



Escola Universitària  
Politécnica de Mataró

**Enginyeria Tècnica Industrial: Especialitat Electrònica Industrial**

**METODOLOGIA DE DESENVOLUPAMENT  
D'APLICACIONS PC EMBEDDED**

**FIDEL MIRAS LLOPART  
JULIÁN HERRILLO TELLO**

PRIMAVERA 2009



## **Resum**

Aquest projecte tracta sobre el procés de disseny i creació d'aplicacions embedded, especialment aquelles implementades sobre plataformes PC. Des d'un punt de vista conceptual, es donen a conèixer els principals aspectes que conformen un sistema embedded: desde les plataformes hardware, les comunicacions i els sistemes operatius fins a la metodologia de creació d'aplicacions.

En el marc de desenvolupament s'explica de forma detallada la creació de la imatge executable d'un sistema operatiu i la seva transferència a un dispositiu embedded, mitjançant Windows Embedded CE 6.0.

El resultat final del projecte s'obté amb la creació i transferència d'una imatge de Windows Embedded CE per al seu funcionament com a sistema operatiu al PC industrial Advantech PCM-9575.

## **Resumen**

Este proyecto trata sobre el proceso de diseño y creación de aplicaciones embedded, especialmente aquellas implementadas sobre plataformas PC. Desde un punto de vista conceptual, se dan a conocer los principales aspectos que conforman un sistema embedded: desde las plataformas hardware, las comunicaciones y los sistemas operativos hasta la metodología de creación de aplicaciones.

En el marco de desarrollo se explica de forma detallada la creación de la imagen ejecutable de un sistema operativo y su transferencia a un dispositivo embedded, mediante Windows Embedded CE 6.0.

El resultado final del proyecto se obtiene con la creación y transferencia de una imagen de Windows Embedded CE para suyo funcionamiento como sistema operativo en el PC industrial Advantech PCM-9575 .



## **Abstract**

The primary goal of this project is the design and development of embedded applications, specifically those running over PC platforms. From a conceptual point of view, the main aspects that constitute an embedded system are presented: from hardware platforms, communication protocols and operative systems to the application development methodologies.

The creation of a runtime operative system image, as well as its transmission to an embedded device using Windows Embedded CE 6.0, is described in more detail within the development framework.

Finally, an example of this approach is presented. The development and transmission of a runtime Windows Embedded CE image to an Industrial PC Advantech PCM-9575, and all the documentation related, will represent a useful tool to present this kind of methodologies within the University context.



## ÍNDIX DE CONTINGUTS

<b>1</b>	<b>Objecte del projecte</b>	<b>2</b>
<b>2</b>	<b>Abast del projecte</b>	<b>3</b>
<b>3</b>	<b>Marc Conceptual</b>	<b>5</b>
3.1	El concepte embedded	5
3.2	El PC i la planta industrial	7
3.3	Els sistemes embedded	9
3.3.1	Plataformes PC embedded	10
3.4	Els estàndards embedded PC/104	12
3.4.1	PC/104	13
3.4.2	PC/104-Plus	14
3.4.3	PCI-104	15
3.4.4	PCI/104-Express	16
3.4.5	Evolució de l'arquitectura PC/104	17
3.4.6	Altres formats PC/104	19
3.4.7	Avantatges del PC/104	20
3.5	El processador	22
3.6	Comunicacions embedded	26
3.6.1	Comunicacions sèrie	27
3.6.1.1	RS-232	29
3.6.1.2	Wireless LAN	31
3.6.2	Comunicacions paral·lel	33
3.6.2.1	Dispositius gràfics d'E/S	34

---

3.6.3	Ethernet	35
3.6.3.1	Per què utilitzar Ethernet?	36
3.6.3.2	Ethernet als sistemes embedded	37
3.6.4	Funcionament de les E/S	39
3.6.5	Board Buses	40
3.7	El sistema operatiu	41
3.7.1	Què és el sistema operatiu?	41
3.7.2	Estructura d'un sistema operatiu	42
3.7.3	Sistemes operatius de temps real	43
3.7.4	Factors de selecció d'un RTOS	45
3.7.5	Principals sistemes operatius embedded	46
3.7.5.1	Linux	47
3.7.5.2	VxWorks	48
3.7.5.3	QNX	49
3.7.5.4	Windows Embedded CE	50
<b>4</b>	<b>Marc de desenvolupament</b>	<b>53</b>
4.1	Desenvolupament d'aplicacions embedded	53
4.1.1	Desenvolupament entre plataformes	53
4.1.2	Emmagatzematge i actualització del software	54
4.1.3	Creació d'un entorn de desenvolupament PC embedded	55
4.2	Windows Embedded CE 6.0	57
4.2.1	Introducció	57
4.2.2	Evolució de Windows Embedded CE	58



---

4.2.3	Instal·lació de Windows Embedded CE 6.0	59
4.2.4	Terminologia de Windows Embedded CE	61
4.2.5	Entorn de disseny i eines de desenvolupament	62
4.2.6	Board Support Packages	64
4.2.7	Creació d'una imatge executable personalitzada	70
4.2.8	Connexió amb el dispositiu target	91
4.2.9	Eines i entorns de depuració	102
4.3	Creació de la imatge executable de SO per al PCM-9575	108
4.3.1	Introducció	108
4.3.2	Característiques principals del PCM-9575	109
4.3.3	Configuració del PCM-9575	110
4.3.4	Transferència de la imatge al PCM-9575	111
<b>5</b>	<b>Conclusions</b>	<b>115</b>
<b>6</b>	<b>Glossari</b>	<b>117</b>
<b>7</b>	<b>Bibliografia</b>	<b>121</b>
7.1	Llibres	108
7.2	Revistes i catàlegs	108
7.3	Pàgines web	109

## ÍNDEX DE FIGURES

<b>1</b>	<b>Objecte del projecte</b> _____	<b>1</b>
	Sense figures	
<b>2</b>	<b>Abast del projecte</b> _____	<b>2</b>
	Sense figures	
<b>3</b>	<b>Marc Conceptual</b> _____	<b>3</b>
	Figura 3.1 Diagrama funcional d'un sistema embedded _____	5
	Figura 3.2 Model d'un sistema embedded _____	6
	Figura 3.3 Backplane PXE-19S2 _____	7
	Figura 3.4 Targeta 104- 1462CLDN _____	8
	Figura 3.5 Dimensions bàsiques i ampliació de mòduls PC/104 _____	10
	Figura 3.6 Components PC/104 com aplicacions _____	10
	Figura 3.7 Dimensions bàsiques de PC/104-Plus _____	11
	Figura 3.8 Evolució de l'arquitectura PC/104 _____	12
	Figura 3.9 Targeta EPIC Nano 7240 _____	13
	Figura 3.10 Targeta EBX PCM-9587 _____	13
	Figura 3.11 Targeta Encore 400 _____	15
	Figura 3.12 Nivells OSI i TCP/IP _____	20
	Figura 3.13 Transmissió Simplex _____	23
	Figura 3.14 Transmissió Half-duplex _____	23
	Figura 3.15 Transmissió Duplex _____	23
	Figura 3.16 Connexió de teletips i mòdems per RS-232 _____	23
	Figura 3.17 Pinatge del DB-25 per a RS-232 _____	26
	Figura 3.18 Pinatge del RJ-45 per a RS-232 _____	27

---

Figura 3.19	Nivells de voltatge de RS-422_____	28
Figura 3.20	Transceiver MAX 3483_____	30
Figura 3.21	Transceiver MAX 3491_____	30
Figura 3.22	Targeta PCI ASUS WL-1386_____	31
Figura 3.23	Angles de visió de IrDA_____	32
Figura 3.24	Components gràfics d'un sistema embedded_____	33
Figura 3.25	Targeta ETHERNUT 1_____	35
Figura 3.26	Capes d'un sistema operatiu genèric_____	41
<b>4</b>	<b>Marc de desenvolupament_____</b>	<b>53</b>
Figura 4.1	Model típic de desenvolupament cross-platform_____	54
Figura 4.2	Vista general de Visual Studio 2005_____	63
Figura 4.3	Finestra de Clone BSP_____	66
Figura 4.4	Finestra de Catalog Ítems View_____	67
Figura 4.5	Carpeta BSP de Catalog Ítems View_____	68
Figura 4.6	Finestra inicial de l'assistent d'un nou projecte_____	70
Figura 4.7	Pantalla inicial de l'assistent OS Design Wizard_____	71
Figura 4.8	Selecció de BSPs_____	72
Figura 4.9	Selecció de la plantilla de disseny_____	72
Figura 4.10	Selecció de les variants de plantilles_____	73
Figura 4.11	Selecció d'aplicacions multimèdia_____	74
Figura 4.12	Selecció d'ítems per a les comunicacions _____	74
Figura 4.13	Administrador de configuracions _____	77

---

Figura 4.14	Estructura d'arxius d'un projecte	78
Figura 4.15	Filtre d'opcions	78
Figura 4.16	Menú Ver	79
Figura 4.17	Menú Proyecto	79
Figura 4.18	Ítems del catàleg	80
Figura 4.19	Propietats del disseny del projecte	81
Figura 4.20	Propietats de configuració del disseny de SO	82
Figura 4.21	Opcions locals del disseny de SO	82
Figura 4.22	Opcions constructives del disseny de SO	83
Figura 4.23	Especificació de variables d'entorn addicionals	85
Figura 4.24	Accions constructives personalitzades	85
Figura 4.25	Variable d'entorn	87
Figura 4.26	Finestra de sortida de Visual Studio 2005	87
Figura 4.27	Resultats de sortida	88
Figura 4.28	Finestra de variables d'entorn	90
Figura 4.29	Menú Generar de VS2005	90
Figura 4.30	Target Device Connectivity Options	92
Figura 4.31	Entrada d'un nou dispositiu	93
Figura 4.32	Configuració d'un nou dispositiu	93
Figura 4.33	Propietats de l'emulador	94

---

Figura 4.34	Finestra de progrés de la descàrrega	95
Figura 4.35	Emulador de dispositius de VS2005	95
Figura 4.36	Aplicació makeimagedisk de VS2005	97
Figura 4.37	Línia de comandaments de Windows XP	98
Figura 4.38	Esquema de connexió física amb el dispositiu target	99
Figura 4.39	Finestra d'opcions de connectivitat	100
Figura 4.40	Finestra Ethernet Download Settings	101
Figura 4.41	Pantalla principal de Kernel Tracker	107
Figura 4.42	PCM-9575 board	108
Figura 4.43	Ports i característiques del PCM-9575	109
<b>5</b>	<b>Conclusions</b>	<b>115</b>
	Sense figures	
<b>6</b>	<b>Glossari</b>	<b>117</b>
	Sense figures	
<b>7</b>	<b>Bibliografia</b>	<b>121</b>
	Sense figures	

## ÍNDEX DE TAULES

<b>1</b>	<b>Objecte del projecte</b> _____	<b>1</b>
	Sense taules	
<b>2</b>	<b>Abast del projecte</b> _____	<b>2</b>
	Sense taules	
<b>3</b>	<b>Marc Conceptual</b> _____	<b>3</b>
	Taula 3.1 Principals arquitectures de processadors_____	16
	Taula 3.2 Comparativa d'interfícies de comunicació_____	22
	Taula 3.3 Comparativa d'estàndards de comunicació sèrie_____	24
	Taula 3.4 Pins de DB25 per a RS-232_____	26
	Taula 3.5 Pins de DB9 per a RS-232_____	27
	Taula 3.6 Pins de RJ45 per a RS-232_____	27
<b>4</b>	<b>Marc de desenvolupament</b> _____	<b>53</b>
	Sense taules	
<b>5</b>	<b>Conclusions</b> _____	<b>115</b>
	Sense taules	
<b>6</b>	<b>Glossari</b> _____	<b>117</b>
	Sense taules	
<b>7</b>	<b>Bibliografia</b> _____	<b>121</b>
	Sense taules	

# 1 OBJECTE DEL PROJECTE

## 1.1 Introducció

Les primeres passes de la meva carrera professional, clarament associades al món industrial, i el meu interès pel món de la automatització i la informàtica es donen la mà en aquest Projecte Fi de Carrera; que aborda un tema de creixent importància en el món industrial: els sistemes PC embedded.

Durant els meus primers anys professionals, m'he familiaritzat notablement amb els quadres elèctrics de maniobra i els famosos sistemes PLC i SCADA que s'integren dins la maquinària de producció industrial.

No obstant, el creixent interès per part de les empreses de controlar els fluxos d'informació i d'integrar-los dins d'una mateixa xarxa, des de la part corporativa fins a la part productiva; i els avenços tecnològics ocorreguts al camp de la informàtica, han obert un nou horitzó on el concepte embedded ha adquirit una vital importància.

Un sistema embedded és capaç de portar els circuits d'informació i control empresarials a tots els punts de l'entramat organitzacional. Pot ser implementat en un automòbil, una PDA, un computador industrial o un caixer automàtic, amb unes prestacions constructives elementals i una grandària considerablement reduïda. Tot això fa que les previsions del mercat indiquin un augment a la demanda d'aquesta tecnologia, i una interessant vessant per als futurs enginyers.

## 1.2 Motivacions

Les meves principals motivacions són iniciar-me en l'estudi dels sistemes embedded, per a poder portar a terme projectes d'aquesta naturalesa; i que el present projecte pugui esdevenir un punt de partida per a futurs estudiants d'enginyeria, en la iniciació en aquest camp.

## **2 ABAST DEL PROJECTE**

Amb aquest projecte es pretén posar sobre la taula les eines i els coneixements necessaris per a iniciar-se dins el món dels sistemes embedded, concretament d'aquells que es desenvolupen dins una arquitectura PC, establint una sèrie de pautes a l'hora de dissenyar un sistema d'aquesta naturalesa per a que resolgui una aplicació concreta.

El projecte està dividit en dos grans blocs: el marc conceptual i el marc de desenvolupament.

El marc conceptual ofereix la informació necessària per a entendre la naturalesa d'un sistema embedded, les parts que el componen i les plataformes actuals disponibles al mercat que el suporten. Està dividit en diferents apartats, que descriuen els components principals que conformen un sistema embedded: plataformes hardware, comunicacions i els sistemes operatius.

El marc de desenvolupament mostra la metodologia per a la creació d'imatges executables de sistema operatiu per a sistemes embedded. Es pretén donar les pautes inicials necessàries per a poder iniciar un projecte de desenvolupament d'una aplicació embedded amb els requisits necessaris per a cada dispositiu.

El marc de desenvolupament es centra en la metodologia de desenvolupament de plataformes amb Windows Embedded CE. Aquesta metodologia de desenvolupament es porta a terme en una aplicació real basada en la creació d'una imatge de Windows Embedded CE i la seva transferència al PC industrial Advantech PCM-9575.

L'objectiu final del projecte és el funcionament òptim d'una imatge executable personalitzada de Windows Embedded CE com a sistema operatiu del PC industrial Advantech PCM-9575, disponible a les instal·lacions de l'Escola Universitària Politècnica de Mataró.



### 3 – MARC CONCEPTUAL

En aquest capítol 3 s'explica, de forma general, tots els aspectes teòrics previs necessaris per al desenvolupament d'aplicacions embedded. Tot enginyer de disseny de sistemes embedded ha de els conèixer aspectes conceptuals bàsics sobre la tecnologia embedded. Només així pot mesurar de forma precisa l'abast dels seus projectes des d'un punt de vista tan tecnològic com econòmic.

Per a oferir una lectura més còmode i entenedora, el capítol 3 d'aquest projecte està dividit en els diferents subapartats:

- **El concepte embedded:** Definició del concepte embedded i estat de l'art de la tecnologia embedded.
- **Plataformes hardware:** Plataformes hardware que suporten sistemes embedded, amb una especial atenció a l'arquitectura PC/104, molt implantada al mercat actual. Processadors a sistemes embedded, principals famílies de processadors i consells per a l'elecció del processador adequat.
- **Comunicacions:** Aspectes generals sobre comunicacions, amb especial atenció a les interfícies emprades a sistemes embedded. En aquests apartat es descriuen els principals protocols de comunicació sèrie i paral·lel i el paper que juguen a les plataformes embedded.
- **El sistema operatiu:** Explicació conceptual del sistema operatiu i la importància d'aquest en un sistema embedded. Concepte de sistema operatiu de temps real i factors clau en l'elecció d'un sistema operatiu per a un sistema embedded.

Nota: Cal tenir en compte que a la informàtica industrial actual, cada cop hi ha més complicitat entre aspectes hardware, software i de comunicacions. Per tant, la divisió anterior que s'estableix no pot ser totalment estricte i el vincle entre diferents apartats és inevitable.

## **3.1 – EL CONCEPTE EMBEDDED**

### **3.1.1 - El PC i la planta industrial**

Desde principis dels anys 70, l'evolució dels sistemes de control i automatització industrials ha caminat de forma paral·lela a l'evolució dels equips informàtics disponibles per a la realització de tasques concretes. En el control industrial, s'ha anat passant progressivament de la utilització de relés electromecànics, a l'ús de sofisticats computadors industrials, amb la conseqüent reducció de grandària dels equips i l'augment de la flexibilitat dels sistemes.

Les importants millores de l'arquitectura PC en aspectes com la relació preu/prestacions, la dissipació de potència i la reducció de grandària possibiliten l'augment de la disponibilitat de productes PC ja disponibles al mercat amb diferents grandàries, configuracions, nivells de prestacions i preus.

D'altra banda, la pressió en relació als temps de sortida al mercat dels productes és cada cop més alta. Això obliga als enginyers de desenvolupament a oferir solucions ràpides i flexibles en el disseny d'aplicacions.

Tradicionalment, per a volums importants de producció, quan els costos de desenvolupament propi eren inferiors al cost d'adquisició d'un determinat equip, s'optava pel disseny propietari. La tendència actual va en detriment del disseny propietari i s'opta per a l'ús de tecnologies conegudes i disponibles.

La omnipresència de l'arquitectura PC al mercat, amb una major disponibilitat de productes, i les notables millores en prestacions i fiabilitat del hardware, han facilitat l'ús del PC a la planta industrial.

D'altra banda, les dures condicions de treball que imposa l'entorn industrial exigeixen al PC industrial major immunitat a les agressions externes. Apareixen en aquest punt presentacions constructives molt diferents a les del PC de sobretaula (xassís per a muntatge en *rack*, format panell, estació de treball, etc.); i equips amb una major robustesa

(estanquitat, resistència mecànica al xoc i vibracions i major resistència elèctrica a sobretensions i paràsits ).

Aquestes millores han anat acompanyades de la introducció de nous sistemes operatius (capacitats de multitasca i temps real), i de millores importants en les interfícies d'usuari i de comunicació amb altres sistemes. Tot plegat a propiciat la creixent entrada d'aquesta tecnologia en el camp de les aplicacions industrials, on la alternativa PC és ja una realitat.

### 3.1.2 – Sistemes embedded

Embedded és un terme anglès que significa integrat, encastat. En termes tecnològics, un sistema embedded és el resultat de la combinació de certes funcions hardware i software, que realitzen una determinada activitat sense casi intervenció d'un usuari i formant part d'un sistema major. Poden trobar-se a les llars, oficines, plantes de producció, maquinària i automoció, i generalment controlen, protegeixen o monitoritzen alguns aspectes vitals del sistema que els conté. Generalment, un sistema embedded es configura al voltant d'un microprocessador o un microcontrolador, i incorpora els elements continguts a la figura 3.1.

Habitualment, un sistema embedded consta, com a elements bàsics, de CPU, memòria de programa, memòria de dades i E/S i, generalment, no incorpora recursos d'emmagatzament de massa, com discs durs, cintes, etc. Tot i que alguns sistemes concrets poden incorporar-los .

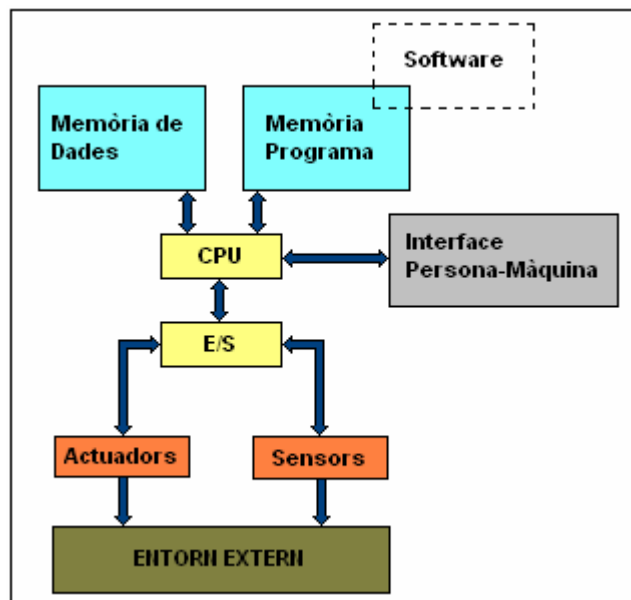


Figura 3.1. Diagrama funcional sistema embedded. Referència: article "PC embedded" del nº 316 de la revista "Automática e Instrumentación" (Març 2001), Julián Horrillo.

