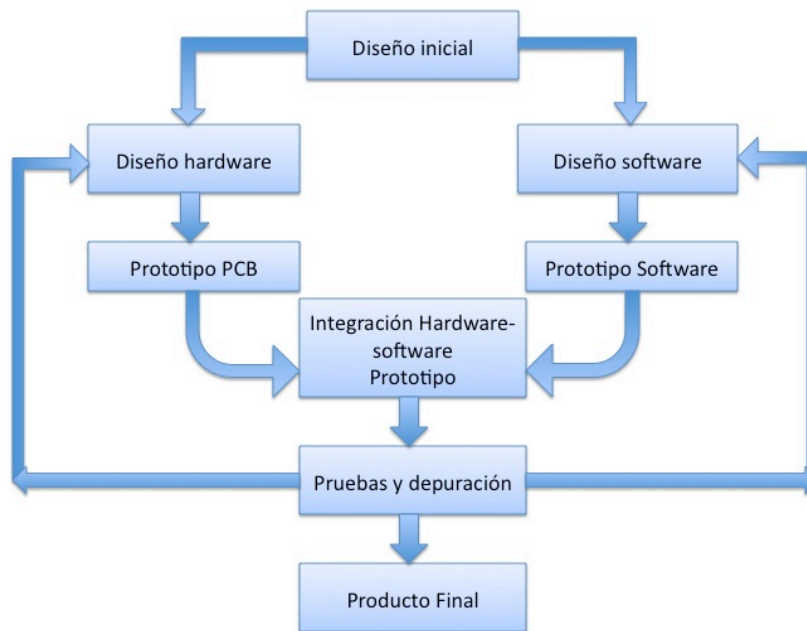


Figura 1.7.- Fases típicas de diseño de un SE.

1.3.3.- Otros aspectos a tener en cuenta en el diseño.

Se comentan a continuación una serie de aspectos que a los que es necesario prestar atención de cara al éxito del proyecto de SE a realizar.

1.3.3.1.- Determinación de los requisitos del sistema.

Una de las partes más importantes del desarrollo de un SE es la elaboración de los requisitos para la aplicación buscada. Cuanto mejor se realice esta fase previa a la creación de un prototipo, menos probables serán los indeseados cambios tanto en hardware como en software. Aún así hay que recordar que el desarrollo del sistema generalmente es un proceso iterativo. Al recorrer los últimos pasos del proyecto puede surgir la necesidad de revisar las primeras fases para conseguir un producto fiable, consistente y que cumpla con todos los requisitos.

Habrá que tener en cuenta una serie de consideraciones técnicas, tales como son:

Definición de los interfaces de control.

- Los tipos de dispositivos que habrá que controlar o con los que se habrá de comunicar, así como las propiedades eléctricas y mecánicas de sus interfaces.
- Una definición general de los requisitos de tipo de memoria (volátil, no volátil, estado sólido, magnética).
- La definición de cómo el sistema interactuará con las personas (monitor, teclado, lector de códigos de barras, etc.).

Definición de la aplicación software.

- Descripción específica de todas las características de la aplicación software.
- Características de depurado.
- Posibilidad de actualización del software.
- Previsión de controladores (drivers) para el funcionamiento del hardware externo.

Requisitos de alimentación.

- Posibilidad de alimentación mediante batería o pilas.
- Alimentación de seguridad para condiciones de caída de la alimentación principal.
- Previsiones generales de consumo del sistema.

Requisitos térmicos del sistema.

- Margen de temperatura nominal.
- Refrigeración por ventilador o por convección.
- Ubicación de los componentes para una efectiva gestión térmica.
- Condiciones ambientales externas para determinar las necesidades de control térmico.
- Tiempo de vida esperado del sistema frente a las necesidades de control térmico.

Diseño de la forma y tamaño.

- Restricciones en cuanto a dimensiones.
- Robustez del producto.

Definición de los subsistemas principales.

- Los subsistemas principales con suficiente complejidad deben tener una definición similar a la del sistema completo para evitar problemas o elevados costes de integración.