El que casi sempre existeix és algun tipus de funció de comunicació amb l'exterior que permet l'establiment dels paràmetres de funcionament de l'aplicació, així com el registre de successos. Per tant, existeix algun tipus d'interfície home-màquina, tot i que la interactivitat acostumi a ser molt reduïda.

El hardware dels sistemes *embedded*, acostuma a estar muntat a plaques de circuit imprès (PCB); on es connecten els diferents components electrònics que formen el sistema. Tot el hardware del sistema *embedded* està localitzat a la capa hardware del model de sistemes *embedded*.



Figura 3.2. Model de Sistema Embedded. Referència: "Real-Time Concepts for Embedded Systems", Qing Li, ed. CMP Books, (2006).

Els components hardware d'un sistema *embedded* es poden classificar en cinc principals grups:

- *Central Processing Unit* (CPU). El processador principal.
- Memòria. On s'emmagatzema el software de sistema.
- Dispositius d'entrada.
- Dispositius de sortida.
- Components del bus.

Les cinc categories estan basades en el model de *Newmann*, mundialment utilitzat per a l'estudi de les arquitectures hardware de dispositius electrònics.

## 3.2 – PLATAFORMES HARDWARE

### 3.2.1 - Plataformes PC Embedded

Existeixen diferents formats d'implementació de la plataforma PC, des d'una placa base activa que incorpora la CPU, els recursos de memòria activa i passiva i els circuits auxiliars del sistema (format utilitzat pels PC de sobretaula), fins a la utilització d'un *backplane* passiu que simplement conté els connectors per a la inserció de les diferents targetes que configuren el sistema.

A la figura 3.3 podem observar un exemple de *backplane* de iEi Technology Corporation que incorpora 1 slot de PCIe i 16 slots PCI, per a la connexió de tot tipus de dispositius. Aquest tipus de *backplane* va fixat en un rack industrial RACK-3200G-R20.

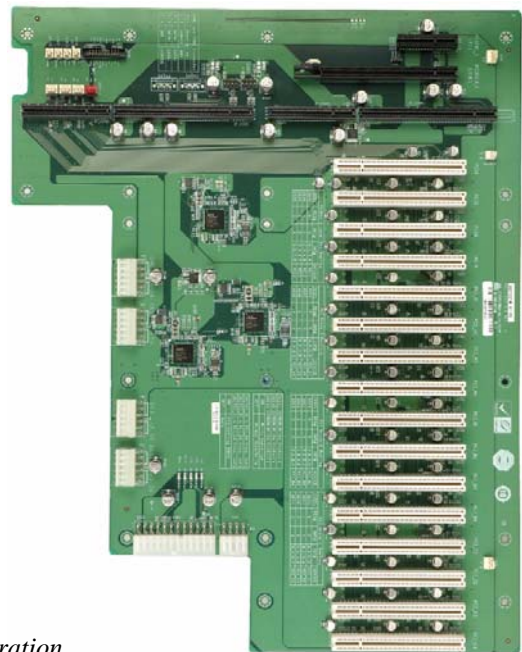


Figura 3.3. Backplane PXE-19S2 de iEi Technology Corporation.

Referència: <http://www.ieiworld.com/>.

Per a aplicacions industrials, no obstant, és molt habitual que bona part dels recursos hardware i software que incorpora un PC de propòsit general no siguin necessaris per a l'aplicació concreta que es desenvolupa. És aquí on apareix en concepte Embedded, com a plataforma que incorpora els recursos estrictament necessaris i que s'integra en un sistema major realitzant unes funcions específiques. A partir d'aquest moment, comença a guanyar força l'ús d'ordinadors en una sola targeta de circuit imprès (SBC: *Single Board Computer*). Els SBC combinen una CPU amb alguns perifèrics, permetent configurar solucions amb un elevat grau d'integració. En la pràctica totalitat dels casos, aquesta targeta incorpora vies d'expansió a través d'algun bus extern que permet acomodar mòduls de disseny propi o comercials.

Els SBC no necessàriament incorporen un processador Pentium IV. En moltes aplicacions serà suficient l'ús d'un 386 o un 486. De fet, les principals empreses de microprocessadors destinen processadors ja obsolets per aplicacions ofimàtiques a les línies de productes d'aplicació industrial, on sí reuneixen les característiques adequades per al seu ús.

Els requisits de l'aplicació determinaran la potència del processador i quins perifèrics ha d'incorporar el SBC; intentant minimitzar d'altra banda, necessitats d'expansió del sistema en futures modificacions.

Una de les arquitectures més adoptades pels fabricants de SBC és l'estàndard PC/104. L'arquitectura PC/104 pot integrar, en una targeta de 96mm x 90 mm, el processador, la memòria, la controladora de vídeo, controladora Ethernet, el bus d'expansió i les E/S, entre d'altres coses.

Per exemple, sota l'estàndard EBX (que forma part també del consorci PC/104) podem trobar SBC que incorporen processadors Intel Core Duo a 667 MHz, *slots* de memòria 2xDDR fins a 3 GB, controladora de vídeo i àudio, ports sèrie i paral·lel, USB 2.0, interfície d'E/S de 8 bits i tots els requeriments necessaris en una *board* de 146.05mm x 203.20mm.

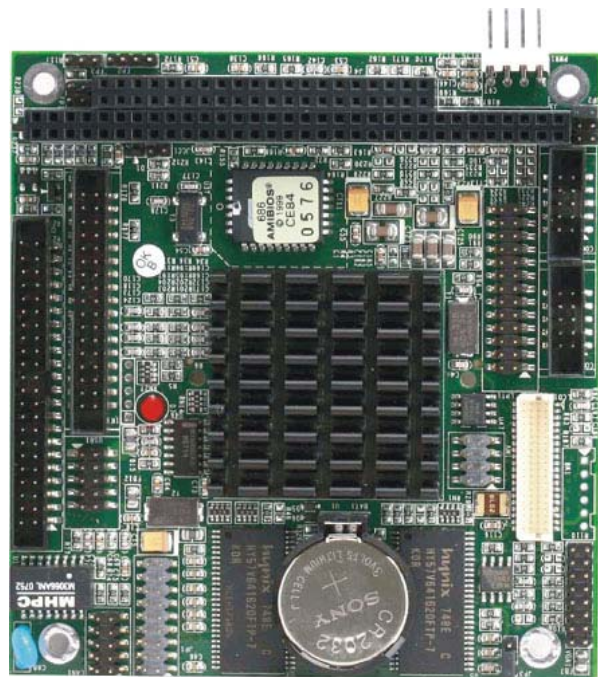


Figura 3.4. Targeta 104-1462CLDN SBC PC/104 486 SBC amb CPU (133MHz), memòria i LCD, VGA, SSD, i LAN. Referència: [www.evoc.com](http://www.evoc.com).

En el següent subapartat s'explica de forma detallada l'estàndard PC/104, desde el seu origen, les diferents configuracions actuals i els principals avantatges i inconvenients d'aquesta tecnologia.

### 3.2.2 - Els estàndards embedded PC/104

A finals dels vuitanta, el grup Ampro Computers ([www.ampro.com](http://www.ampro.com)), va crear el consorci PC/104 Embedded; per a la creació d'especificacions d'arquitectures *desktop* a sistemes embedded. Així es va crear l'arquitectura PC/104, un estàndard que defineix el format de la placa base (*form factor*) i el bus del sistema per aplicacions PC embedded. L'especificació formal del format PC/104 (IEEE P996.1), no obstant, no es publica fins l'any 1992. El nom de l'arquitectura es deu a la configuració d'una plataforma PC al voltant d'un bus ISA de 16 bits (mantenint els seus 104 pins i afegint-ne algunes línies de massa per a millorar la integritat dels senyals).

Al 1997 apareix l'estàndard PC/104-Plus (bus de 120 pins), versió compacta que incorpora també el bus PCI. Al 2003 es publica l'estàndard PCI-104; que ve seguit de l'especificació PCI/104 Express que publica el consorci durant l'any 2008. Actualment, el consorci administra els estàndards PC/104, PC/104-Plus, PCI-104, EBX i EPIC, amb una gran presència al mercat; i que poden ser consultats desde la pàgina web del consorci [www.pc104.org](http://www.pc104.org) .

#### PC/104

Un sistema PC/104 compren una targeta amb la CPU i alguns recursos generals, incorporant-se els recursos perifèrics necessaris per a l'aplicació, a targetes addicionals. Aquesta arquitectura no necessita *backplanes* ni *motherboards*; les targetes disposen de connectors que permeten la connexió directa entre elles, utilitzant una estructura de targetes apilades. D'aquesta manera es millora el grau d'integració i es redueix la grandària dels equips informàtics. Els inconvenients d'aquesta arquitectura radiquen en els problemes d'accessibilitat a les targetes d'una columna, especialment si es tracta de la zona intermitja; lo qual dificulta la substitució d'algun mòdul. Entre les seves característiques principals es destaquen:

- **Grandària:** mòduls de 3.55 x 3.775 polzades ( 90.17 x 95,89 mm ), tal com indica la figura 3.4.

- **Arquitectura expansible:** les targetes es poden apilar, amb una separació entre mòduls de 0.6 polzades (tal com indica la figura 3.5), reduint espai i cost.
- **Resistència:** al xoc i les vibracions.
- **Baix consum:** estalvi energètic amb consums de 6mA (1 o 2 watts) per mòdul.
- **Baix cost**

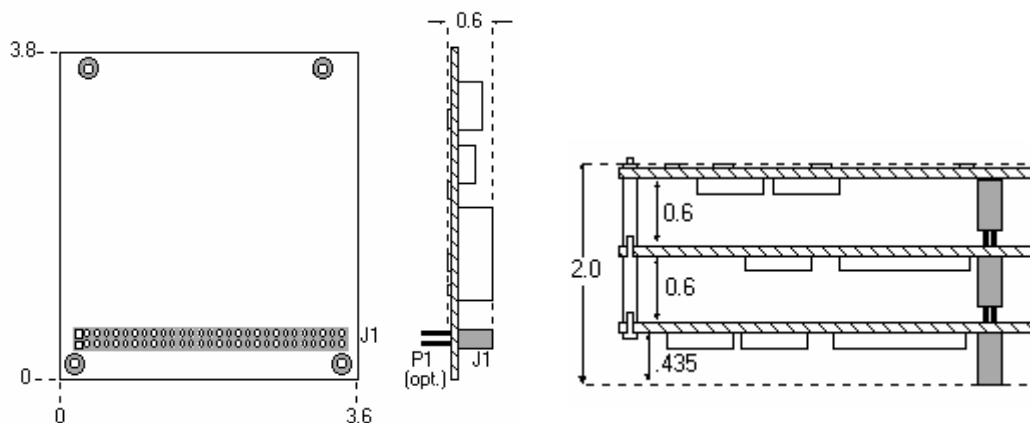


Figura 3.5. Dimensions bàsiques (versió de 8 bits) i apilació de mòduls autònoms. Referència: [http://www.pc104.org/pc104\\_specs.php](http://www.pc104.org/pc104_specs.php).

Una altra possibilitat que ofereix l'arquitectura PC/104 és la d'utilitzar els mòduls com a components altament integrats, assemblats dins de targetes que contenen una lògica i una interfície per a una aplicació específica; tal com indica la figura 3.6. L'arquitectura està preparada per l'addició de múltiples mòduls en una ubicació; facilitant l'actualització i ampliació del hardware durant la depuració o test del programa.

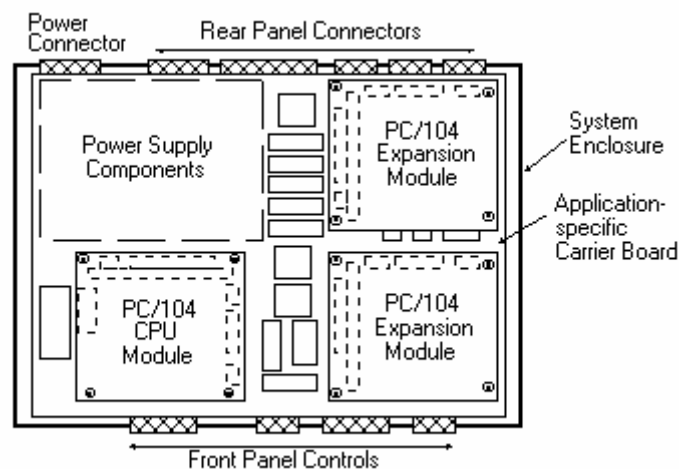


Figura 3.6. Components com a aplicacions. Referència: [http://www.pc104.org/pc104\\_specs.php](http://www.pc104.org/pc104_specs.php).



## PC/104-Plus

L'especificació PC/104-Plus estableix l'estàndard d'utilització del bus d'alta velocitat PCI a aplicacions embedded. Es tracta d'una versió del bus PCI, de 32 bits i 120 pins d'alta densitat (2mm), que possibilita la ràpida transferència de dades (132 Mbytes per segon). Format flexible, fiable i robust que manté les característiques de l'arquitectura PC/104.

Nombrosos fabricants ofereixen productes de desenvolupament amb l'especificació PC/104-Plus com computadores SBC, *high-speed* LAN, interfícies de vídeo, interfícies de comunicacions (100BaseT, USB, IEEE-1394, etc.) i mòduls de control i adquisició de dades. A la figura 3.7 es poden observar les dimensions de les targetes PC/104-Plus.

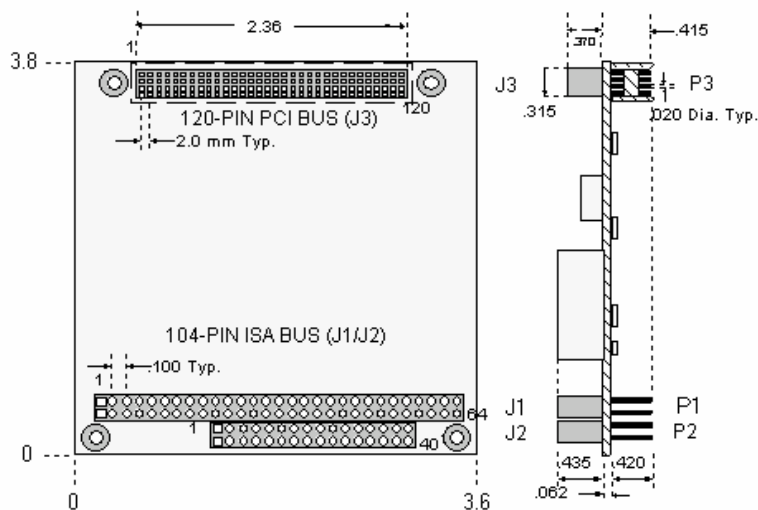


Figura 3.7. Dimensions bàsiques de PC/104-Plus.

Referència: [http://www.pc104.org/pc104\\_plus\\_specs.php](http://www.pc104.org/pc104_plus_specs.php).

## PCI-104

La diferència de l'estàndard PCI-104 respecte el seu antecessor està en l'eliminació dels connectors AT i AX, del bus ISA, encara presents a l'especificació PC/104-Plus. Com a contrapartida a la seva robustesa, el bus PCI-104, de 120 pins, no suporta extensions de 64 bits, ni els senyals JTAG, PRSNT o CLKRUN; a diferència del bus local de 32 bits PCI, amb connector de 124 pins.

## PCI/104-Express

PCI/104-Express estableix un estàndard d'utilització del bus d'alta velocitat PCI Express a aplicacions embedded. S'opta per PCI Express donada la seva omnipresència al mercat PC, la seva escalabilitat i la creixent disponibilitat de la tecnologia del silici a tot el món. Proporciona una nova interfície física de gran rendiment, compatible a nivell de software amb la infraestructura del bus PCI; mantenint alhora les especificacions de PC/104.

## Evolució de l'arquitectura PC/104

L'evolució de l'estàndard PC/104 ha anat acompanyada de la del PC *desktop*, amb l'objectiu de proveir hardware i software totalment compatible amb aquesta plataforma. Els principals formats que ofereix PC/104 a l'actualitat són els següents:

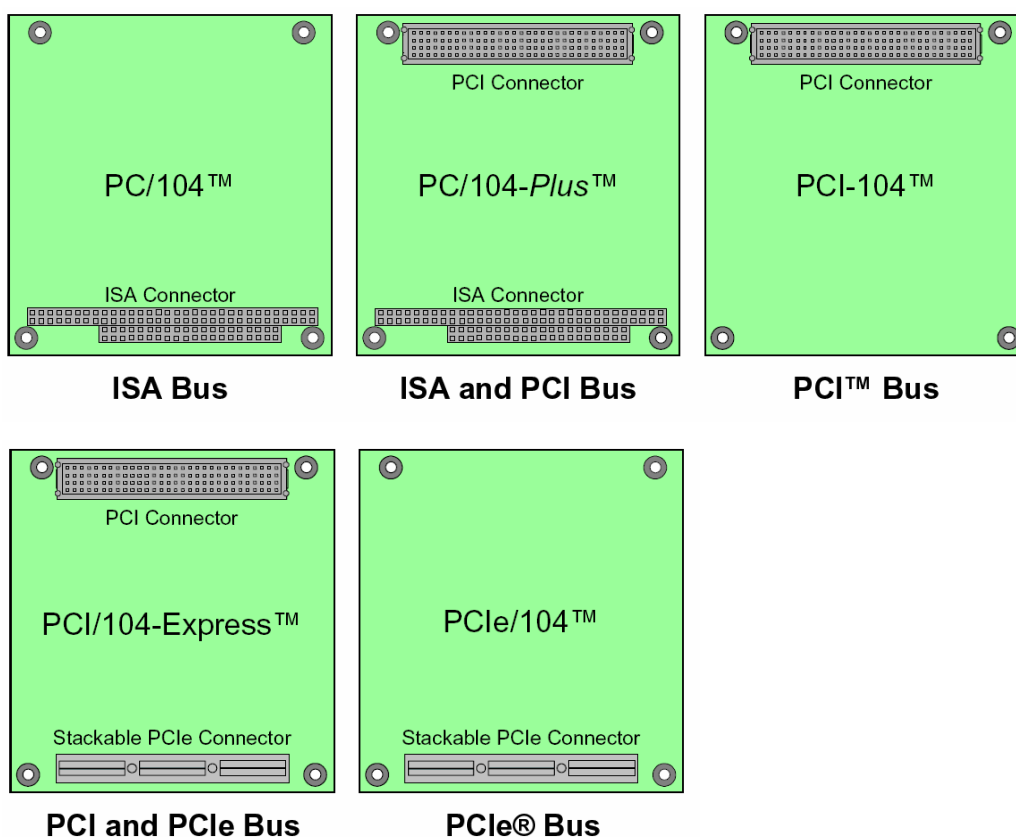


Figura 3.8. Evolució de l'arquitectura PC/104. Referència: [http://www.pc104.org/pc104\\_specs.php](http://www.pc104.org/pc104_specs.php)

## Altres formats PC/104

EPIC i EPIC Express són plataformes embedded, per aplicacions industrials, introduïdes pel consorci PC/104. Amb una grandària un mica superior al format 104 (115 x 165 mm), incorporen l'espai necessari per a la ubicació de ventiladors d'alta refrigeració per a processadors i connectors d' E/S habituals als PC estàndard.



Figura 3.9. Targeta EPIC Nano 7240 de iEi Technology Corp amb processador On board Intel® ULV Celeron® 400MHz processor. Referència: [www.ieiworld.com](http://www.ieiworld.com).

EBX i EBX Express són plaques embedded expandibles amb el format 5 ¼, emprat a molts SBC. En una superfície de 146.05 x 203.20 mm incorporen una computadora completa, amb E/S i memòria estàndard DIMM; mantinguent la flexibilitat i expandibilitat de l'arquitectura PC/104.

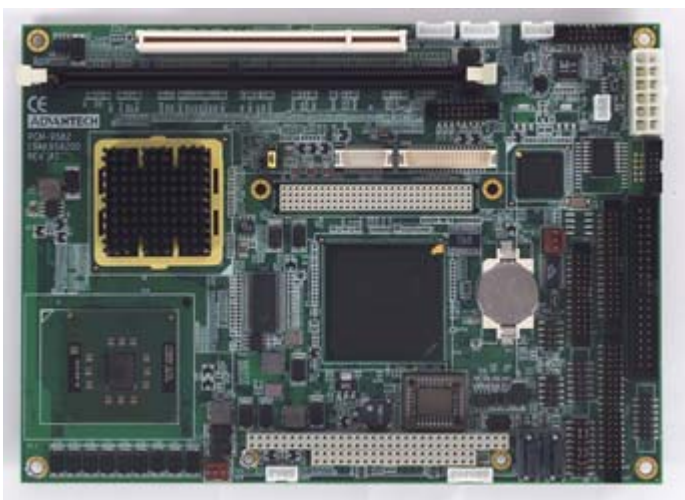


Figura 3.10. Targeta EBX PCM-9587 del grup Advantech amb processador Intel Celeron M i LAN, LVDS, DVI, VGA, i Audio. Referència: [www.advantech.com](http://www.advantech.com).