Necesidades de rendimiento.

- Necesidades de velocidad de procesado.
- Capacidad de manejar interfaces gráficas.
- Capacidad de ejecutar software exterior.
- Capacidad de comunicarse con otros interfaces de alta velocidad.

1.3.3.2.- Consideraciones a nivel de proyecto.

Previsión del tiempo que el producto vivirá en el mercado.

- Disponibilidad de componentes.
- Soporte y mantenimiento del producto vendido.

Estimación del tiempo de desarrollo.

- Ventana temporal de la oportunidad de mercado.
- Velocidad para sacar los primeros prototipos.
- Disponibilidad de encontrar desarrolladores con experiencia.

Nivel de experiencia de diseño o desarrollo.

- Experiencia de diseño de hardware.
- Familiarización con los entornos de programación.

Actualizaciones del software.

- Número esperado de actualizaciones.
- Previsión del crecimiento del tamaño del programa.
- Posibilidad de realizar actualizaciones remotas del programa.

Efectividad de la plataforma de desarrollo.

- Tiempo estimado para la finalización del primer prototipo.
- Disponibilidad de herramientas.
- Capacidad de utilización o integración de periféricos.
- Facilidad de uso.
- Mantenimiento.
 - Procedimiento de actualización del hardware y del software.
 - Mantenimiento o sustitución de componentes.
 - Mantenimiento o sustitución de subsistemas.
 - Mantenimiento térmico del sistema.
 - Actualizaciones del hardware.
 - Estudio de las necesidades de actualización.
 - Expansión y modularidad del diseño.

1.3.3.3.- Consideraciones a nivel de diseño del SE.

Los fabricantes de semiconductores proporcionan los microprocesadores, microcontroladores y DSPs en forma de chip bajo una amplia gama de encapsulados, en general cada vez con dimensiones mas reducidas.

El hardware se diseña en dos fases: creación de los esquemas y creación del layout de la placa de circuito impreso (PCB).

Elaboración de los esquemáticos.

El diseño electrónico del sistema se realiza mediante unos diagramas llamados esquemas, los cuales representan los componentes mediante símbolos y sus interconexiones mediante líneas, de modo que sea fácil o digerible de visualizar. Estos diagramas se realizan con herramientas software CAD (Computer Aided Design, Diseño Asistido por Ordenador) tales como Orcad, Altium o VeriBest de Intergraph.

Unos buenos esquemas deben incluir información extra requerida para entender el funcionamiento del sistema. Entre otras cosas deben figurar nombres representativos en las líneas que representan buses o señales entre dos componentes. Todos los componentes que se vayan a colocar en la placa de circuito impreso, que se puede fabricar gracias al layout cuya realización se describe en el siguiente punto, deben tener su representación en los esquemas mediante símbolos, normalmente rectangulares con sus pines alrededor.

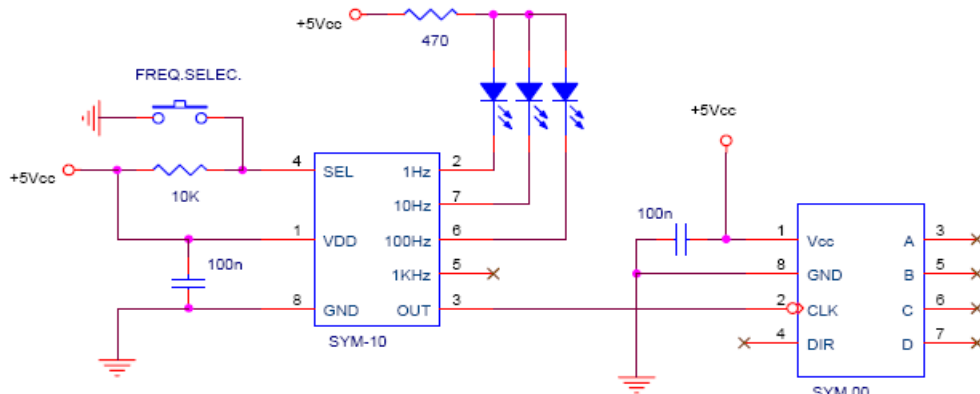


Figura 1.8.- Ejemplo de esquemático de un SE.