## Avantatges de PC/104

L'arquitectura PC/104, amb les seves diferents configuracions anteriorment descrites, té un alt grau d'acceptació al mercat actual. Els seus principals avantatges es descriuen a continuació:

- **Mòduls apilables:** El sistema embedded es pot configurar, segons les necessitats específiques de l'aplicació, de forma fàcil, apilant els mòduls en columnes.
- **Interoperabilitat:** Les targetes PC/104 són distribuïdes en múltiples dissenys per desenes de fabricants en tot el món. Estan dissenyades per a funcionar interconnectades i ofereixen la possibilitat de futures ampliacions.
- **Compacte:** La seva reduïda mida (similar a la d'un *diskette* ) permet la seva implementació en instruments portàtils, taulells de comandament de cotxes, vehicles militars, satèl·lits, etc. Per exemple, la targeta *Prometheus PC/104* de Diamond Systems (visitar <http://www.diamondsystems.com>) es va utilitzar en un petit satèl·lit de recerca, que mesurava tan sols 4" x 4" x 12".
- **Tecnologia basada en PC:** PC/104 utilitza els busos ISA i PCI, i processadors com Intel i altres de la família x86. Els dissenyadors de sistemes, que estan altament familiaritzats amb la tecnologia PC i els seus components, podran dissenyar fàcilment amb PC/104, donada la seva similitud. La portabilitat del software permet programar còmodament desde plataformes PC i bolcar, de forma senzilla, els programes al PC/104.
- **Robustesa:** Les targetes PC/104 són de forma inherent robustes, donats els seus connectors de bus de sistema d'alta densitat i els seus forats de fixació a les quatre cantonades. Això fa que siguin vàlides per aplicacions en entorns agressius, com vehicles militars, amb xocs i vibracions, que malmetrien sistemes basats en tecnologia *desktop*.

### 3.2.3 - El processador

El processador és la principal unitat funcional i el principal responsable del processament d'instruccions i dades. Un sistema embedded té, com a mínim, un processador mestre, actuant de controlador principal del sistema, que pot anar acompanyat de processadors esclaus, governats pel processador principal. Els processadors esclaus poden complementar funcions del processador principal o encarregar-se del control dels busos, memòria i/o dispositius d'E/S.

Un exemple d'això es pot observar a la figura 3.11, corresponent al diagrama de blocs de la targeta Encore 400 del grup Ampro, on el processador STPC exerceix com processador mestre i els controladors d'E/S i Ethernet són processadors esclaus.

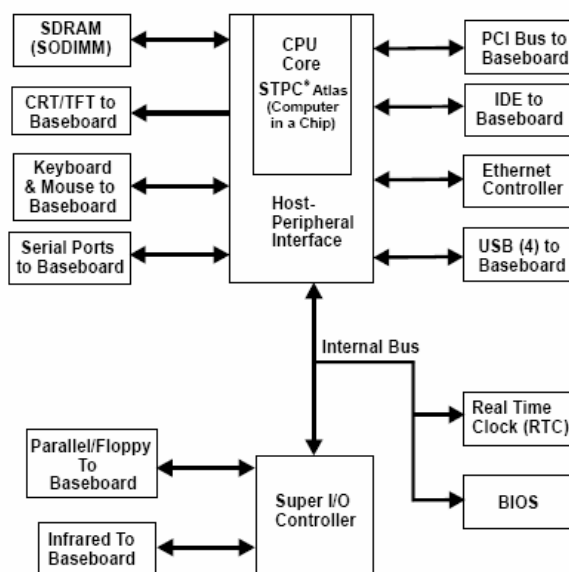


Figura 3.11. Targeta Encore 400 de Ampro. Referència: <http://www.ampro.ru/pdf/EnCore-400-EN.pdf>.

### Elecció del microprocessador

A l'hora d'escollir el microprocessador del nostre sistema *embedded* és recomanable escollir una arquitectura altament estesa al mercat; amb gran disponibilitat de hardware compatible i solucions software. La disponibilitat de productes compatibles servirà per a minimitzar la càrrega de treball del traspàs a un nou processador quan les circumstàncies ho demandin.

Dins el mercat embedded hi ha centenars de processadors disponibles. Aquests poden ser classificats en diferents grups, anomenats architectures. Les diferents architectures varien en funció del conjunt d'instruccions de codi màquina que els processadors poden

executar. De fet, els processadors es consideren de la mateixa arquitectura si poden executar el mateix *set* d'instruccions de codi màquina. A la taula 3.1 es mostren les principals arquitectures, models i fabricants de processadors.

Arquitectura	Processador	Fabricant
AMD	Au xxx	Advanced Micro Devices, ...
ARM	ARM7, ARM9, ...	ARM, ...
C16X	C167CS, C165H, C164CI, ...	Infineon, ...
ColdFire	5258, 5272, 5307, 5407, ...	Motorola / Freescale, ...
1960	1960, Vmetro, ....	
M32 / R	32170, 32180, 32182, 32192, ...	Renesas / Mitsubishi, ...
M Core	MMC2113, MMC2114, ...	Motorola / Freescale, ...
MIPS32	R3K, R4K, 5K, 16, ...	MTI4kx, IDT, MIPS Technologies, ...
NEC	Vr55xx, Vr54xx, Vr41xx	NEC Corporation, ...
PowerPc	82xx, 74xx, 8xx, 7xx, 6xx, 5xx, 4xx	IBM, Motorola / Freescale, ...
68k	680x0 ( 68K, 68030, 68040, 68060, ... ), 683xx	Motorola / Freescale, ...
SuperH ( SH )	SH3 ( 7702, 2207, 7708, 7709 ), SH4 ( 7750 )	Hitachi, ...
SHARC	SHARC	Analog Devices, Transtech DSP, Radstone, ...
strongARM	strongARM	Intel, ...
SPARC	UltraSPARC II	Sun Microsystems, ...
TMS320C6xxx	TMS320C6xxx	Texas Instruments, ...
x86	X86 [ 386, 486, Pentium (II, III, IV) ... ]	Intel, Transmeta, National Semiconductor, Atlas, ...
TriCore	TriCore 1, TriCore 2, ...	Infineon, ...

Taula 3.1. Principals arquitectures, models i fabricants de processadors. Referència: “Embedded hardware, know it all”, Jack Ganssle, ed. Newnes, 2008.

En general, actualment de la llista anterior destaquen sis processadors de 32 bits molt utilitzats al mercat: Motorola 680x0, Intel x86, PowerPC, MIPS, SuperH, i ARM. Existeixen també altres arquitectures menys populars o propietàries; moltes d'aquestes associades a aplicacions específiques com impressores làser o reproductors de DVD.

La utilització d'arquitectures compatibles amb la família x86 (i compatibles en codi d'AMD, National Semiconductor, Via Technologies, etc.) té una sèrie d'interessants avantatges:

- Es pot utilitzar qualsevol sistema operatiu compatible amb PC i eines de desenvolupament de software gratuïtes.

- La instal·lació de SO és senzilla; en molts casos amb l'ajuda instal·ladors automàtics, que instal·laran el *kernel* apropiat i els *drivers* necessaris.
- Igual de senzilla és la comunicació amb una gran quantitat de perifèrics. Es tracta d'unes arquitectures de processador presents en el mercat de consum, amb una gran disponibilitat de productes a preus molt competitius.
- Existeixen *drivers* de suport (inclosos en els sistemes operatius) per a cada component hardware que es vulgui incloure al sistema embedded.
- Disponibilitat d'una gran varietat de *mainboards*, amb diferents formats i combinacions de perifèrics.
- La migració a una plataforma lleugerament diferent, degut a la escassetat de suport dels proveïdors o les necessitats evolutives dels clients, es pot fer de forma relativament senzilla. En molts casos, amb recompilar i reinstal·lar el sistema operatiu i preparar la nova imatge del disc mestre n'hi ha prou.

Com a contrapartida als avantatges anteriorment indicats també existeixen una sèrie d'inconvenients, que es descriuen a continuació:

- Els components x86 són bastant més cars, en volums importants de producció, que les alternatives RISC de rendiment comparable. Això afecta a la competitivitat de preus del producte que es pot oferir al mercat.
- Hi ha relativament poques alternatives x86 que siguin veritablement dispositius *system-on-chip*; de manera que és probable que es tingui que afegir hardware extern per a formar el sistema. Habitualment, per a obtenir una petita i específica funció, serà necessari utilitzar un mòdul multifuncional complex; davant la no disponibilitat d'aquesta funció concreta en un dispositiu senzill. Això pot fer més complex i car el sistema.

- Els x86 tenen inconvenients associats al consum de potència, la dissipació de temperatura (amb la necessitat d'incorporar un ventilador al xip) i grandària.

És molt recomanable la x86 com la plataforma d'elecció si és vol desenvolupar un sistema en reduïdes unitats o un prototip que contingui moltes característiques variades de hardware sense una gran despesa en temps de depuració del programa.. És també una bona elecció per a una sèrie de producció inicial, per a ser testejada, de cara a una segona etapa de disseny amb hardware més barat, personalitzant més el disseny.

Si parlem de sistemes *embedded* integrats en aplicacions de producció industrial, l'arquitectura x86, fonamentalment dissenyada per a plataformes PC, pot ser una bona elecció. Com ja s'ha explicat en seccions anteriors, la omnipresència de productes compatibles amb PC al mercat ofereix grans facilitats d'actualització i modificació dels equips d'aquesta arquitectura. Això, juntament amb la gran familiaritat per part dels dissenyadors en la programació i configuració del software en l'arquitectura PC, són els factors claus per a l'elecció dels sistemes PC *embedded*.

El fet d'utilitzar un processador x86 no significa que hagi d'anar implementat en una *motherboard* típica d'un PC *desktop*, de grandària considerable. Els formats SBC, amb el suport d'estàndards com PC/104 (del que s'ha parlat àmpliament al tema 3), ofereixen solucions amb una alt poder computacional en reduïdes mides.

Una altra família de microprocessadors de 32 bits a sistemes *embedded* digna de ser comentada és l'arquitectura ARM. És un dels processadors més utilitzats en el món *embedded*, de la mateixa manera que els x86 ho són en el món PC *desktop*. És una arquitectura consolidada, amb molts productes compatibles al mercat, que garanteixen el subministrament i suport de la plataforma en diferents mides, funcionalitats i dissenys. A més, són processadors petits i amb un baix consum en relació amb el seu rendiment. Amb tot això es pot dir que és una alternativa interessant per als sistemes *embedded*.



## 3.3 – COMUNICACIONS

### 3.3.1 – Aspectes generals

Un camp de vital importància dintre de la informàtica industrial és el de les comunicacions embedded. Sovint, els sensors, actuadors, monitors i d'altres elements que componen un sistema embedded estan ubicats a diferents llocs. Això fa necessària l'existència d'una xarxa de comunicacions. A més, el sistema embedded també ha de comunicar-se amb el sistema superior al que està subordinat, o amb altres sistemes embedded.

Aquesta funció d'interconnexió la pot fer una targeta d'E/S ( *I/O board* ) o una targeta CPU embedded que tingui les E/S integrades a la placa. Les interfícies d'E/S poden estar compostes per algun o varis de les següents elements:

- **Medi de transmissió**, cablejat o sense fils, connecta el dispositiu d'E/S amb la targeta embedded per la comunicació i l'intercanvi de dades.
- **Port de comunicació**, a través del qual el medi de transmissió connecta amb la targeta d'E/S. En el cas de comunicacions *wireless*, serà el receptor del senyal *wireless*.
- **Interfície de comunicació**, que gestiona la comunicació de dades entre la *master CPU* i el dispositiu o controlador d'E/S i és el responsable de codificar i descodificar les dades.
- **Controlador d'E/S**, un processador *slave* que gestiona el dispositiu d'E/S.
- **Busos d'E/S**, la connexió entre la targeta d'E/S i el processador *master*.
- **Processador mestre**, amb E/S integrades.

Quan parlem de xarxes de comunicació, hem de parlar del model OSI. El model de referència d'Interconnexió de Sistemes Oberts OSI (*Open System Interconnection*) publicat al 1984, fou un el model de xarxa descriptiu creat por ISO (*International Organization for Standardization*); esdevenint un marc de referència per a la definició d'arquitectures d'interconnexió de sistemes de comunicacions. El model OSI està format per diferents nivells o capes. Les capes de nivell més baix proporcionen serveis a les capes immediatament superiors. No obstant, totes les capes tenen una responsabilitat durant la transmissió de dades. Per als sistemes de comunicació embedded, els nivells OSI més importants són els nivells 1 i 2, corresponents al nivell físic i d'enllaç de dades respectivament (observar figura 3.12).

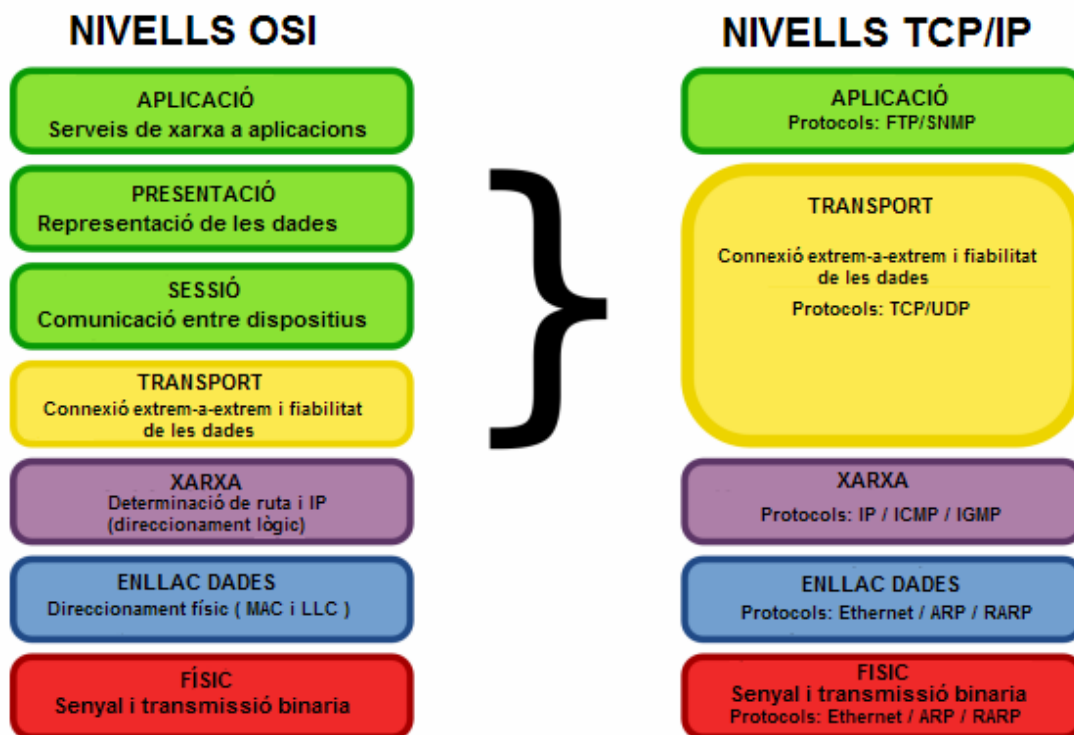


Figura 3.12. Capes o nivells OSI i TCP/IP d'un sistema de xarxa. Com es pot observar, Ethernet correspon als nivells físic i d'enllaç de dades. Referència: [http://es.wikipedia.org/wiki/Modelo\\_OSI](http://es.wikipedia.org/wiki/Modelo_OSI).

La figura 3.12 ens mostra una comparativa entre els models OSI i TCP/IP. Tal com es pot observar, Ethernet pertany als nivells físic i d'enllaç de dades; que són els més importants per als sistemes embedded, tal com ja hem indicat.

TCP/IP (*Transmission Control Protocol / Internet Protocol*) és el nom que s'empra per a referir-se a un important número d'especificacions de protocols (TCP/IP/ARP/RARP/ICMP/IGMP/UDP/BOOTP/D HCP/FTP/HTTP), i que es basa, d'igual manera que el model OSI, en el concepte de capa o nivell, on cada nivell realitza un treball específic i delimitat.

Ethernet té diverses propietats que el fan ideal per la majoria d'aplicacions de sistemes embedded distribuïts. Primer, Ethernet és un protocol completament descentralitzat. No hi ha una estació centralitzada de *polling* que pugui fallar. Segon, Ethernet en la majoria de les seves versions bàsiques corre en 10 Mbit/s, una magnitud molt major que la de la majoria dels protocols embedded. I tercer, és un protocol de xarxa conegut i àmpliament estès a plataformes desktop, lo qual redueix el temps de configuració i assegura la connectivitat. Seguirem parlant d'Ethernet amb més profunditat més endavant.

Protocols com Fieldbus, ControlNet, Interbus, i CAN han estat desenvolupats específicament per a la connexió de sistemes embedded de temps real. Aquests protocols acostumen a ser lents (al voltant de 1 Mbit/s), però ofereixen un major determinisme que múltiples mètodes d'accés utilitzats per xarxes *desktop*. Per a aplicacions de temps real dur (del que en parlarem més endavant), el determinisme és clau, atès que la pèrdua de missatges pot tenir resultats catastròfics. Ethernet no es determinista. No obstant això, per a sistemes de temps real suau, que no fallen perillosament si un missatge és perdut ocasionalment, és possible que el protocol d'una xarxa local d'alta velocitat com LAN (Local Area Network) pugui proporcionar un funcionament comparable a un protocol embedded específic.

A la taula 3.2 es mostra una comparativa de diferents interfícies de comunicació. Podem observar com els protocols més actuals, com firewire o USB, ofereixen una velocitat molt superior a protocols tradicionals com el RS-232, d'un ús més industrial.

Interface	Format	Número màxim de dispositius	Longitud màxima (m)	Velocitat màxima (bits/seg.)
RS-232 (TIA/EIA-232)	asíncron	2	15 - 30	20K (115K amb alguns drivers)
	sèrie			
RS-485(TIA/EIA-232)	asíncron	32 unitats en càrrega	1200	10M
	sèrie			
IrDA	asíncron	2	1.8	115K
	infraroig sèrie			
Microwire	síncron	8	3	2M
	sèrie			
SPI	síncron	8	3	2.1M
	sèrie			
I <sup>2</sup> C	síncron	40	5.5	400K
	sèrie			
USB	asíncron	127	5	12M
	sèrie			
Firewire	sèrie	64	4.5	400M
IEEE-488 (GPIB)	paral·lel	15	18	1M
Ethernet	sèrie / paral·lel	1024	487	10M
MIDI	bucle de corrent sèrie	2	4.5	31.5K
Paralell Printer Port	paral·lel	2 o 8 (amb suport)	03-Sep	1M

Taula 3.2. Comparativa d'interfícies de comunicació populars. Referència: "Serial port complete: programming and circuits for RS-232 and RS-485 links and networks", Jan Axelson, ed. Ilustrated.

### 3.3.2 - Comunicacions sèrie

A les comunicacions sèrie les dades són emmagatzemades, transferides i rebudes una a una (un bit per instant de temps). Les interfícies sèrie gestionen la transmissió i recepció de dades entre la *CPU master* i el controlador o dispositiu d'E/S. Aquestes dades poden ser transmeses de tres maneres diferents:

- **Simplex:** Un node exerceix de transmissor i l'altre de receptor, i no poden intercanviar els papers. Observar figura 3.13.
- **Half-Duplex:** Ambdós nodes poden transmetre i rebre dades, però no simultàniament. Quan un node està enviant dades l'altre exerceix de receptor i a la inversa. Observar figura 3.14.
- **Full-Duplex:** Ambdós nodes poden transmetre i rebre dades simultàniament.

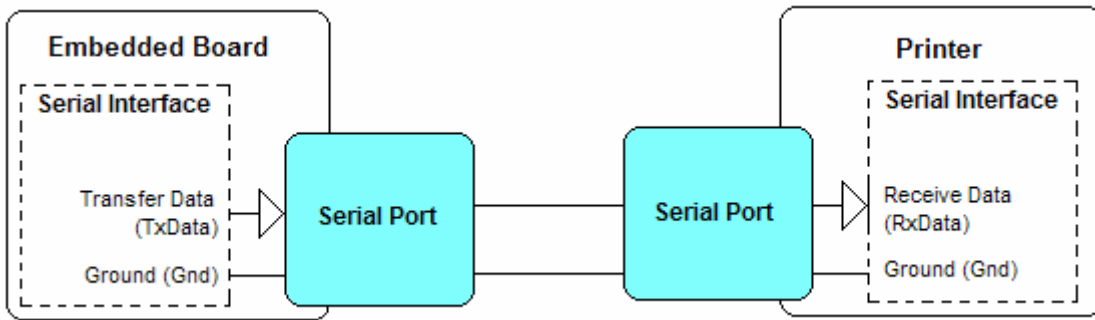


Figura 3.13. Esquema exemple d'una transmissió Simplex.

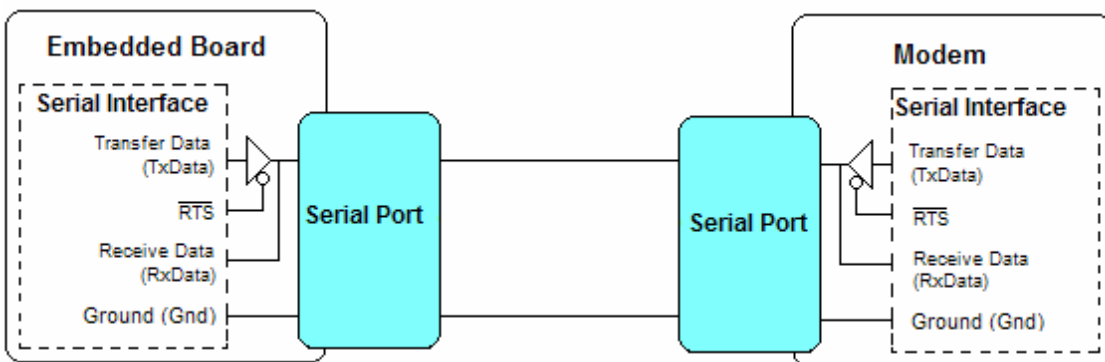


Figura 3.14. Esquema exemple d'una transmissió Half-duplex.

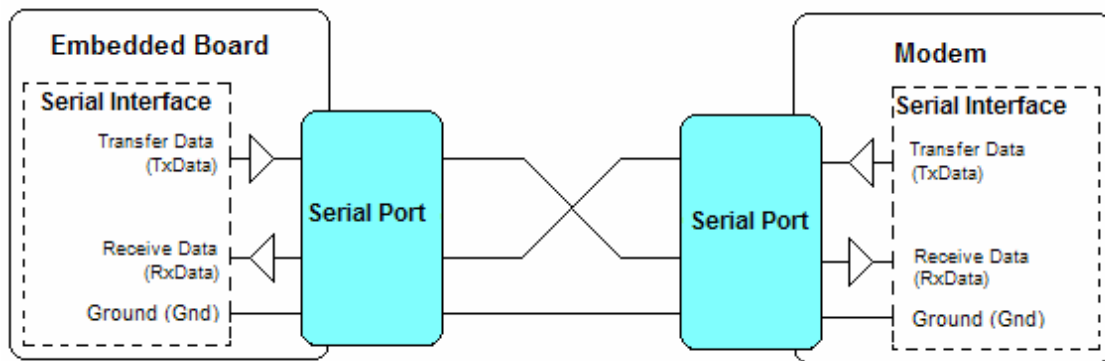


Figura 3.15. Esquema exemple d'una transmissió Full-duplex. Referència: "Embedded hardware, know it all", Jack Ganssle, ed. Newnes, 2008.

Aquestes transmissions poden fer-se de forma síncrona, per intervals regulars de temps (el *clock* de la CPU determina les transferències) o asíncrona, amb intervals de temps irregulars.

El gran avantatge de les comunicacions sèrie és la suficiència d'un o dos pins d'E/S per a realitzar la comunicació, en comparació amb els vuit o més pins necessaris a les

comunicacions paral·leles. Molts perifèrics de sistemes embedded comuns, com convertidors ADC i DAC, LCD, i sensors de temperatura, suporten interfícies sèrie.

Els busos sèrie poden també realitzar comunicacions entre processadors. Això permet realitzar tasques, amb grans volums de processament de dades, que requeririen un processador més potent, realitzar-les amb petits processadors, generalment més econòmics.

Els estàndards sèrie més populars i àmpliament utilitzats són els pertanyents a la família EIA-RS-232/422/485. Ideals per aplicacions de baix cost que requereixen interconnexió sèrie. A la taula 3.3 es mostra una comparativa dels tres estàndards, amb les seves característiques principals.

Paràmetre	RS-232	RS-422	RS-485
Mode d'operació	Simple	Diferencial	Diferencial
Número de dispositius	1 emissor - 1 receptor	1 emissor - 10 receptors	32 emissors - 32 receptors
Màx. longitud cable	15 m	1200 m	1200 m
Màx. Velocitat	20 Kbps	10 Mbps	10 Mbps
Càrrega <i>driver</i>	De 3k a 7k	100 mínim	60 mínim
Entrada <i>receiver</i>	De 3k a 7k	4k	16k
Tensió en mode comú	$\pm 25V$	$\pm 7V$	De -7V a 12V

Taula 3.3. Anàlisi comparatiu entre estàndards de comunicació sèrie. Referència: "Instrumentación virtual: Adquisición, procesado y análisis de señales", Antoni Mànuel, Edicions UPC1901.

## RS-232

L'estàndard RS-232 o EIA-232 (*Electronic Industries Association-232*) és un dels protocols sèrie més àmpliament implementats a transmissions síncrones i asíncrones. Va ser utilitzat originàriament per a connectar teletips i computadors a mòdems. Molts dispositius, ja obsolets, com impressores i plotters van utilitzar l'estàndard RS-232C. Amb la necessitat de transferir grans quantitats de dades de forma ràpida, el RS-232 va ser suplantat com a connexió estàndard per a protocols més ràpids, com Ethernet. No obstant, segueix sent una interfície molt útil per a la connexió de dispositius a sistemes embedded.

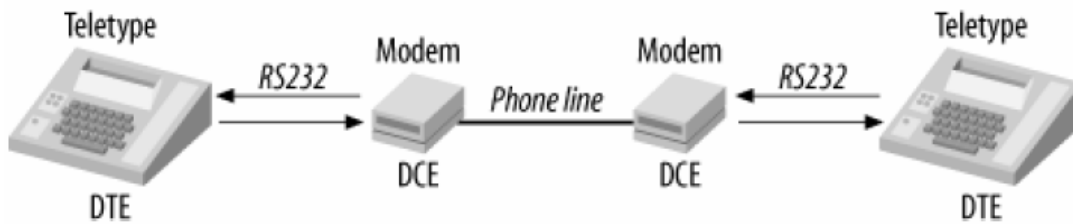


Figura 3.16. Aplicació original del RS-232: connectar teletips a mòdems. Referència: “*Designing Embedded Hardware*”, John Catsoulis, ed. O’Reilly.

Es tracta d’un interfície punt a punt, de velocitats de comunicació baixes i distàncies curtes (20Kbps i 18 metres respectivament). Tot i no estar reflexat a l’especificació, l’interfície pot treballar a majors velocitats i longituds amb resultats acceptables. Connexions de fins a 115 Kbps, amb connexions de curta distància, i fins a 60 metres, emprant cable de baix capacitança, han estat implementades amb èxit.

R-232 és un bus no equilibrat capaç d’establir comunicacions *full-duplex* entre dos parells receptor/transmissor, anomenats *data terminal equipment* (DTE) i *data communication equipment* (DCE). Cadascun té un senyal de transmissió connectat al pin del senyal de recepció de l’altre. Normalment, el PC o el sistema embedded és un DTE i el perifèric connectat és un DCE. Cada transmissor envia les dades variant el voltatge a la línia. Un voltatge més alt de 3V és un “0” binari, mentre que un voltatge de menys que -3V és un “1” binari. Entre aquests voltatges, el valor és indefinit. Per a convertir aquests nivells en nivells de lògica (0 i 5V), pot ser utilitzat un CI de conversió a R-232, com els 1488, 1489, o MAX232. Una comunicació típica RS-232 consisteix en un bit de *start*, bits de dades, bits de paritat (si n’hi ha) i un bit de *stop* (8N1).

L’estàndard RS-232 defineix un total de 25 senyals, junt amb el connector és el DB25. No obstant, EIA RS-232 Standard EIA547 defineix només vuit senyals, compatibles amb el connector DB9 i l’estàndard EIA561 defineix vuit senyals, compatibles amb el connector RJ45.

DB25 Pin	Name	Signal	Description	Voltage	DTE	DCE
1		FG	Frame Ground/Shield		Out	In
2	BA	TxD	Transmit Data	-12	In	Out
3	BB	RxD	Receive Data	-12	Out	In
4	CA	RTS	Request To Send	+12	In	Out
5	CB	CTS	Clear To Send	+12	In	Out
6	CC	DSR	Data Set Ready	+12		
7	AB	SG	Signal Ground			
8	CF	DCD	Data Carrier Detect	+12	In	Out
9			Positive Test Voltage			
10			Negative Test Voltage			
11			Not Assigned			
12		sDCD	Secondary DCD	+12	In	Out
13		sCTS	Secondary CTS	+12	In	Out
14		sTxD	Secondary TxD	-12	Out	In
15	DB	TxC	DCE Transmit Clock		In	Out
16		sRxD	Secondary RxD	-12	In	Out
17	DD	RxC	Receive Clock		In	Out
18	LL		Local Loopback			
19		sRTS	Secondary RTS	+12	Out	In
20	CD	DTR	Data Terminal Ready	+12	Out	In
21	RL	SQ	Signal Quality	+12	In	Out
22	CE	RI	Ring Indicator	+12	In	Out
23		SEL	Speed Selector DTE		In	Out
24	DA	TCK	Speed Selector DCE		Out	In
25	TM	TM	Test Mode	+12	In	Out

Taula 3.4. Descripció dels senyals i pins de RS-232 a connectors DB25. Referència: "Embedded hardware, know it all", Jack Ganssle, ed. Newnes, 2008.

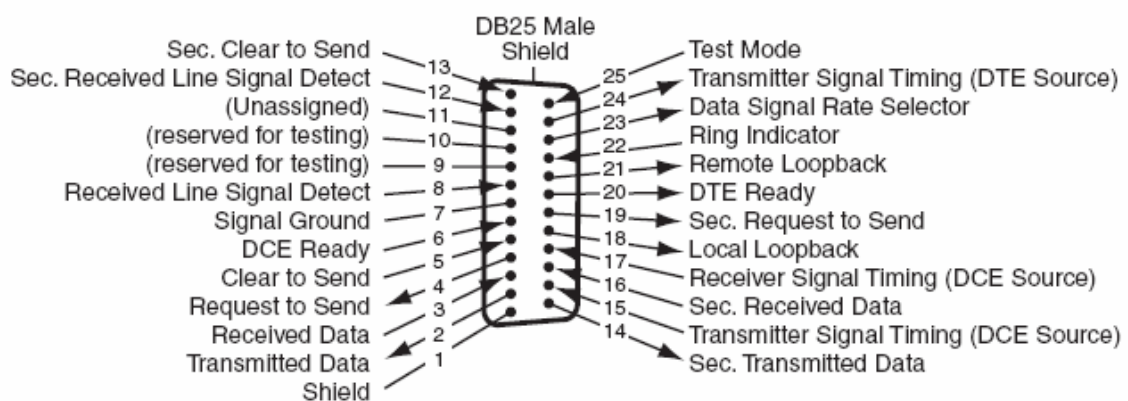


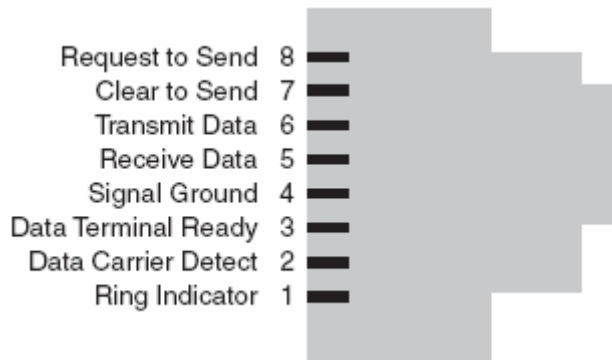
Figura 3.17. Pinatge del connector mascle DB25, RS-232. Referència: "Embedded hardware, know it all", Jack Ganssle, ed. Newnes, 2008.



DB9 Pin	Name	Signal	Description	Voltage	DTE	DCE
1	109	DCD	Data Carrier Detect	+12	In	Out
2	104	RxD	Receive Data	-12	Out	In
3	103	TxD	Transmit Data	-12	In	Out
4	108	DTR	Data Terminal Ready	+12	Out	In
5	102	SG	Signal Ground			
6	107	DSR	Data Set Ready	+12		
7	105/133	RTS	Request To Send	+12	In	Out
8	106	CTS	Clear To Send	+12	In	Out
9	125	RI	Ring Indicator	+12	In	Out

Taula 3.5. Descripció dels senyals RS-232 al connector DB9. Referència: “Embedded hardware, know it all”, Jack Ganssle, ed. Newnes, 2008.

DB9 Pin	Name	Signal	Description	Voltage	DTE	DCE
1	125	RI	Ring Indicator	+12	In	Out
2	109	DCD	Data Carrier Detect	+12	In	Out
3	108	DTR	Data Terminal Ready	+12	Out	In
4	102	SG	Signal Ground			
5	104	RxD	Receive Data	-12	Out	In
6	103	TxD	Transmit Data	-12	In	Out
7	106	CTS	Clear To Send	+12	In	Out
8	105/133	RTS	Request To Send	+12	In	Out



Taula 3.6 i Figura 3.18. Descripció dels senyals RS-232 al connector RJ45. Referència: “Embedded hardware, know it all”, Jack Ganssle, ed. Newnes, 2008.

Dos dispositius DTE poden comunicar-se entre si, utilitzant una variació interna del cables sèrie anomenada *null modem*, que canvia la naturalesa dels DTE i permet comunicacions coordinades.

Molts sistemes embedded utilitzen el bus RS-232 per a la comunicació amb PC o amb perifèrics de PC com mòdems. Normalment, els fabricants de microcontroladors ofereixen productes que inclouen suport de hardware per a RS-232. Per a comunicacions síncrones els anomenats SPI (*Serial Peripheral Interface*). Per a

comunicacions asíncrones, activades normalment per interrupció, els UART (*Universal Asynchronous Receiver / Transmitter*).

## Principals inconvenients de l'estàndard RS-232

La tendència dels dissenyadors de sistemes va ser la de buscar alternatives al RS-232 (com els estàndards sèrie RS-422 i RS-485) degut a les seves següents limitacions:

- Limitació de distàncies: 15 metres. Redueix molt les possibilitats en la connexió de dispositius.
- Limitació de velocitats: 20 kbps. Escassa velocitat per a molts dels equips actuals.
- Voltatges de -3V a -25V i +3V a +25V no són compatibles amb la majoria de fonts d'alimentació actuals, sobretot en el cas de les FA dels computadors, de +5V i +12V.
- Molt susceptible al soroll

## RS-422

L'estàndard RS-422, introduït a principis dels anys 70, defineix un interfície de comunicació sèrie equilibrat; utilitzant 2 fils separats per a cada senyal. El fet de tenir un voltatge diferencial entre els dos fils, fa que sigui molt menys susceptible al soroll que el seu predecessor RS-232. El soroll elèctric afecta als dos fils, però la diferència de potencial entre els dos no es veu massa afectada. Es capaç d'establir comunicacions de longituds de fins a 1200 metres a velocitats de 10 Mbps.

Els voltatges de línia són de -2V a -6V per a un "1" lògic i de +2V a +6V per a un "0" lògic.

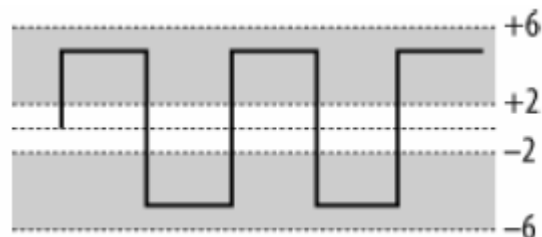


Figura 3.19. Nivells de voltatge d'una comunicació Rs-422.

## RS-485

L'estàndard RS-485 és el més versàtil dels estàndards EIA. Es tracta d'una expansió de l'estàndard RS-422, que permet la mateixa distància i velocitat de comunicació que aquest, però incrementa el número de nodes que poden ser connectats al mateix. Millora àmpliament les prestacions de l'estàndard RS-232 i està molt estès per a l'ús industrial.

La seva senzillesa i la possibilitat de connectar múltiples dispositius (nodes) mitjançant un simple cable de parell trenat el fa molt atractiu. Està basat en una arquitectura *master-slave*, on totes les transaccions estan dirigides pel transmissor mestre, i l'esclau només pot transmetre quan és autoritzat a fer-ho.

Les principals característiques de l'estàndard RS-485 són les següents:

- Distància màxima superior a 1200m.
- Velocitats de transferència superiors a 10Mbps.
- Fins a 32 *drivers* permesos per línia.
- Fins a 32 *receivers* permesos per línia.

El voltatge estipulat per a un "1" lògic està dins el rang de -1.5V i -6V i per al "0" lògic està comprès el rang de +1.5V a +6V. El voltatge de línia és de 5V diferencial entre els 2 fils. Per a sistemes *full-duplex* es necessiten 5 fils conductors. Per a sistemes *half-duplex*, n'hi ha prou amb 3 fils.

El principal avantatge del RS-485 és el funcionament en tres possibles estats de la línia, *tristate*. Els tres estats són "0" lògic, "1" lògic i "alta impedància", que deshabilita el dispositiu de la línia. D'aquesta manera es poden connectar 32 dispositius a la mateixa línia, però que en cada instant només un d'ells estigui actiu. L'estat inactiu o d'alta impedància es pot activar amb un senyal a través del pin de control; de manera que un computador, exercint de *master*, pot gestionar quin transmissor/receptor està actiu a cada moment. Cada dispositiu o terminal té assignada una adreça diferent, per a evitar conflictes de xarxa.

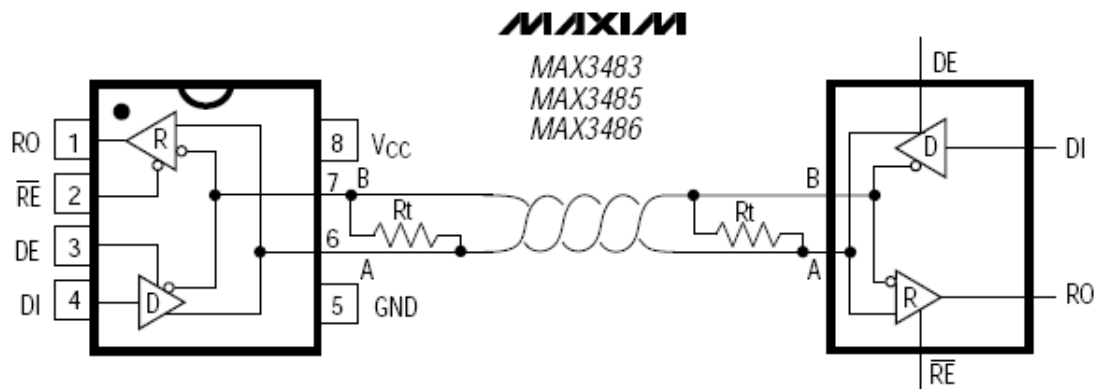


Figura 3.20. Circuit d'operació amb la seva configuració dels pins d'un transceiver Maxim MAX3483. Referència: <http://www.datasheetcatalog.org/datasheet/maxim/MAX3483-MAX3491.pdf>.

Les interfícies del RS-485 van implementades amb *transceivers* del tipus Maxim MAX3483. Es tracta d'un transceptor per a comunicacions RS-422 i RS-485 que conté un *driver* i un *receiver* per cada CI.