El diseñador de los esquemas debe consultar la documentación técnica o especificaciones de los componentes utilizados de tal forma que se realice la conexión entre ellos de una manera correcta. Se deben cuidar aspectos como la temporización en las comunicaciones entre ellos y las cargas (esto es, que una salida proporcione suficiente corriente eléctrica para las entradas a las que llega). Sobre todo a altas frecuencias de trabajo del sistema (por encima de 50 MHz), hay que cuidar la adaptación entre componentes, esto es, que no se produzcan reflexiones de ondas eléctricas en las interconexiones entre componentes ya que harán imposible la transferencia de información legible.

Creación del layout de la PCB.

El layout consiste en una representación exacta de las huellas “footprint” de los componentes y de las pistas que unen sus pines o terminales. Se entiende por “footprint” o huella de un componente los puntos de cobre que necesita en la placa base para poder ser soldado y fijado, mientras que las pistas son las líneas de cobre de una cierta anchura que permite la propagación eléctrica de las señales entre los componentes. La placa está formada por una lámina de fibra de vidrio o similar con

varias capas de cobre (2 exteriores y posibles internas, a modo de sándwich). Mediante los gerbers, que son unos dibujos que representan el layout, el fabricante de PCBs puede construir la placa con las huellas y pistas necesarias. Tras ello se pueden soldar los componentes.

El layout también se realiza mediante una herramienta CAD, integrada con la de creación de esquemas. Así los esquemas dirigen la creación del layout estableciendo cuales deben ser las uniones entre los componentes, ahora en forma de pistas que recorrerán el camino necesario para unir dos puntos. Esto normalmente se hace no sin esquivar un cierto número de obstáculos (otras pistas, componentes, etc.). Entre los aspectos a cuidar en la creación de la PCB está la posibilidad de elegir encapsulados de diferentes formas o tipos para un mismo componente, así como la compatibilidad electromagnética de la placa (esto es, que las señales de alta frecuencia no interfieran con otras de la misma placa ni con otras del exterior). Mediante la herramienta de autoenrutado o autorouter el programa CAD puede dibujar todas las pistas sin más que elegir el sitio donde irá cada componente (ubicación o placement). Dependiendo de la densidad de pistas pueden ser necesarias más o menos capas de pistas de la PCB.

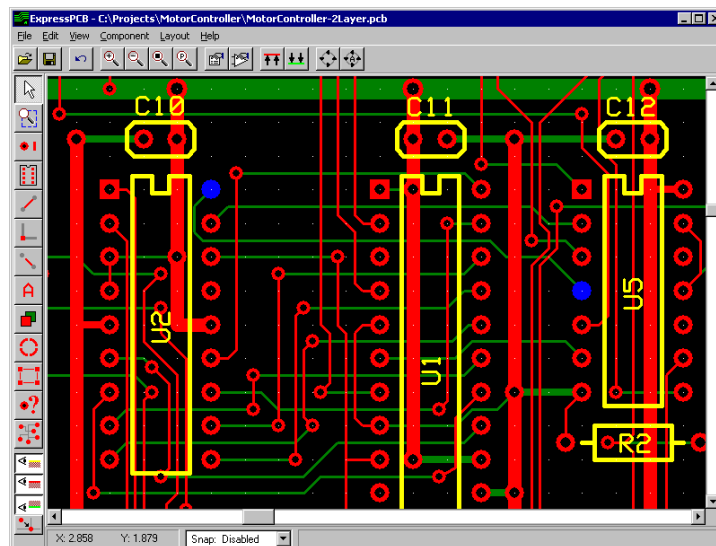


Figura 1.9.- Programa CAE para la generación del layout de un PCB.