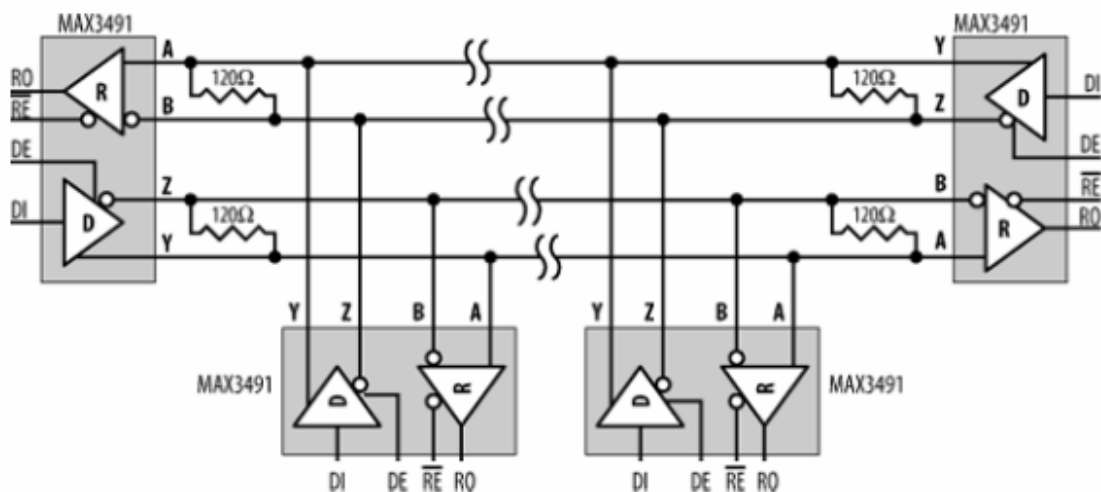


Figura 3.21. Circuit de comunicació full-duplex RS-485 mitjançant transceivers MAX3491. Referència: "Designing Embedded Hardware", John Catsoulis, ed. O'Reilly.

A la figura 3.21 es pot observar un circuit *full-duplex* RS-485 implementat amb *transceivers* MAX3491. Es connecten resistències de final de línia (entre els dos fils del parell trenat) de  $120\Omega$ , per a evitar reflexions, especialment en llargues distàncies.

## Wireless LAN

La família d'estàndards de comunicacions IEEE 802.11 defineix els principals components d'un sistema Wireless LAN. Aquests components poden ser software o hardware. Les arquitectures i targetes embedded poden implementar components 802.11 de diferent manera. Avui dia, a la majoria de les plataformes els estàndards 802.11 són implementats gairebé totalment amb hardware. Els components hardware d'un sistema Wireless LAN corresponen al nivell físic del model OSI; i el software necessari per habilitar les funcionalitats de 802.11 pertanyen al nivell d'enllaç de dades.

Els mòduls Wireless LAN poden estar implementats dins una àmplia varietat de formats, incloent PCMCIA, Compact Flash, i formats PCI. En general, les targetes embedded necessiten funcionalitats 802.11 integrades al xip mestre o en un controlador esclau per a donar suport a un dels connectors estàndard, on estan connectats els altres formats (PCI, PCMCIA, Compact Flash, etc.). Això significa que qualsevol fabricant de *chipsets* de 802.11 pot incloure el *firmware* de les seves *PC cards* per a una solució embedded 802.11 concreta, per a un petit volum de dispositius; o a una *PC card* estàndard, amb el mateix *chipset* del fabricant, per a dispositius fabricats en un major volum.

Per a l'integració d'un chipset de 802.11, el disseny d'una targeta embedded necessitarà tenir en consideració els requeriments wireless LAN de transmissió de senyals i d'ubicació de l'antena.

El dissenyador haurà d'assegurar que no hi ha cap obstacle per a la correcta transmissió i recepció de dades. Si la *master CPU* no integra el 802.11, també s'haurà de dissenyar l'interfície de connexió entre la CPU i el hardware de la targeta 802.11.



Figura 3.22. Targeta 802.11g Wireless PCI ASUS WL-138G de 54Mbps. Referència: [www.asus.com](http://www.asus.com).

## IrDA

IrDA és un estàndard de comunicacions que especifica la forma de transmissió i recepció de dades per mitjà de rajos infrarojos. Va ser creat al 1993 per HP, IBM i Sharp, entre d'altres. Actualment, al voltant de 150 companyies formen el consorci “*Infrared Data Association*”, amb que mantenen i desenvolupen l'estàndard.

El principal propòsit de IrDA és proporcionar comunicacions dispositiu-a-dispositiu de distàncies curtes, sense la necessitat de cables. És un protocol molt emprat per a dispositius portàtils com PDAs, PALMs i telèfons mòbils, del que podem trobar tota la informació necessària a la pàgina web del consorci [www.irda.org](http://www.irda.org).

És un protocol punt-a-punt que emprava transmissions sèrie asíncrones de distàncies curtes. L'especificació original IrDA 1.0 suporta velocitats entre 2,4 Kbps i 115,2 Kbps a distàncies d'un metre; no obstant, alguns *transceivers* IrDA ofereixen aquest rendiment a majors distàncies. Inicialment, les comunicacions IR es donen a 9,6 Kbps, i els dispositius negocien la pujada o baixada de velocitat, segons les capacitats i necessitats de la comunicació. Això significa que l'usuari no necessita establir ni conèixer el *bit rate*, a diferència que en comunicacions RS-232, per a establir la connexió.

La figura 3.23 mostra els “angles de visió” que marca l'especificació IrDA per a una distància d'un metre.

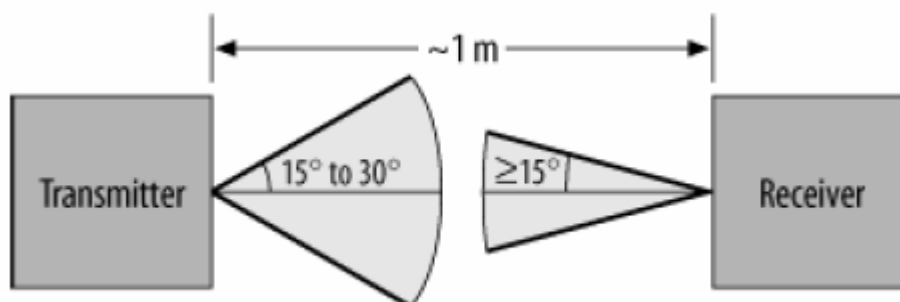


Figura 3.23. Angles de visió IrDA per a distàncies de un metre. Referència: “*Designing Embedded Hardware*”, John Catsoulis, ed. O'Reilly.

### 3.3.3 - Comunicacions Paral·lel

Les comunicacions paral·lel permeten la transmissió de múltiples bits simultàniament. Les interfícies de comunicació en paral·lel gestionen la transferència i recepció de dades entre la CPU mestre i qualsevol controlador o dispositiu d'E/S. Són els responsables de codificar/decodificar les dades i inclouen també *buffers* de recepció i transferència de dades.

D'igual forma que en les sèrie, les comunicacions paral·leles generalment difereixen entre sí en la direcció en la que les dades poden ser transmeses i en la relació temporal de les transferències. En el cas de les direccions de transmissió, d'igual forma que en les E/S sèrie, les paral·leles utilitzen els modes *simplex*, *half-duplex* i *full-duplex*. D'igual forma també, els dispositius d'E/S paral·lel poden transmetre dades síncronament o assíncronament. No obstant, les comunicacions paral·leles tenen l'avantatge de poder transmetre i rebre múltiples bits simultàniament.

Controladors IEEE 1284 (per a impressores i displays), ports DRT, i dispositius SCSI (per a dispositius d'emmagatzament) utilitzen comunicacions paral·leles.

### Dispositius d'E/S gràfics

Tècnicament, els models i imatges creats, emmagatzemats i manipulats als sistemes embedded són els gràfics. Els tres components principals dels gràfics d'E/S a targetes embedded es mostren a la figura 3.24.

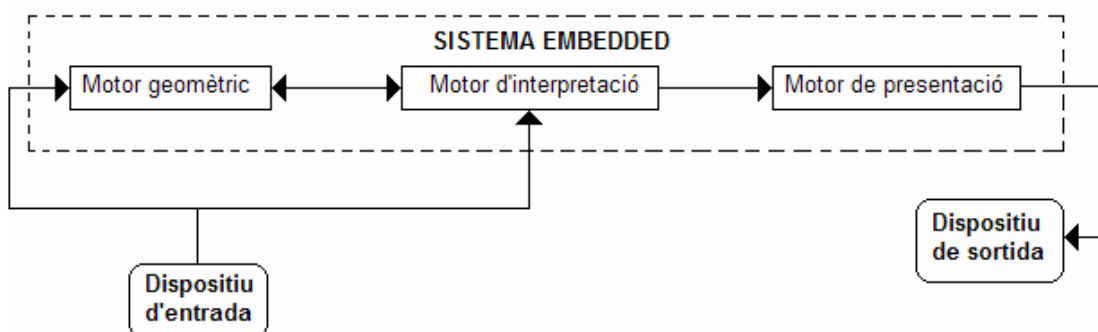


Figura 3.24. Components gràfics d'un sistema embedded. Referència: "Embedded hardware, know it all", Jack Ganssle, ed. Newnes, 2008

El motor geomètric és el responsable de definir els models de color, la geometria física i les propietats lluminoses i materials dels objectes. El motor d'interpretació (*rendering engine*) és el responsable de capturar la descripció dels objectes; proveint funcionalitats de suport per a les transformacions geomètriques, projeccions, mapejat, sombrejat i il·luminació. El motor de trama i presentació és, llavors, el responsable de la presentació física de l'objecte; allà on entra el hardware de sortida.

Els dispositius gràfics de sortida són principalment pantalles i/o impressors i treballen amb interfícies en paral·lel. Les configuracions de ports paral·lels actuals difereixen d'estàndard a estàndard en termes de senyals i cables requerits.

### 3.3.4 – Ethernet

#### Per què utilitzar Ethernet?

Les comunicacions mitjançant Ethernet han estat emprades tradicionalment per a connectar estacions de treball d'empresa i per a transferir dades en temps no real. L'èxit d'Ethernet en el món *desktop* és a causa de la seva senzillesa, expandibilitat, robustesa, i econòmica implementació. Basades en l'èxit d'Ethernet com a xarxa de dades, les xarxes de comunicació embedded de temps real suau van començar a aplicar l'estàndard Ethernet de 10 Mbit/s per motius d'economia, familiaritat, i compatibilitat amb les xarxes d'empresa.

A partir d'aquest moment, començaren a sonar amb força paraules com *Embedded Ethernet* o *Embedded TCP/IP* i, actualment, ja són possibles implementacions TCP/IP amb reduïts requisits d'emmagatzament i de potència computacional, exigències pròpies dels sistemes embedded.

El mètode ideal per a resoldre la interconnexió de diferents sistemes embedded, i de permetre alhora que aquests puguin ser globalment accessibles per a les xarxes d'empresa, és la utilització de protocols de xarxa ja existents (per exemple, TCP/IP), de manera que no sigui necessària la programació del mateix, reduint tant els costos com el temps de desenvolupament.





Figura 3.25. Targeta PCB Ethernet model ETHERNUT 1. Amb microcontrolador de 8 bits ATMEGA 128 treballant a 14.7456 MHz, controladora Ethernet de 10 Mbit/s REALTEK i 32 KB de SRAM externa. Referència: <http://www.ethernut.de/>.

Al mercat hi ha disponibles SBC amb implementacions de l'especificació original *Ethernet* a 10Mbits/s i *FastEthernet* a 100Mbits/s. En aquest últim cas es requereix major potència de processament, i sol ser necessari un bus de tipus PCI, que és possible trobar inclús als SBC del format més petit, com el PC/104.

## Ethernet als sistemes embedded

Ethernet és el protocol LAN més àmpliament utilitzat. Està basat principalment en la família d'estàndards 802.3, que defineixen els principals components que formen un sistema Ethernet. Diferents arquitectures i sistemes embedded poden implementar components Ethernet de diferents maneres. No obstant, a la majoria dels casos Ethernet és implementat quasi totalment amb hardware. El hardware correspon al nivell físic del model OSI i, el software necessari per habilitar les funcions d'Ethernet correspon a la capa o nivell d'enllaç de dades.

Els dispositius Ethernet es connecten en xarxa a través dels cables Ethernet, que poden ser del tipus coaxial, parells trenats, fibra òptica, etc. Els cables utilitzats estan referenciats segons tres components: el ràtio de transmissió de dades, el tipus de senyal utilitzada i el tipus o la longitud del cable utilitzat.

Per exemple, un cable 10Base-T té un ràtio de transmissió de dades de 10 Mbps, empra només senyals Ethernet i és del tipus parells trenats. El cable 10Base-F, en canvi, té les mateixes característiques de funcionament, però és un cable de fibra òptica.

El tipus de cable que es connectarà a la targeta d'E/S és, de fet, el que determinarà si la transmissió Ethernet serà sèrie o paral·lela. El port de comunicacions de la targeta d'E/S per on es connectarà el cable s'anomena MDI (*Medium Dependent Interface*). Existeixen diversos MDI per als diferents tipus de cables Ethernet. Per exemple, el cable Base10-T té un MDI del tipus jack RJ-45.

Alguns dispositius LAN implementen diferents combinacions de components Ethernet i assolixen grans velocitats de transmissió. És el cas de l'IEEE 802.3u *Fast Ethernet* (de 100Mbps) i de l'IEEE 802.3z *Gigabit Ethernet* (de 1000Mbps).

### 3.3.5 – Funcionament de les E/S

És un dels aspectes més importants d'un disseny embedded, ja que poden alterar negativament el funcionament del sistema esdevinguent “el coll d'ampolla” del sistema. s'ha d'entendre el funcionament de cada dispositiu d'E/S de forma individual. Sobretot, en el cas de dissenys propietaris, el dissenyador haurà de tenir en consideració cada dispositiu d'E/S per a prevenir aquestes situacions. Alguns dels aspectes més importants del funcionament compartit d'E/S que poden afectar al sistema són:

- **Les velocitats de transmissió dels diferents dispositius d'E/S.** Diferents dispositius d'E/S, connectats a una mateixa targeta, poden variar en tasses de transmissió de dades d'uns pocs caràcters per segon (com els teclats o ratolins) fins a varis Mbytes per segon (com en el cas de discos).
- **Sincronització de velocitats entre el processador mestre i les E/S.** Agafant d'exemple els casos de funcionament més extrem, les velocitats de treball entre el processador mestre i les E/S poden variar considerablement. Tenint connectat un dispositiu d'E/S extremadament ràpid, el processador podria no

estar llest a temps per a processar el volum de dades entrant, cada cop que les E/S demanessin la seva atenció. D'altra banda, amb un dispositiu d'E/S processant molt més lentament que la velocitat de transmissió del processador, es podrien perdre les dades transmeses.

- **Comunicació entre el processador mestre i les E/S.** Això varia segons si hi ha un controlador d'E/S que fa d'intermediari entre el processador mestre i les E/S, alliberant la CPU per a processar dades més ràpidament. Respecte el controlador d'E/S, pot variar el comportament del sistema segons si funciona per interrupcions, *pulling* o mapeig a memòria (amb una DMA dedicada també a alliberar de tasques a la CPU). En el funcionament per interrupcions, àmpliament estès, uns dispositius d'E/S poden interrompre'n d'altres o es poden formar cues d'espera de torn; funcionant el sistema de forma eficient.

Per a evitar aquests “colls d'ampolla” els dissenyadors han d'examinar els diferents esquemes de comunicacions entre processadors mestres i E/S per assegurar el correcte funcionament del sistema. S'han de tenir en compte certs aspectes, com el temps de resposta o d'execució dels diferents dispositius. Idealment, es realitzarà el disseny tenint en compte que el dispositiu de menor rendiment serà el que determinarà el rendiment global del sistema.

### 3.3.6 – Busos interns

Els principals components d'un sistema (processador, memòria, E/S) són connectats a la targeta embedded mitjançant, com a mínim un bus. No obstant, targetes més complexes poden integrar varis bussos; amb *bridges* (ponts) instal·lats a la targeta que interconnecten els diferents busos entre sí. Els *bridges* poden proveir automàticament informació de mapejat d'adreces, quan les dades són transmeses d'un bus a un altre; o implementar diferents requeriments de senyals de control de diversos busos. Els busos interns poden classificar-se en les tres categories que es descriuen a continuació:

- **Busos de sistema:** També anomenats busos principals, locals o de processador-memòria, interconnecten la memòria externa principal i el *cache*

a la *master* CPU o als ponts cap als altres busos. Són normalment més curts i ràpids que els busos convencionals.

- **Busos de backplane:** Interconnecten la memòria, el processador mestre i les E/S en un únic bus. També anomenats busos d'expansió, externs o de *host*, actuen com a extensions del bus de sistema, connectant la resta de components a la *master* CPU, als *bridges* d'interconnexió de busos i/o el mateix sistema embedded a un port de comunicacions.
- **Busos d'E/S:** Acostumen a ser busos estandarditzats, que poden ser petits i ràpids com PCI o USB, o més grans i lents com SCSI.

La principal diferència entre els busos d'E/S i els busos de sistema és la possible presència de senyals de control IRQ (*interrupt request*) als busos d'E/S. Una línia IRQ habilita els dispositius d'E/S al bus, per a indicar al processador mestre que un esdeveniment està tenint lloc o que una operació ha estat completada. Diferents busos d'E/S tenen diferents comportaments en els mecanismes d'interrupció. Un bus ISA, per exemple, necessita que cada targeta que genera una interrupció tingui assignat un valor únic d'IRQ (configurant els *switches* o *jumpers* de la mateixa targeta). El bus PCI, d'altra banda, assigna un mateix valor d'IRQ a més d'una targeta d'E/S.

Dins d'una mateixa categoria de busos, aquests poden ser dividits entre expandibles o no expandibles. Un bus expandible (PCMCIA, PCI, IDE, SCSI, USB, entre d'altres és aquell al qual se li poden afegir i connectar components addicionals a la targeta. Els busos no expandibles (DIB, VME i I2C en són exemples) no permeten la addició i connexió de components addicionals a la targeta i comunicar aquests a través del bus amb els altres components.

Tot i que els sistemes que implementen busos expandibles són molt més flexibles, aquests també tendeixen a ser més cars d'implementar. Si la placa no està inicialment dissenyada amb tots els possibles components que pugui necessitar en un futur, el funcionament es veurà afectat negativament per la addició d'excessius components addicionals.

## 3.4 – EL SISTEMA OPERATIU

### 3.4.1 – Què és el Sistema Operatiu?

El sistema operatiu és aquell conjunt de programes (software i firmware) responsable de gestionar els recursos en un ordinador, posant a disposició de l'usuari tot el seu poder computacional. En altres paraules, el sistema operatiu constitueix una capa d'abstracció que permet als programes de l'usuari accedir als recursos del sistema, utilitzant els serveis que ofereix el propi SO. D'aquesta manera, l'ordinador és vist per l'usuari com una màquina virtual fàcil d'utilitzar.

Aquest fet és l'impulsor de l'auge que està experimentant la utilització de sistemes basats en PC *embedded* en el desenvolupament d'aplicacions específiques, tant en l'àmbit domèstic com en l'industrial. Els aspectes funcionals que puguin ser resolts pel SO no tindran que ser dissenyats específicament, amb el conseqüent estalvi de temps i costos de desenvolupament. El SO s'encarrega de la gestió, assignació i maximització de l'ús dels recursos, d'implementar les funcions de control necessàries per a fer operatius els programes d'usuari i els dispositius d'E/S, i d'incorporar funcions de gestió de la memòria, la CPU i la E/S; tot això amb l'objectiu de facilitar l'ús de la màquina, resoldre els aspectes generals de control del sistema i aconseguir un funcionament eficient del mateix. Aquesta situació permetrà al dissenyador centrar tota la seva atenció en el desenvolupament del software – i en ocasions del hardware- de l'aplicació que s'està desenvolupant.

Un aspecte interessant del sistema operatiu és la modularitat, que permetrà incorporar a l'aplicació concreta aquelles funcionalitats que són estrictament necessàries per a la mateixa. No s'ha d'oblidar que en la majoria dels casos, el PC *embedded* no incorpora sistema d'emmagatzament de massa i, per tant, el SO ha de residir en memòria del semiconductor, raó per la qual interessa que ocupi el menor espai possible.

Algunes de les característiques que incorpora el sistema operatiu són les següents:

- **Multitasca.** Realitza varis procediments per a funcions com la creació de noves tasques; o decidir quina tasca s'ha d'executar en cada moment, proporcionant mecanismes per a que el processador pugui canviar d'una tasca a una altra.
- **Proporciona una interfície de màquina virtual** als dispositius d'E/S connectats al sistema. Això significa que el sistema operatiu conté un software per a comunicar aplicacions d'usuari al hardware. A més, el sistema operatiu ha de controlar l'accés a aquest hardware, tenint en compte que diverses tasques poden voler accedir al mateix perifèric en el mateix moment.
- **Proporciona funcions de gestió de memòria.** Aquestes funcions permeten l'assignació de memòria a tasques d'usuari.
- **Proporciona software de control d'arxius;** el qual habilita aplicacions de manipulació de fitxers a dispositius com discos; sense haver de preocupar-se de l'assignació de blocs físics en el mitjà d'emmagatzament dels arxius.

Els sistemes operatius es poden classificar en dues categories principals segons la naturalesa del seu comportament en vers al temps:

- Sistemes operatius de temps real (RTOS)
- Sistemes operatius de no temps real (NRTOS)

L'elecció d'una de les dues categories dependrà del temps de resposta necessari, a partir de les especificacions de l'aplicació. Si el temps de resposta pot ser gran, un SO de propòsit general. Per assegurar la resposta dins d'interval de temps reduïts, un RTOS. L'elecció del SO es un aspecte clau del desenvolupament de l'aplicació, i ha de realitzar-se després d'establir de forma detallada les especificacions de la mateixa. La oferta de SO per a PC *embedded* és cada cop més amplia i interessant i, en el moment

actual, ja permet cobrir les necessitats de la majoria d'aplicacions en l'entorn industrial, que requereixen sistemes de gama baixa i mitja. Poden trobar-se al mercat sistemes de propòsit general, de temps real i reduïda grandària, desenvolupats especialment per a controlar sistemes *embedded*.

### 3.4.2 - Estructura d'un sistema operatiu

Els sistemes operatius estan basats en una estructura de capes o nivells, on cada nivell té un rol ben definit. L'estructura típica d'un sistema operatiu basat en capes es mostra a la figura 3.26.

La part central anomenada *kernel* és la encarregada de donar suport multitasca al sistema. És a dir, és la part encarregada de canviar d'una tasca a una altra dins del sistema. Normalment està escrit en llenguatge *assembler*, donat que requereix l'accés a funcionalitats de molt baix nivell en el processador. La tendència moderna als sistemes operatius és tenir un "micro-kernel" ("un kernel molt petit", tot i que en alguns casos pot arribar a ocupar 100 kbytes).

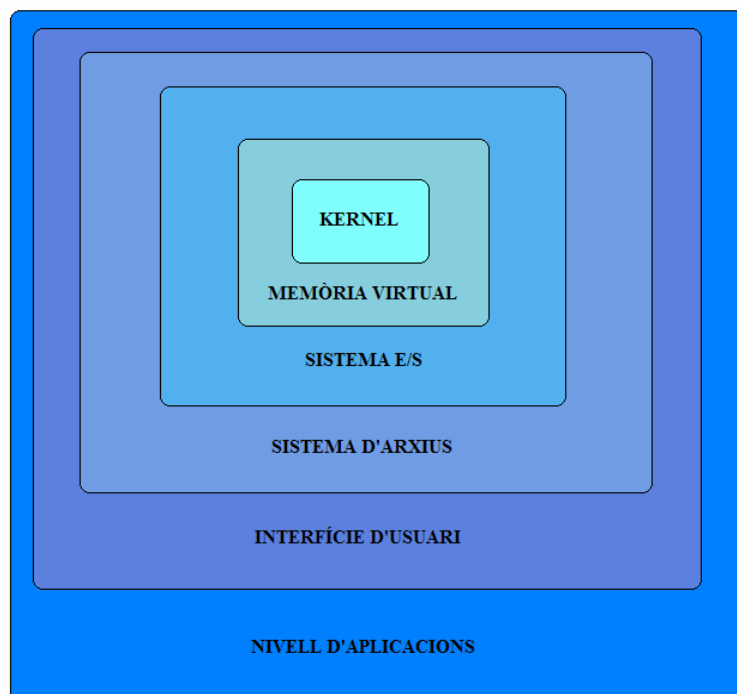


Figura 3.26. Estructura de capes d'un sistema operatiu genèric. Referència: "Introduction to Real-Time Operating Systems", Robert Betz, 2001.

Per sobre del *kernel*, es troba la memòria virtual. És una capa també molt privilegiada en l'accés al processador i s'encarrega de l'assignació de memòria i la protecció del sistema. Les altres capes acostumen a consistir en grans quantitats de codi i proporcionen facilitats a les E/S, al sistema de fitxers, a l'interfície d'usuari, etc. La profunditat d'una capa en un sistema operatiu en mode protegit indica el grau de confiança. Un clar exemple és el nivell d'usuari, amb menys privilegi, per a evitar l'error humà. No obstant, la figura 3.21 ens mostra un model d'estructura de capes molt bàsic. Els sistemes operatius actuals tenen estructures molt més complexes, com veure'm més endavant.

### 3.4.3 – Sistemes operatius de temps real

Els sistemes operatius de temps real (*RTOS: Real-Time Operating Systems*) faciliten la creació de sistemes de temps real, però no garanteixen que el resultat final sigui un sistema de temps real estricte; aquests requereixen un desenvolupament correcte del software. Un RTOS no necessàriament ha de tenir una velocitat de funcionament alta; més aviat, proporciona facilitats, les quals, utilitzades apropiadament, garanteixen la finalització de les operacions dins d'un temps establert. Els RTOS es divideixen en:

- **Soft Real-Time Operating Systems:** Els Sistemes operatius de temps real tou són aquells que finalitzen les operacions GENERALMENT dins de les restriccions temporals marcades.
- **Hard Real-Time Operating Systems:** Els Sistemes operatius de temps real dur són aquells que finalitzen les operacions DETERMINÍSTICAMENT dins de les restriccions temporals marcades.

En poques paraules, un RTOS es valora més per lo ràpidament i/o previsiblement que pot respondre a un esdeveniment particular, que per la quantitat de treball que pot realitzar sobre un període de temps donat.



Els factors clau en un RTOS són, per tant, la latència mínima d'interrupció (*minimal interrupt latency*) i la latència mínima de commutació de fil (*minimal thread switching latency*).

La *latència d'interrupció* és el temps transcorregut entre el moment en que és generada una interrupció, per part d'un dispositiu, i el moment en el que aquesta interrupció és atesa pel sistema.

La *latència de commutació de fil* és el temps que necessita el sistema operatiu per canviar de la CPU a un altre fil.

Una altra classificació que es pot donar per als sistemes operatius es la següent:

- *Sistema operatiu estàtic*: Un sistema operatiu estàtic és aquell en el que el nombre de tasques pot ser definit, de forma exacta, abans de que s'executin. Per tant, quan s'inicia el sistema es crea el nombre de tasques a realitzar, i aquest nombre no canvia.
- *Sistema operatiu dinàmic*: És aquell en el qual el nombre de tasques a efectuar-se no se sap a priori. Les tasques són creades i destruïdes mentre el sistema està operant. Un clar exemple d'això el trobem als sistemes informàtics de temps compartit, en el que el nombre de tasques canvia de forma indeterminada segons els usuaris van entrant i sortint del sistema.

Moltes aplicacions de temps real dels sistemes operatius *embedded* van emmagatzemades en una ROM. Com el SO i el codi estan emmagatzemats en una ROM, les tasques no poden ser creades o esborrades. Per tant, molts sistemes de temps real són estàtics. Evidentment hi ha moltes excepcions a aquesta afirmació, i es poden trobar RTOS dinàmics en aplicacions que requereixin una interactivitat amb l'usuari.

### 3.4.4 -Selecció d'un RTOS

El tractament de sistemes amb un RTOS no és nou. A aquestes alçades, existeixen ja aproximadament uns dos-cents productes al mercat. A vegades la diferència entre uns i altres pot ser molt evident i, en d'altres ocasions pot arribar a ser molt subtil. Estudis detallats del mercat revelen algunes clares tendències en els factors que determinen les decisions finals de compra de productes RTOS.

- **Hard real time:** un dels factors més importants. No cal que siguin “molt ràpids”, sinó lo “suficientment ràpids”. Es valora molt que siguin previsibles i deterministes.
- **Royalty free:** La idea de llicenciar qualsevol software i pagar cada cop que en treus un benefici del mateix resulta poc atractiva. Per a grans volums, en particular, un model royalty-free és ideal. La tendència actual marca models de negoci flexibles, on el mercat dels RTOS també hi té un paper a jugar.
- **Support:** Un RTOS modern és un producte altament sofisticat. La disponibilitat d'un servei de suport tècnic no és gaire negociable, és un requisit quasi obligatori.
- **Eines:** la disponibilitat d'eines de suport al software són molt importants. A totes les webs de productes RTOS fan un especial èmfasi en aquesta qüestió.
- **Facilitat d'utilització:** Que es tracti d'un sistema operatiu fàcil d'utilitzar resulta molt atractiu per als potencials compradors. En realitat, programar sistemes de temps real no és una tasca precisament fàcil. El venedor pot ajudar en aquesta missió, subministrant codi llegible i comentat, integrant-ho dins del mateix paquet.
- **Connectivitat:** Quan parlem de sistemes *embedded*, estem parlant de sistemes interconnectats, on el concepte *networking* és molt important.

- **Suport ampli a la CPU:** El suport, per part d'una arquitectura de RTOS donada, a un ampli rang de microprocessadors és un gran benefici. Aquest suport augmenta el rendiment del sistema i fa més portable el codi. Això allibera als enginyers de disseny d'aquestes qüestions vitals, i accelera el temps de sortida al mercat del producte.

L'elecció del sistema operatiu és un factor clau en el disseny d'un sistema embedded. Per a aplicacions embedded, existeix cada cop al mercat una més amplia oferta de sistemes operatius. A continuació es descriuen algunes alternatives interessants:

- **Linux** ([www.linux.org](http://www.linux.org)): Diferents versions del SO Linux enfocades a sistemes embedded.
- **VxWorks** ([www.windriver.com](http://www.windriver.com)). RTOS líder al mercat. Les seves capacitats multitasca i *scheduling* proporcionen un entorn de gran eficiència en l'execució de codi en temps real.
- **QNX** ([www.qnx.com](http://www.qnx.com)). Competidor directe de VxWorks en el mercat de RTOS.
- **Windows Embedded CE** ([www.microsoft.com/windows/embedded/ce](http://www.microsoft.com/windows/embedded/ce)). Sistema operatiu de 32 bits per aplicacions *embedded* d'altas prestacions. Plataforma modular dissenyada per a intensificar i estendre capacitats de SO *embedded*.
- **Windows XP Embedded** (<http://www.microsoft.com/windowseembedded/en-us/products/wexpe>). Versió componentitzada de Windows XP Professional amb grans facilitats per al desenvolupament amb les últimes tecnologies.

### 3.4.5 - Linux

Tot i que el sistema operatiu Linux original està dissenyat per a treballar sobre plataformes PC desktop, als últims anys, Linux ha tingut especial atenció al món *embedded*, i com a resultat es poden ja veure funcionaments específics *embedded* a les corrents principals del codi Linux. Per exemple, les versions més corrents del *kernel* Linux suporten sistemes d'arxius basats en ROM i en nombrosos formats de tecnologia Flash; incloent NAND Flash i M-Systems DiskOnChip.

No es tracta d'un sistema operatiu de temps real en el sentit estricte; però es pot fer treballar com a tal, per a petites restriccions o requisits temporals, mitjançant una sèrie de funcionalitats *real-time* incorporades al seu *kernel* original. Els dissenyadors acostumats a treballar amb veritables sistemes operatius de temps real quedaran desencisats amb el tractament per part de Linux d'aquesta qüestió; però per a moltes aplicacions no crítiques, utilitzant simplement un processador suficientment ràpid i gestionant bé les tasques secundàries serà suficient.

La diferència entre aquest i un RTOS veritable és que Linux necessitarà l'addició d'APIs (*Application Programming Interface*) per a garantir el funcionament en temps real. Per exemple, una API garantirà que una interrupció de nivell 0 serà atesa en un màxim de 2 ms des del moment en que el hardware rebí la petició d'interrupció.



Si és vol una versió de “temps real” de Linux, existeixen varies opcions, però probablement la més coneguda és la distribució comercial anomenada Hard Hat Linux de Monta Vista Software ([www.hardhatlinux.com](http://www.hardhatlinux.com)). Monta Vista també crea versions d'aquest software per a la indústria de les telecomunicacions. Una altra opció que es pot descarregar gratuïtament és el RTLinux (disponible a [www.fsmlabs.com](http://www.fsmlabs.com)). També, per a rebre suport és interessant visitar el lloc web [www.linuxdevices.com](http://www.linuxdevices.com), on hi ha multitud de fabricants anunciant productes per a aquesta plataforma.

Un dels grans avantatges de Linux és que no només pot ser portat a una gran varietat d'arquitectures; sinó que el procés d'instal·lació per a moltes plataformes de referència està bastant ben documentat. A més, es pot descarregar el seu *kernel* precompilat amb precises instruccions que ajuden a estalviar un enorme temps i feina a l'inici d'un projecte.

Un altre gran benefici d'utilitzar Linux Embedded, un SO de codi obert (*open source*), en vers a un RTOS tradicional, és que la comunitat de desenvolupament (*development community*) de Linux ofereix suport a més *drivers* de dispositius i protocols que la resta de fabricants; com per exemple, els *drivers* de la targeta d'interfície de xarxa (NIC) o *drivers* de ports sèrie i paral·lel.

També s'ha de destacar que Linux pot funcionar en la majoria dels microprocessadors amb una gran varietat de perifèrics i dispositius embedded, a causa del seu suport de sistemes de multiprocessador, que li dona una alta escalabilitat. Aquesta capacitat li dona a un dissenyador l'opció d'executar una aplicació de temps real en un sistema de doble processador, augmentant el poder computacional. Així, per exemple, és possible controlar un sistema en un processador, mentre en l'altre s'executa la interfície gràfica d'usuari (*GUI: Graphical User Interface*) simultàniament.

Existeixen ja diferents tipus de sistemes embedded Linux, dels quals es nombren els principals a continuació:

- **ETLinux:** Una distribució completa Linux dissenyada per a treballar a petits computadors industrials, especialment a mòduls PC/104. Adreça web: <http://www.etlinux.org/>.
- **RTLinux:** Versió Linux amb la interfície de programació d'aplicacions (*API: Application Programming Interface*) variada per al seu funcionament com a SO de temps real. Pàgina web: <http://www.rtlinuxfree.com/>.
- **Hard Hat Linux:** Versió amb estructura RTOS de Linux específica per a sistemes embedded. Pàgina web: <http://www.mvista.com/>.

## WIND RIVER

### 3.4.6 – VxWorks

Es un dels RTOS més prestigiosos i alhora més cars del mercat embedded. Es tracta d'un sistema operatiu de temps real, basat en Unix i desenvolupat per Wind River Systems. Aquest entorn d'execució i desenvolupament de temps real i elevada complexitat, comprèn un RTOS d'altres prestacions que s'executa en el sistema *target*, un conjunt de potents eines per al desenvolupament en el *host* i un ampli rang d'opcions software per a la comunicació entre el *target* i el *host* (línia sèrie, Ethernet, etc.).

VxWorks suporta una àmplia varietat d'estàndards industrials com POSIX (*Portable Operating System Interface for Unix*), ANSI c i TCP/IP. Inclou un *micro-kernel* dissenyat per a reduir la càrrega de la CPU, proporcionant una ràpida resposta a esdeveniments externs i operant de forma determinista. El seu *kernel* multitasca, amb planificador *preemptive* (els processos poden prendre la CPU arbitràriament), proporciona un àgil canvi de context, una ràpida resposta a les interrupcions, i una àmplia comunicació i sincronització entre processos.

Las característiques més destacables de VxWorks son:

- Capacitats multitasca i planificació de la CPU que possibiliten la creació d'entorns de temps real molt eficients en l'execució del codi. Ideal per al desenvolupament d'aplicacions amb fortes restriccions temporals.
- Augment al suport de SMP per a l'optimització del funcionament de la CPU.
- Eines de desenvolupament per a la depuració i desenvolupament de sistemes SMP i AMP a processadors multi-nucli.
- Suports per a login remot, RPC (*Remote Procedure Calls*), NFS (*Network File System*) tant per al servidor com el client i execució remota de comandaments.

- Augment de capacitats per a xarxes 4G per aplicacions WiMAX i LTE (*Long Term Evolution*).
- Optimitzacions de hardware per a processadors mono-nucli i multi-nucli.
- Possibilitat de comunicació a través de diferents mitjans com Ethernet, línia sèrie, *backplane* i altres interfícies de xarxa.
- Suports per a *login* remot, RPC (*Remote Procedure Calls*) , NFS (*Network File System*) tant per al servidor com el client i execució remota de comandaments.
- Incorpora eines per a diagnosi del sistema, monitorització de prestacions, facilitats de prototipat i muntatge i càrrega dinàmica.
- Pot executar-se a pràcticament totes les CPU modernes del mercat de sistemes embedded; incloent la família de CPU x86, MIPS, PowerPC, SH-4, ARM, StrongARM y xScale.
- Permet als dissenyadors de sistemes seleccionar la configuració de disseny multi-nucli òptima, multiprocessament simètric (SMP) o asimètric (AMP).
- Reduït consum de recursos.
- Ofereix versions específiques per als sectors automobilístic, de l'electrònica de consum, d'aplicacions industrials i dispositius de xarxa.

La última distribució de propòsit general de Wind River és la versió VxWorks 6.7, pensada per a gestionar la nova generació de dispositius amb els més alts requeriments de funcionament, mantenint o inclús reduint el consum de recursos.

### 3.4.7 – QNX

Sistema operatiu de temps real desenvolupat per QNX Software Systems Ltd. Està constituït per un autèntic *micro-kernel* rodejat d'un conjunt de mòduls opcionals (*resource managers*) que poden ser inclosos segons les necessitats de l'usuari.

Cada driver, aplicació, *protocol stack* i sistema d'arxius s'executen fora del *kernel*, a l'espai d'usuari de memòria protegida. Això fa que la falla de qualsevol component no afecti el *kernel* i, per tant, la fiabilitat del sistema.



QNX Neutrino (la seva última versió) pot acabar un procés amb un component en falla i refer el sistema, sense la necessitat de reiniciar-lo. Els seus algoritmes de planificació són similars als emprats per VxWorks, reunint les característiques pròpies del emprats pels RTOS de més altes prestacions. QNX ofereix tots els avantatges d'un RTOS d'aplicació en sistemes embedded: mida reduïda (permet la ubicació del SO en memòria ROM), escalable, extensible i ràpid. Una configuració reduïda de QNX, incorporant el *micro-kernel* i algun mòdul opcional bàsic, poc ocupar al voltant de 64KB de memòria.

Les principals característiques de QNX Neutrino són les següents:

- Processos de *watchdog* per a aplicacions de monitorització i recuperació de processos de comunicacions, amb *drivers* de reiniciació de dispositius i serveis de sistemes operatius
- Suport a tecnologies networking IPv4, IPv6, IPSec, FTP, HTTP, SSH i Telnet entre d'altres.
- Completa interfície gràfica d'usuari Photon microGUI.



- Eines de visualització per a la traça d'esdeveniments del sistema, incloent interrupcions, sincronització i utilització de la CPU
- Alta escalabilitat i capacitats incorporades de multiprocessament multi-nucli.
- Gran portabilitat amb el suport extensiu de l'estàndard POSIX , que permet la ràpida migració de Linux, Unix i altres programes de codi obert.
- Suport en temps d'execució i BSP per als principals chipsets, incloent MIPS, PowerPC, SH-4, ARM, StrongArm, XScale, i x86.

### 3.4.8 – Windows Embedded CE

Windows Embedded CE és un sistema operatiu de 32 bits, de temps real dur, desenvolupat per Microsoft per a satisfer les necessitats de portàtils, mòbils, i dispositius *embedded*.

Amb suport per a múltiples arquitectures de processador (x86, MIPS, ARM i SH4), Windows Embedded CE pot ser adaptat a una gran varietat de dispositius com Smartphones, PocketPCs, càmeres digitals, caixers automàtics, *routers* de xarxa, projectors wireless, computadors industrials, etc.



Alguns dels dispositius, com els Smartphone o els PocketPC, tenen una interfície molt visual. D'altres aplicacions, en canvi, no són visibles al usuari final. Aquests dispositius són construïts per a proporcionar un conjunt de funcions concretes, com és el cas d'instruments de prova, dispositius d'adquisició de dades, computadors industrials, entre d'altres.

La tendència dominant als anys 90 en el desenvolupament de sistemes *embedded* era la utilització de microprocessadors de 8 o 16 bits amb funcionalitat limitada, requerint l'ús de llenguatges de programació de baix nivell. La ràpida evolució de la tecnologia del

silici, va donar entrada a processadors de 32 bits, més petits i ràpids, de forma relativament econòmica.

Una nova generació de hardware de 32 bits és ara fàcilment disponible a baix cost. Les companyies poden desenvolupar nous productes amb característiques addicionals al baixar el temps de desenvolupament d'aquests. Això possibilita l'ús de sistemes operatius de 32 bits com és el cas de Windows Embedded CE.

Windows Embedded CE és una versió minimalista de les versions *desktop* de Windows. La seva nova arquitectura de *kernel*, amb capacitat expandida per a processos, és capaç de funcionar en un mínim espai de memòria. El SO, per tant, podria configurar-se com a un sistema tancat, sense dispositius d'emmagatzematge ni interfície d'usuari; introduït directament en una ROM.