Existe una jerarquía de diseño de un sistema embebido, que podemos expresar, según nivel de integración en los siguientes puntos:

1. Nivel de chip. Empleo de sistemas CAE para diseño electrónico para proceder a su diseño y simulación.

2. Nivel de tarjeta (PCB). Sobre un sustrato aislante con unas dimensiones físicas determinadas se procede a diseñar el layout de la tarjeta. Posteriormente se sueldan los dispositivos. Las tecnologías de diseño suelen definirse como:

- Convencional.
- Montaje superficial, SMT
- Híbrida capa fina y capa gruesa.

A su vez, según la forma de interconectar los dispositivos podemos hablar de

- Simple capa.
- Doble capa.
- Multicapa.

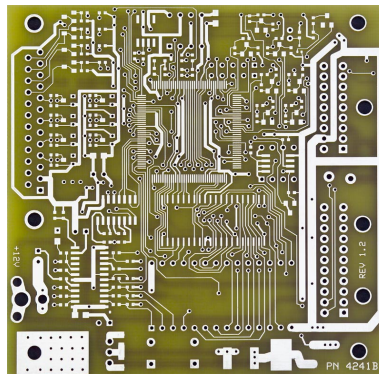


Figura 1.10.- Ejemplo de prototipo inicial PCB.

Montaje de los componentes sobre la PCB.

El prototipo es un equipo hardware que tiene partes idénticas a las del producto final pero no está completo. O incluso puede ser diferente al producto final pero con fuertes similitudes. El caso es que sirva para ir probando módulos software y con ello también el hardware, que también debe ser depurado. Este prototipo se puede construir adquiriendo los componentes necesarios y fabricando una PCB o bien se puede comprar a un tercero una placa de evaluación del microprocesador utilizado de modo que se ahorre bastante tiempo. Nos estamos refiriendo al emulador hardware que se mencionó en el anterior apartado de diseño del software. Para este último caso lo que se hace es probar el software, pero de cualquier forma el hardware propio habrá que probarlo tarde o temprano, en última instancia en la integración del sistema.

Integración del sistema.

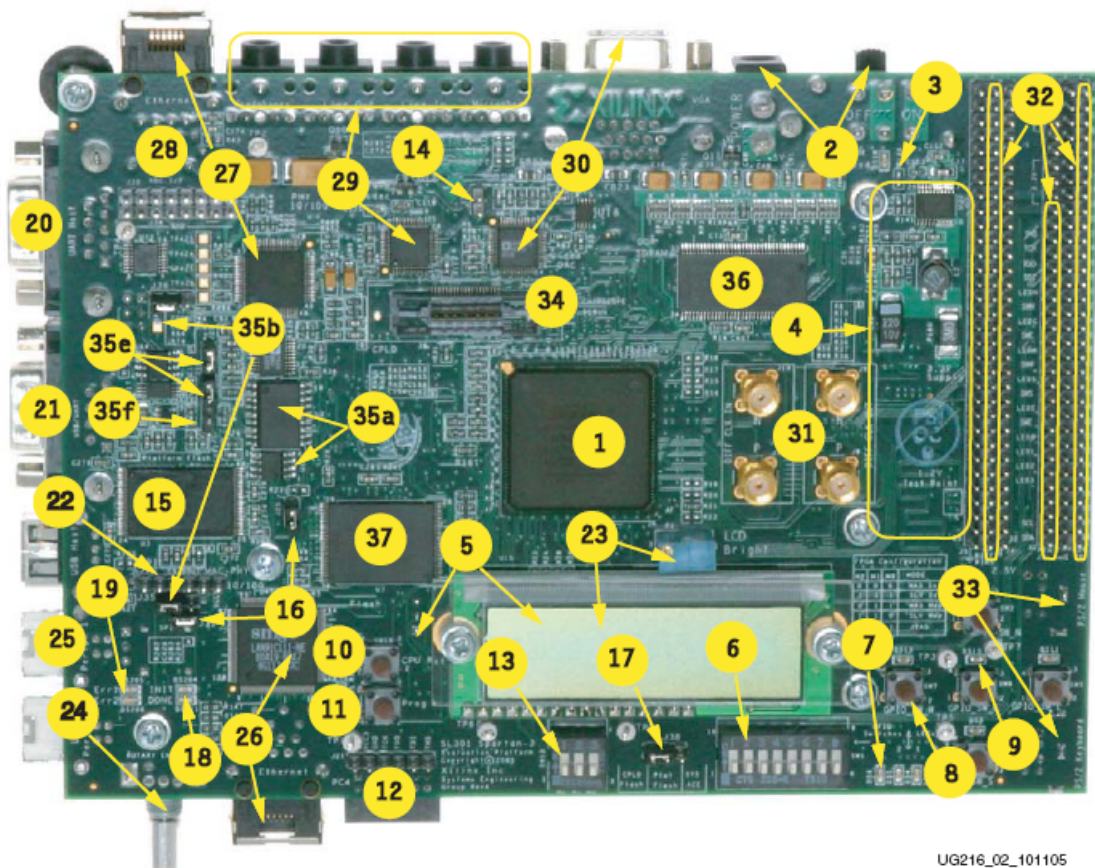
Esta es la primera vez que se prueba el sistema completo (tanto software como hardware a la vez). En realidad se debe estar probando el hardware, ya que el software debe estar correcto tras las pruebas sobre emuladores hardware o software cuyo comportamiento es casi idéntico al que debe ofrecer el hardware propio.