Un altre gran atractiu de Windows Embedded és la possibilitat d'implementar-lo tant a dispositius cablejats com inalàmbrics, oferint suport a connexions PAN (Personal Area Network), LAN (Local Area Network) i WAN (Wide Area Network), incloent Bluetooth, IrDA i WiFi 802.11.

El comportament de Windows Embedded CE és el d'un autèntic RTOS, amb una latència d'interrupció determinista i capaç de gestionar 256 nivells de prioritat. Incorpora, a més, nombroses eines de desenvolupament com Visual Studio, Embedded Visual C++ i Free Pascal.

Ofereix una gran varietat de configuracions, amb la possibilitat d'ajustar el SO a les estrictes necessitats de l'aplicació. Aquesta flexibilitat es fa patent tant en el software com en les formes de distribució del producte. Pot adquirir-se una versió de prova totalment funcional de Windows Embedded CE 6.0, la seva última versió, per un període de 6 mesos a la pàgina web de Microsoft <http://www.microsoft.com/windowsembedded>. A més, disposa d'una extensa comunitat de desenvolupadors que ofereixen suport a les diferents plataformes hardware existents.

## 4. MARC DE DESENVOLUPAMENT

### 4.1 – Desenvolupament d'aplicacions embedded

En aquest apartat s'explica de forma general el procés de disseny d'un sistema embedded. És molt important a les primeres etapes del desenvolupament la correcta elecció dels components hardware i software que composaran el disseny final del dispositiu. En els següents subapartats es dona informació útil sobre aquests aspectes i els elements principals que integren qualsevol disseny d'una aplicació embedded.

#### 4.1.1 – Desenvolupament entre plataformes

Una de les característiques habituals del desenvolupament de sistemes *embedded* és la seva metodologia de desenvolupament del software, anomenada *cross-plataform development*, desenvolupament entre plataformes. Això vol dir que el software d'un sistema *embedded* es desenvolupa en una plataforma i es transfereix a una altra definitiva en la qual funciona. En aquest context, la plataforma és el hardware (com, per exemple, el tipus de processador), el sistema operatiu i les eines de desenvolupament de software que s'empren en el desenvolupament.

El sistema amfitrió, *host system*, és el sistema en el qual és desenvolupa el software embedded. Normalment es tracta d'un computador *desktop*, que se l'acostuma a anomenar estació de treball o *workstation*. El sistema target, *target system*, és el sistema embedded en desenvolupament.

L'eina de software principal del desenvolupament *cross-plataform* és el *cross-compiler*. És un compilador que funciona en una arquitectura concreta de processador, com podria ser la família x86, però produeix codi objecte per a diferents tipus d'arquitectures de processadors. Un exemple d'això el trobem en el compilador DIAB de Wind River Systems. El compilador DIAB funciona sobre un sistema operatiu Microsoft Windows en una arquitectura de processador IA-32; i és capaç de produir codi objecte per nombrosos tipus de processadors, com els Motorola s68000, MIPS, i ARM.

Els components bàsics en l'entorn de desenvolupament de sistemes embedded són el sistema amfitrió, el sistema embedded *target*, i les eines de connectivitat entre les dues plataformes.

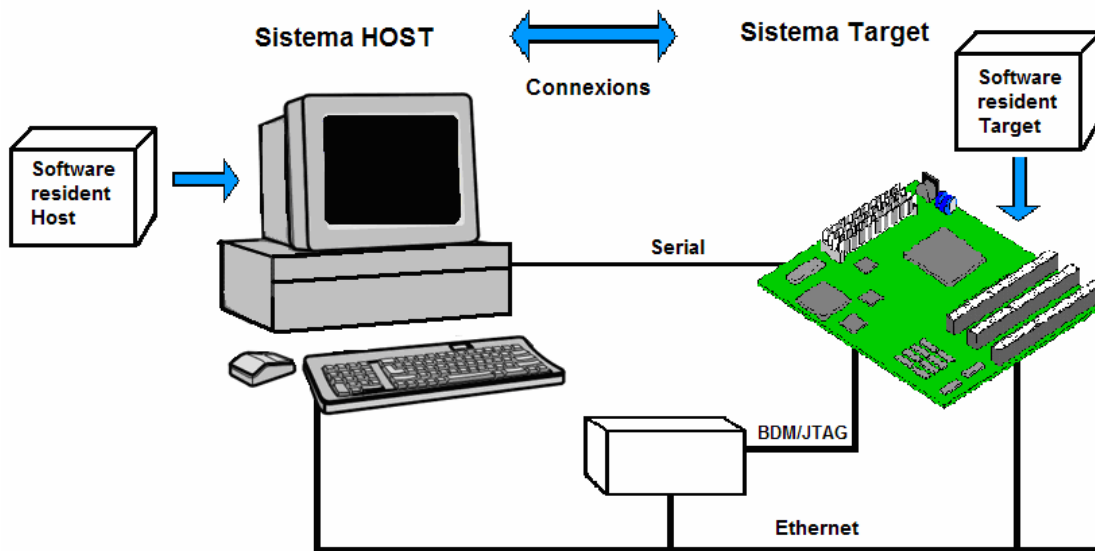


Figura 4.1. Model típic d'entorn de desenvolupament Cross – plataforma. Referències: “Real – Time Concepts for Embedded Systems”, Qing Li i Carolin Yao, ed. CMP Books, 2003.

Les principals eines de desenvolupament que ofereix el sistema *host* són el compilador, el *linker* o enllaçador, i la depuració del codi font. El sistema *target* probablement inclourà un *dynamic loader*, un *link loader*, un monitor, i un agent depurador. El sistema *host* transferirà les imatges de programa al sistema *target*, a través de les diferents connexions entre ambdós (sèrie, Ethernet, USB, etc.). També es pot transferir informació sobre la depuració entre el *host debugger* i el *target debug agent*.

Es desenvolupa el software de sistema, el RTOS, el *kernel* i el codi d'operació; es compila, i s'empaqueta tot plegat, dins una imatge executable. El programador escriu aplicacions que executa en el mateix entorn utilitzat en el desenvolupament; sense la necessitat de traspassar, a cada moment, els canvis creats per a la seva comprovació.

#### 4.1.2 – Emmagatzematge i actualització del software

El software dels sistemes *embedded* es pot dividir en dues parts: software de sistema i software d'aplicació. Aquest és habitualment emmagatzemat a dispositius de memòria

ROM i NVRAM. Actualitzar el software pot suposar crear una nova PROM, desplegar uns equips especials o mètodes especials per a reprogramar la EPROM, o reprogramar la memòria flash.

L'elecció del dispositiu d'emmagatzematge del software té un impacte en el desenvolupament. El procés de reprogramar una EPROM, que pot consumir molt de temps, és habitual durant el desenvolupament. Cal tenir cura, però, de no malmetre la EPROM al treure-la del seu sòcol.

### 4.1.3 – Creació d'un entorn de desenvolupament PC embedded

Un dels primers passos a seguir, a l'hora de crear la nostra pròpia bancada de desenvolupament de sistemes *embedded*, és conèixer els requeriments hardware i software de la mateixa. La nostra bancada de desenvolupament ha de tenir les eines necessàries per a que puguem construir imatges de SO, desenvolupar *drivers*, crear aplicacions; i transferir-ho tot al sistema *embedded target*, amb èxit.

El sistema host és, normalment, un PC de propòsit general amb les interfícies de comunicació necessàries per a comunicar-se amb el sistema *target*. Aquest PC contindrà també un sistema operatiu de propòsit general, com per exemple Windows XP Professional. Per sobre d'aquest SO de propòsit general anirà instal·lat el constructor de plataformes, amb el paquet d'eines de desenvolupament d'aplicacions, que ens permetrà empaquetar tot el software necessari per a fer funcionar el sistema *embedded target*.

S'elegirà un RTOS concret en funció de les necessitats, la relació qualitat/preu i les seves prestacions. Existeix, actualment, una gran quantitat i varietat de RTOS al mercat. De fet, són bastants els factors a tenir en compte en l'elecció d'un RTOS; tal com ja hem pogut veure a la secció 3.4.4 d'aquest projecte. De forma resumida es pot considerar que els principals factors són:

- *Restriccions temporals* del SO necessàries per al sistema *embedded*
- *Suport*: disponibilitat d'un suport tècnic i una comunitat de desenvolupament que ofereixi ajuda als dissenyadors.

- *Eines de desenvolupament*: que permetin construir el software necessari per al funcionament del sistema embedded.
- *Facilitat d'utilització*: que permeti reduir el temps de desenvolupament i fer més còmode la feina dels dissenyadors.
- *Compatibilitat*: amb les diferents arquitectures de processadors i tecnologies *embedded*.

El sistema operatiu elegit condicionarà els requisits hardware i software del sistema *host* de la nostra bancada de desenvolupament. Per exemple, Windows Embedded CE 6.0 requereix com a mínim un processador de 933 MHz i 512 MB de memòria RAM.

Un cop escollit el sistema operatiu, la metodologia que es durà a terme per a la creació del sistema *embedded* és la següent:

- [1] Seleccionar la plataforma hardware.
- [2] Seleccionar els components perifèrics hardware addicionals.
- [3] Adquirir o dissenyar el software de compatibilitat amb la plataforma hardware i els seus perifèrics.
- [4] Desenvolupar el disseny del sistema operatiu.
- [5] Desenvolupar les aplicacions.
- [6] Construir el prototip.
- [7] Realitzar la depuració i el test del sistema.
- [8] Transferir al sistema *target* la imatge final de SO al *target*.

La plataforma software escollida en aquest projecte és Windows Embedded CE 6.0, que es mostra de forma detallada durant el següent apartat del projecte.

## **4.2 – Windows Embedded CE 6.0**

A l'apartat 4.2 d'aquest projecte es dona a conèixer una de les principals plataformes software de sistemes embedded: Microsoft Windows Embedded CE 6.0.

### **4.2.1. Introducció**

Windows Embedded CE no és un sistema operatiu de propòsit general (com Windows XP o Vista). Windows CE està preconfigurat per a configuracions i hardware de dispositius embedded.

El seu disseny modular permet personalitzar imatges de SO per a plataformes hardware de limitades prestacions computacionals, incorporant només allò necessari per a l'aplicació concreta del dispositiu.

A la majoria dels casos, no és necessari modificar configuracions per defecte del dispositiu embedded (BIOS, Jumpers, DMA, IRQ, adreces de memòria i E/S, etc.), ja que la última versió de Windows CE, la 6.0, incorpora suport software a la majoria del hardware de dispositius embedded del mercat.

Aquest sistema operatiu està molt estès al mercat actual, sobretot en el sector de l'electrònica de consum; on dispositius mòbils com PDAs, GPS o consoles tenen una gran presència. Windows Embedded CE pot ser implementat en un caixer automàtic, un reproductor MP4, o inclús en un computador industrial, gestionant la lògica de processos automatitzats.

En aquest apartat es mostra una metodologia de creació d'aplicacions embedded amb Windows CE 6.0. Partint desde zero, podràs conèixer factors clau com el procés d'instal·lació de les eines de desenvolupament, la creació d'una imatge executable de SO i la transferència de la imatge creada al dispositiu target.

### 4.2.2. Evolució de Windows Embedded CE

La primera versió de Windows Embedded CE (CE 1.0) va ser creada per Microsoft l'any 1996. Inicialment va ser creat per a dispositius diferents d'un PC, posicionant-se especialment dins el mercat del Pocket Pc. Al 1997, el sistema (CE 2.0) es tornà componentitzat i dissenyat per a un ampli rang de dispositius i múltiples famílies de processadors. La versió CE 3.0, llançada al 2000, conté suport per a operacions en temps real i tecnologies multimèdia avançades com DirectDraw, DirectShow, i Windows Media Player.

La versió CE 4.0 arribà al 2001. Aquesta versió conté suport per a Direct3D, *Universal Disk File System* (UDFS), *Simple Object Access Protocol* (SOAP) i base de dades SQL Server CE. Els posteriors llançaments 4.1 i 4.2 van proporcionar als dissenyadors funcionalitats d'accessibilitat expandida, afegint suport per a visionar arxius, perfils Bluetooth i IPv6, i suport per a telefonia VoIP (*Voice over Internet Protocol*).

Al 2005, Microsoft llançà la versió CE 5.0 amb suport per a *Universal Serial Bus* (USB) 2.0, *Simple Object Access Protocol Input/Output* (SDIO), Windows Media 9, i Microsoft Explorer 6; així com un sistema constructiu unificat i una infraestructura de desenvolupament general de BSPs amb capa d'adaptació OEM anomenada OAL (*OEM Adaptation Layer*) disponible per al dissenyador.

Amb Windows Embedded CE 6.0, llançat al 2006, l'arquitectura del sistema ha experimentat canvis substancials. Ara cada procés té 2 GB de memòria virtual (anteriorment 32 MB), i el nombre de possibles processos en marxa de forma simultània incrementa a 32000 (anteriorment 32). En versions anteriors, parts del sistema *kernel* eren implementades com un conjunt de processos separats, mentre que a Windows Embedded CE 6.0, estan combinades en un sol *kernel*. Els processos del sistema són DLLs (*Dynamic Link Libraries*) carregades dins l'espai del *kernel*. Això incrementa el rendiment del sistema operatiu, redueix les crides a APIs (*Application Programming Interface*) i unifica la interfície del *kernel*. Ara el dissenyador pot carregar *drivers* a l'espai del *kernel* i també crear *drivers* que es carreguin en un procés especial d'usuari.



### **4.2.3. Instal·lació de Windows Embedded CE 6.0**

Un dels factors més importants per a establir un entorn apropiat per al desenvolupament és tenir les eines instal·lades apropiadament, amb tots els components necessaris.

La incorrecta instal·lació o manca d'eines necessàries per al desenvolupament, pot causar problemes i ocasionar que no es pugui completar la creació de la imatge de SO d'un projecte.

Per això, és necessari seguir de forma precisa la seqüència d'instal·lació del software i mantenir –lo amb les actualitzacions necessàries.

#### **Suport a processadors**

Windows CE està dissenyat per a suportar quatre famílies de processadors (CPUs): ARM, MIPS, SuperH, i x86. Hi ha nombrosos models de processadors d'aquestes famílies disponibles per a diferents fabricants amb diferents nivells de prestacions i preu.

Microsoft proporciona llibreries de suport a les quatre famílies de processadors. Les llibreries de suport a processadors facilitades per Microsoft es troben al directori:

**WINCEROOT\PLATFORM\COMMON.**

Nota: En aquest projecte, es prendrà com a convenció anomenar WINCEROOT al directori arrel de Windows Embedded CE (amb WINCE600 de nom per defecte).

#### **Obtenció del software**

Existeixen diverses alternatives en l'obtenció d'una versió de Windows Embedded CE 6.0. Es pot adquirir el producte amb llicència a través d'un proveïdor oficial; però si no disposes d'una versió llicenciada de Windows Embedded CE, pots obtenir una versió de prova de 6 mesos, emplenant prèviament un formulari de registre, a la pàgina web <http://www.microsoft.com/windowseembedded> . La versió trial disponible a la pàgina web no incorpora les eines de Visual Studio 2005 Professional, necessàries per al desenvolupament; per tant, hauràs d'instal·lar prèviament el Visual Studio 2005 per a poder desenvolupar els dissenys.

## **Requisits mínims per a la instal·lació**

A l'hora de fer la instal·lació de Windows Embedded CE 6.0 s'han de tenir en compte els requisits mínims de la instal·lació. Un aspecte important és l'espai lliure al disc. Microsoft especifica que és necessari tenir 18 GB d'espai lliure per a la instal·lació al disc i 1 GB d'espai disponible al disc dur. A més, cada projecte de disseny de SO requerirà addicionalment 1 o 2 GB d'espai d'emmagatzematge. Els requisits mínims (especificats per Microsoft) per a la instal·lació de Windows Embedded CE 6.0 són els següents:

- Microsoft Windows 2000 Professional amb Service Pack 4 o Microsoft Windows XP amb Service Pack 2.
- Mínim 933 MHz de processador (recomanat 2 GHz).
- Mínim 512 MB de memòria RAM (recomanat 1 GB).
- 18 GB d'espai lliure de disc per a la instal·lació.
- 1 GB d'espai lliure al disc del sistema.
- Dispositiu DVD-ROM.

## **Seqüència d'instal·lació**

La seqüència d'instal·lació adequada per al correcte funcionament de tots els processos és la següent:

[1] Instal·la Visual Studio 2005.

[2] Instal·la Windows Embedded CE 6.0.

[3] Instal·la Visual Studio 2005 SP1

Disponible a l'espai web:

<http://www.microsoft.com/downloads/details.aspx?FamilyID=BB4A75AB-E2D4-4C96-B39D-37BAF6B5B1DC&displaylang=en> .

[4] Instal·la l'actualització de VS2005 per a Windows Vista (en cas de no treballar amb Windows Vista pots saltar aquest pas). Disponible al web:

<http://www.microsoft.com/downloads/details.aspx?displaylang=en&FamilyID=90e2942d-3ad1-4873-a2ee-4acc0aace5b6> .

[5] Instal·la Windows Embedded CE 6.0 SP1. Disponible a la pàgina web:

<http://www.microsoft.com/downloads/details.aspx?displaylang=en&FamilyID=bf0dc0e3-8575-4860-a8e3-290adf242678> .

Nota: Si algun dels enllaços anteriors no estigues operatiu, visita l'espai de descàrregues de Microsoft <http://www.microsoft.com/downloads> i escriu les paraules clau per a la descàrrega.

#### 4.2.4. Terminologia de Windows Embedded CE

Tot entorn de desenvolupament té un conjunt de termes freqüentment utilitzats. Alguns d'aquests són comuns entre varies plataformes i d'altres són exclusius de cada plataforma de desenvolupament.

Per a poder seguir amb normalitat el manual d'utilització de Windows Embedded CE 6.0 elaborat en aquest projecte, és necessari conèixer aquesta terminologia.

Tot seguit es descriuen algunes paraules pròpies de la terminologia de Windows Embedded CE 6.0:

- **BSP (Board Support Package).** És un conjunt de components de software que combinen el codi d'adaptació de capes OEM necessari, *drivers* de dispositius, i arxius de configuració per a una plataforma específica de hardware.

- **OEM Adaptation Layer (OAL).** És el codi de baix nivell que actua d'interfície entre el hardware i el software de Windows Embedded CE.
- **SDK (Software Development Kit).** Conjunt d'eines software per a la creació d'aplicacions amb codi natiu amb Windows Embedded CE.
- **CETK (CE Test Kit).** És una eina per al test i depuració de Windows Embedded CE, integrada a la instal·lació del programa.

#### 4.2.5. Entorn de disseny i eines de desenvolupament

Conèixer l'entorn de desenvolupament és un factor clau a l'hora de desenvolupar projectes de sistemes embedded dins d'una qualitat i temps acceptables.

L'eina per al desenvolupament de dissenys i la creació d'imatges de SO personalitzades amb Windows Embedded CE s'anomena *Platform Builder*. En els llançaments anteriors de Windows Embedded CE, l'eina *Platform Builder* era exclusiva i dedicada al desenvolupament de CE.

Per al llançament de la versió 6.0, el *Platform Builder* de Windows CE és un *plug-in* de Microsoft Visual Studio 2005. Aquest *plug-in* permet construir BSPs, crear *drivers* de dispositius, construir imatges *runtime*, i exportar SDKs per al suport del desenvolupament d'aplicacions.

Al següents subapartats s'explica de forma detallada la utilització dels BSPs (*Board Support Packages*), el procés de creació d'una imatge executable de sistema operatiu i la connexió amb el dispositiu target. Alhora, es mostren els principals menús i eines de l'entorn de desenvolupament de Visual Studio 2005. Per a conèixer millor el *Platform Builder* de Windows Embedded CE 6.0 obre el programa i segueix atentament les explicacions dels propers apartats.

A la part superior de la pantalla estan els menús desplegables; la selecció dels quals varia segons l'entorn actual seleccionat: edició de codi, depuració, etc. D'entrada tots els menús desplegables estàndard estan habilitats. Altres menús com *Generar* o *Depurar* no apareixen fins que no es selecciona un projecte.

La secció esquerra de la pantalla conté l'espai de treball o *workspace*, amb diverses vistes. Per defecte en té de seleccionades: *Explorador de soluciones*, *Catalog Items View* i la *Vista de classes*. La part dreta de la pantalla conté l'editor del codi font.

La secció inferior de la pantalla conté varies finestres, incloent la finestra *Resultados* (que conté les opcions *CE Debug* i *Windows CE Log*), *Ventana Definición de código* i *Examinador de llamadas*.