La integración es “el momento de la verdad”. Es un paso de elevada ansiedad en el proceso de desarrollo. Aquí es donde se encuentran los errores costosos y que deben ser remediados. Desafortunadamente, la integración del sistema normalmente ocurre cerca del final del ciclo de vida del proyecto, cuando los errores alcanza el máximo coste para su reparación. Dependiendo del tipo de sistema embebido, el coste añadido para corregir un error aumenta de 2 a 10 veces por cada fase de vida. Otro problema de los errores encontrados en la integración del sistema es que puede retrasar el plazo de entrega del proyecto.

Existen numerosas formas de reducir el tiempo de integración. Una de las mejores consiste en crear un prototipo hardware dentro de la fase de diseño hardware y permitir pruebas de partes del software que se va desarrollando. Además el desarrollo del software se acelera dado que no hay que perder tiempo implementando software que simule el aún inexistente hardware.

Pruebas, depuración y producto final.

La prueba final consiste en la comprobación de que el sistema cumple con todos los requisitos esperados. Las pruebas pueden ser Alfa si se realizan en la propia empresa de desarrollo del sistema o pueden ser Beta si se realizan con el cliente o usuario en el lugar que él designe. Tras esta fase sólo queda la política de mantenimiento (principalmente software) que se establezca entre ambas partes.



UG216_02_101105

Figura 1.11.- Ejemplo de producto final.

1.3.3.4.- Otras consideraciones acerca del diseño del software.

El software que hay que realizar para que el sistema cumpla su funcionalidad se suele empezar a desarrollar a la par que el diseño del hardware. Una vez que el hardware y el software están concluidos se podrá efectuar la integración del sistema. Mientras que llega ese momento, el software se desarrolla probándolo y depurándolo con emuladores software o hardware que proporcionan los fabricantes de los micros empleados u otras empresas. Si se utiliza una arquitectura tipo PC se puede simplificar el desarrollo del software dado que existen en el mercado números módulos ya implementados que se pueden comprar. Dónde puede haber mayor complejidad es en los requisitos de tiempo real que pueda tener el sistema.

El software se empieza a desarrollar por una fase inicial de captura de requisitos de usuario (lo que el usuario final espera obtener del sistema). Tras ello se establecen los requisitos de sistema (lo que el sistema debe tener para poder cumplir con los requisitos de usuario). Los requisitos de sistema son más técnicos que los de usuario

y por lo general en mayor número, ya que para cubrir un requisito de usuario pueden ser necesarios más de un requisito de sistema.

Hay que destacar también la importancia de documentar el software, al igual que el hardware. En el código hay que añadir líneas de comentarios describiendo la funcionalidad o utilidad de las variables y funciones, e incluso explicaciones de trozos de código. También se debe generar un documento con la descripción del software. Éste queda prácticamente constituido con los diagramas que se van generando con la metodología UML.