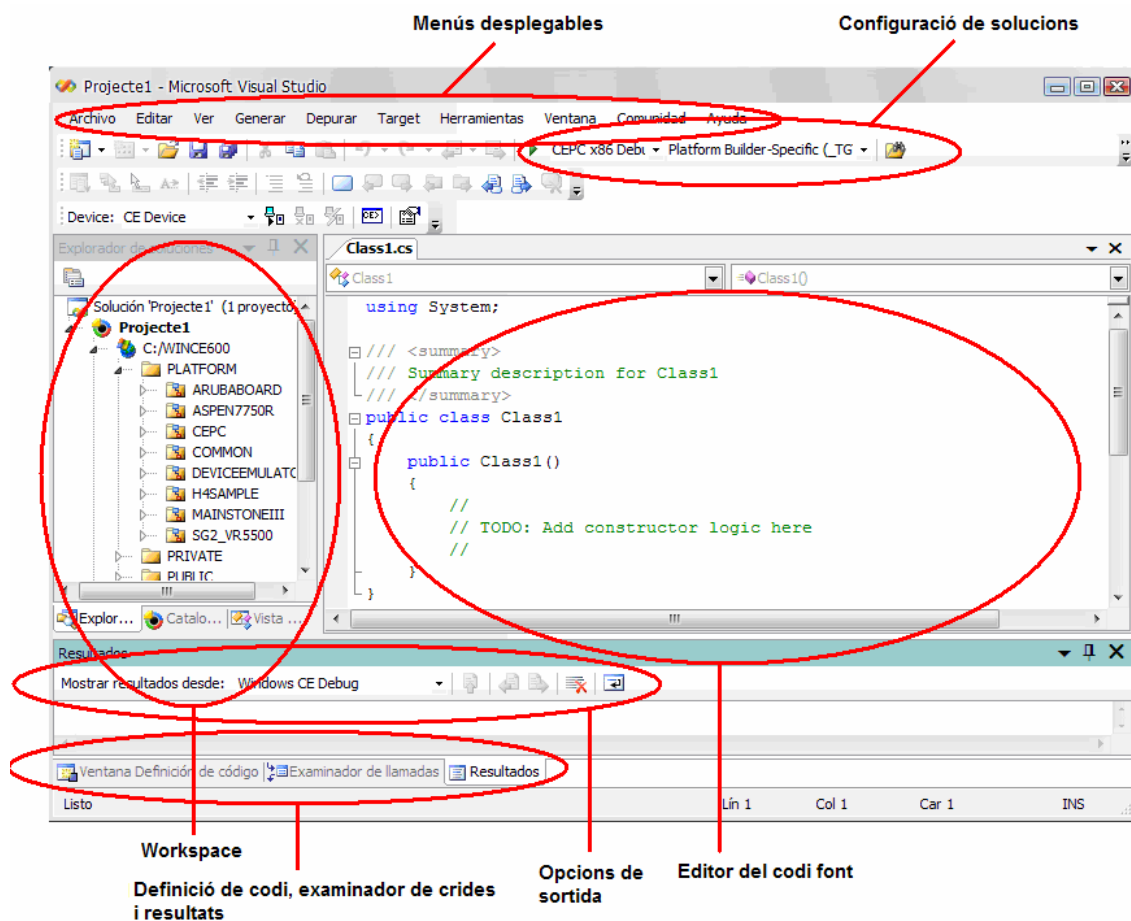


Figura 4.2. Vista general de Microsoft Visual Studio 2005 Professional.

## 4.2.6. Board Support Packages

El BSP és un dels components crítics necessaris per a desenvolupar el disseny de SO i així crear la imatge *runtime* de CE. Una documentació tècnica completa sobre els BSPs per a Windows Embedded CE podria ocupar un llibre sencer. En aquest apartat, es proporciona tan sols una vista general sobre els BSPs disponibles i com clonar i personalitzar a la nostra mida BSPs ja existents.

### Introducció als BSPs

Un BSP (*Board Support Package*) és un conjunt de components de software que combinen el codi d'adaptació de capes OEM necessari, *drivers* de dispositius, i arxius de configuració per a una plataforma específica de hardware.

El BSP (*Board Support Package*) és un nom comú de referència a una biblioteca del software necessària per a carregar i executar el sistema operatiu en un dispositiu hardware donat. Desenvolupar el BSP és un dels passos més importants durant l'etapa de desenvolupament. Si un BSP no està ja disponible, crear-ne un és generalment un dels primers passos a donar. El desenvolupament de BSPs inclou típicament el següents passos:

- Desenvolupar el *boot loader*.
- Desenvolupar el OEM *Adaptation Layer* (OAL).
- Desenvolupar els *drivers* de dispositius.
- Desenvolupar els arxius de configuració de la imatge executable de SO.

Windows Embedded CE 6.0 proporciona BSPs de suport a diverses famílies de CPUs. Aquests es troben a la ruta **WINCEROOT\PLATFORM\COMMON**.

Els BSPs disponibles a Windows CE són instal·lats a subcarpetes separades, amb el mateix nom que reben a la vista del catàleg d'ítems de Visual Studio 2005; des d'on els pots veure.

Els BSPs de referència proporcionats per Microsoft i instal·lats de manera predeterminada, són els següents:

- **BSP Aruba Board.** S'instal·la per a donar suport a processadors ARMV4I.
- **BSP CEPC.** S'instal·la per a donar suport a processadors x86.
- **BSP Device Emulator.** S'instal·la quan és seleccionat el suport a processadors ARMV4I.
- **BSP H4Sample OMAP2420.** S'instal·la quan és seleccionat el suport a processadors ARMV4I.
- **BSP MainstoneIII PXA27X.** S'instal·la també quan és seleccionat el suport a processadors ARMV4I.

No és estrictament necessari dominar els procediments de creació de BSPs, ja que normalment els fabricants de plataformes hardware proporcionen el BSP concret de cada dispositiu. D'altra banda, és possible que un fabricant no disposi del BSP d'un dispositiu concret per a Windows CE; i que no es pugui utilitzar un BSP genèric de Microsoft per a aquesta plataforma. En aquest cas, l'hauràs de dissenyar tu mateix.

Es recomanable, abans de res, poder el BSP del fabricant del hardware, en cas de que estigui disponible. Això garanteix una el correcte funcionament i compatibilitat entre els *drivers* del dispositiu i el hardware.

Pots consultar els BSPs disponibles per a Windows Embedded CE a les pàgina web: <http://www.microsoft.com/windowembedded/en-us/msdn/bspsearch.aspx> .

## Desenvolupament de BSPs

Desenvolupar un BSP no és una tasca fàcil. Implica l'ús de llenguatges de baix nivell, del tipus assemblador, i la creació de *drivers* per a la plataforma hardware. Això requereix tenir un coneixement profund de la plataforma hardware a la que va destinat; i pot esdevenir una tasca quasi inacabable.

Per a reduir el temps de desenvolupament, és habitual clonar un BSP ja existent; canviant només el codi font referent al nou hardware. Això és possible si, per exemple, ambdós comparteixen la mateixa arquitectura de CPU.

Es pot trobar molta informació sobre desenvolupament de BSPs per a Windows Embedded CE a la direcció web:

<http://channel9.msdn.com/wiki/cedevoloper/windowscebsp/>.

## Clonació d'un BSP

El procés de clonació d'un BSP ja existent té la funció de poder modificar la còpia feta sense que els canvis afectin al BSP original.

Fem un exercici d'exemple. Obre el Visual Studio 2005 i dins el menú desplegable *Herramientas* selecciona del submenú *Platform Builder for CE 6.0* la opció *Clone BSP*, per a obrir l'assistent *Clone Board Package Wizard* (figura 4.3).

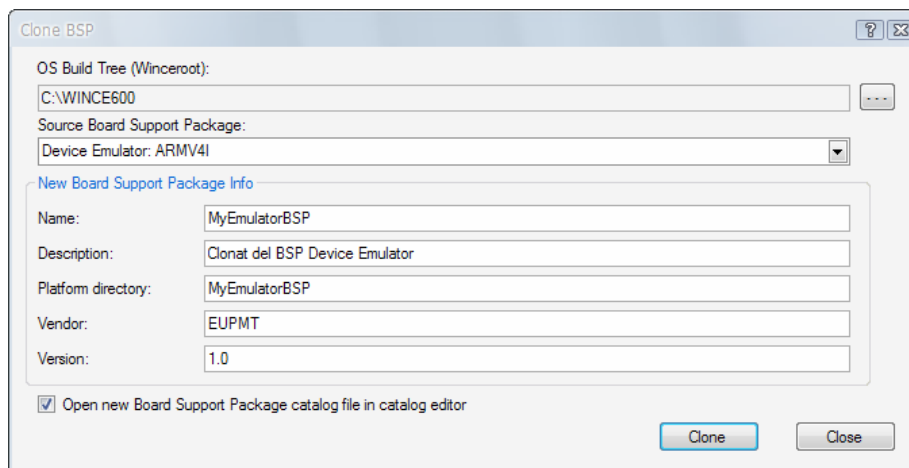


Figura 4.3. Finestra de Clone BSP.

A la casella *Source Board Support Package* selecciona el BSP que desitgis clonar. Per aquest exercici d'exemple, selecciona *CEPC: x86*. A la casella *Name* introdueix el nom del nou BSP, per exemple, **MyCEPCBSP** i crea un directori (casella *Platform directory*) amb el mateix nom.



Les caselles de *Vendor* i *Version* són informatives i no tenen cap impacte sobre el disseny. Pots emplenar-les o simplement deixar-les en blanc. Cliqueja finalment el botó *Clone*.

Els nous BSPs clonats els pots veure a la vista del catàleg d'ítems, a la carpeta *Third Party* (figura 4.4). En aquesta carpeta es dipositen tots els BSPs creats i els subministrats pels fabricants de hardware.

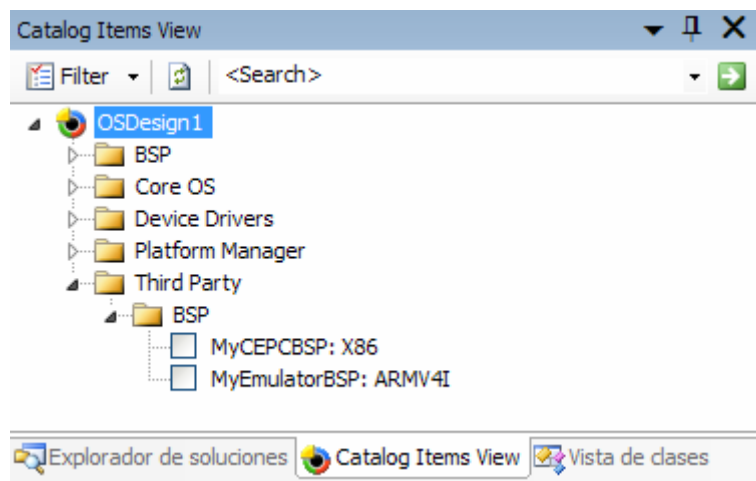


Figura 4.4. Finestra Catalog Items View.

Ara anem a editar el nou BSP clonat. Seleccionant la opció *Open new BSP catalog file in Catalog Editor* de l'assistent de clonació de BSPs es pot accedir a l'edició dels BSPs.

Un altre mètode per a editar el BSP que ja has clonat és accedint, a través del navegador, a la carpeta `PLATFORMROOT\Nom_del_BSP\CATALOG`. Entra a la carpeta del BSP desitjat, i obre'l amb VS2005.

En el cas d'aquest exercici d'exemple, la ruta del BSP clonat serà `PLATFORMROOT\MyCEPCBSP\CATALOG`. Dins d'aquesta carpeta selecciona l'arxiu `MyCEPC.pbxml` i cliqueja *Abrir con... Microsoft Visual Studio 2005*.

El BSP CEPC és un dels BSPs inclosos a Windows Embedded CE com a part de la instal·lació del *Platform Builder*. Dona suport a plataformes hardware basades en processadors x86.

Si tens un ordinador vell amb processador x86 de 32 MB o més de memòria de sistema, del tipus 486 o superior, és possible que aquest BSP et serveixi, com a suport software, per al teu vell PC. És a dir, pots crear una imatge de SO de Windows Embedded CE 6.0 i fer-la funcionar al teu ordinador obsolet.

## Eliminar components del BSP

Ara anem a esborrar components del BSP clonat. A la interfície del VS2005, expandeix la carpeta *Storage Drivers* de l'arxiu *MyCEPCBSP.pbcxml* obert.

Per al hardware típic basat en x86, és poc probable que necessitem els *drivers* de *PCI NAND Flash* i *Spansion Flash Memory*. Cliqueja amb el botó dret del ratolí cadascun dels dos *drivers* i selecciona *Remove*, per a eliminar-los.

## Agregar components al BSP

Ara anem a veure com s'afegeixen components a un BSP clonat. A l'editor de *MyCEPCBSP.pbcxml* del VS2005 posiciona el ratolí sobre *MyCEPCBSP: x86* i fes clic al botó dret del ratolí. Selecciona la opció *Add Catalog Item*.

A la finestra de Propietats del costat esquerre, introdueix de títol **ATAPI Storage Driver** i a la casella *Sysgen Variable* escriu **SYSGEN\_ATAPI**.

El *driver VGA Linear (Flat) Framebuffer* suporta la majoria de displays emprats a equips PC. Per afegir-lo a la nova BSP segueix els mateixos passos que en l'exemple anterior. Introdueix **VGA Linear (Flat)** a la casella *Title* i **BSP\_DISPLAY\_FLAT** a la casella de *Sysgen Variable*.

A la figura 4.5 es mostra l'aspecte del catàleg del *MyCEPCBSP.pbcxml* un cop eliminats i afegits els corresponents components.

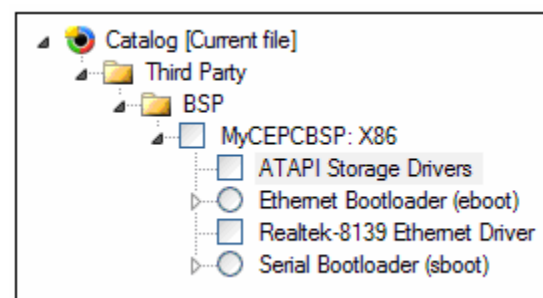


Figura 4.5. Carpeta BSP de Catalog Items View.

## Clonació de codi públic de BSPs

Un cop clonats els BSPs, és el moment de modificar-los. Encara que els components `SYSGEN_ATAPI` i `BSP_DISPLAY_FLAT` ja s'han afegit al `MyCEPCBSP`, hem de introduir les variables d'entorn dels *drivers* del dispositiu. El codi font i els arxius binaris per a aquests components es troben dins el directori `PUBLICROOT`.

Per a modificar el codi font dels nous components del `MyCEPCBSP` és recomanable clonar el codi font del BSP original. Ara anem a fer un exercici d'exemple de clonació del codi font dels *drivers VGA Linear (Flat) Framebuffer* al BSP `MyCEPCBSP`, i nombrar la versió clonada *MyVGA display driver*. Els passos a seguir es descriuen a continuació:

- [1] Crear una carpeta per a *MyVGA display driver* al directori:  
    `_PLATFORMROOT\MyCEPCBSP\SRC\Drivers\`.
- [2] Copiar tots els arxius de `_PUBLICROOT\Common\Oak\Drivers\Display\VGAFlat` a la nova carpeta creada.
- [3] Renombrar l'arxiu `ddi_flat.def` amb el nom **MyVGA.def**.

I per últim, s'ha de modificar el contingut del codi font del nou BSP clonat. Per a fer-ho s'ha d'obrir amb un editor de text (com per exemple, el bloc de notes) l'arxiu *Source* de la carpeta `_PLATFORMROOT\MyCEPCBSP\SRC\Drivers\` i fer els canvis desitjats.

Com ja s'ha comentat al principi de l'apartat, explicar de forma detallada el disseny i edició de BSPs podria ocupar un llibre sencer. Si desitges obtenir més informació sobre aquest tema, pots consultar la documentació de MSDN Visual Studio 2005 i la direcció web: <http://channel9.msdn.com/wiki/cedevloper/windowscebsp/>.

També es recomanable consultar el llibre "Microsoft Embedded CE 6.0" de edicions Wrox, escrit per Samuel Phung; d'on s'ha estret gran part de la informació continguda en aquest apartat.

### 4.2.7. Creació d'una imatge executable personalitzada

En aquest apartat es mostren els passos a donar per a crear un disseny de sistema operatiu, incloent components addicionals del catàleg de components, configurar variables d'ambient, personalitzar el disseny de SO, i generar una imatge executable.

Per a crear una imatge executable CE 6.0, hem de seguir els següents passos:

- [1] Crear el projecte de disseny del sistema operatiu.
- [2] Seleccionar els BSPs (*Board Support Packages*).
- [3] Seleccionar els components addicionals necessaris del catàleg de components de CE 6.0.
- [4] Generar la imatge executable del SO del projecte de disseny del SO.

### Creació d'un disseny de SO inicial

Començarem creant un nou projecte. Fes clic al menú *Archivo*, després a *Nuevo*, i per últim a *Proyecto*. Apareixerà la següent finestra (figura 4.6):

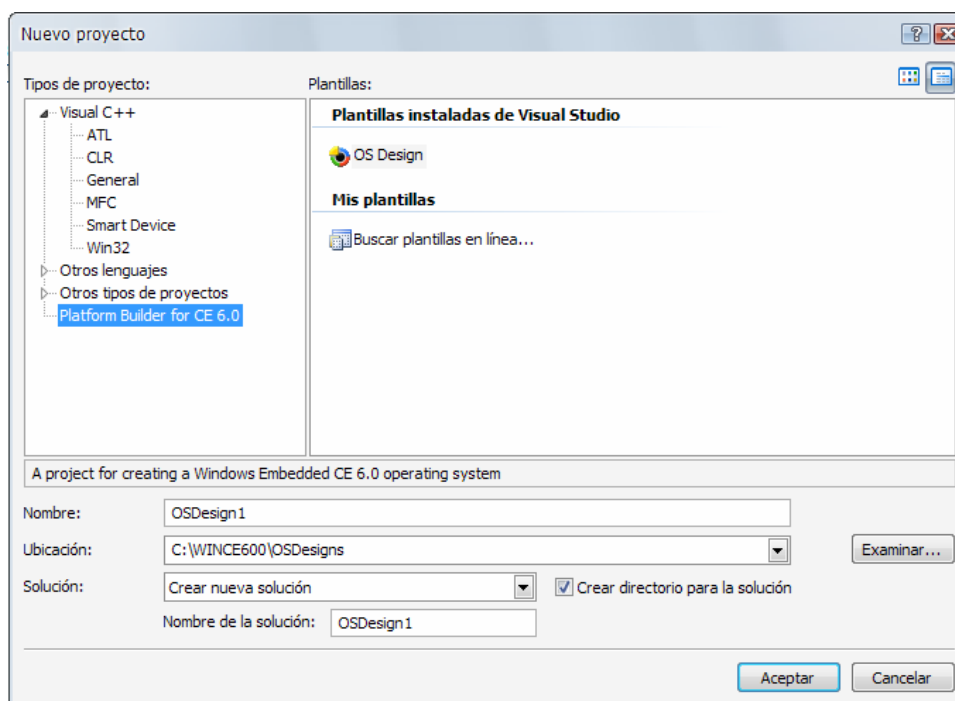


Figura 4.6. Finestra inicial de l'assistent d'un nou projecte.

Selecciona la opció *Platform Builder for CE 6.0*. Introdueix a *Nombre* el nom del projecte. Per defecte, apareix el nom *OSDesign1*. Deixa d'entrada aquest nom i cliqueja *Aceptar*.



Figura 4.7. Pantalla inicial de l'assistent OS Design Wizard.

En aquest punt, l'assistent *OS Design Wizard* es mostra a la pantalla (figura 4.6). Aquest segueix una sèrie de passos, presentant opcions diferents per a seleccionar els BSPs, la plantilla de disseny del SO, components multimèdia, i components d'aplicació; permetent també crear l'entorn inicial de desenvolupament per al projecte.

Per a avançar al pròxim pas, fes clic a *Siguiete*. La següent pantalla és l'assistent de selecció del BSPs (*Board Support Packages*), tal com es pot observar a la figura 4.8. Aquesta pantalla mostra un llistat de tots els BSPs actualment instal·lats a la teva estació de desenvolupament.

Per a un mateix projecte, poden ser seleccionats múltiples BSPs; tot i que cada un està dissenyat per a donar suport a un únic hardware. Seleccionant múltiples BSPs, el disseny de SO pot configurar-se per a generar múltiples imatges *runtime* de SO, una per a cada BSP (donant suport així al hardware associat a cada BSP).

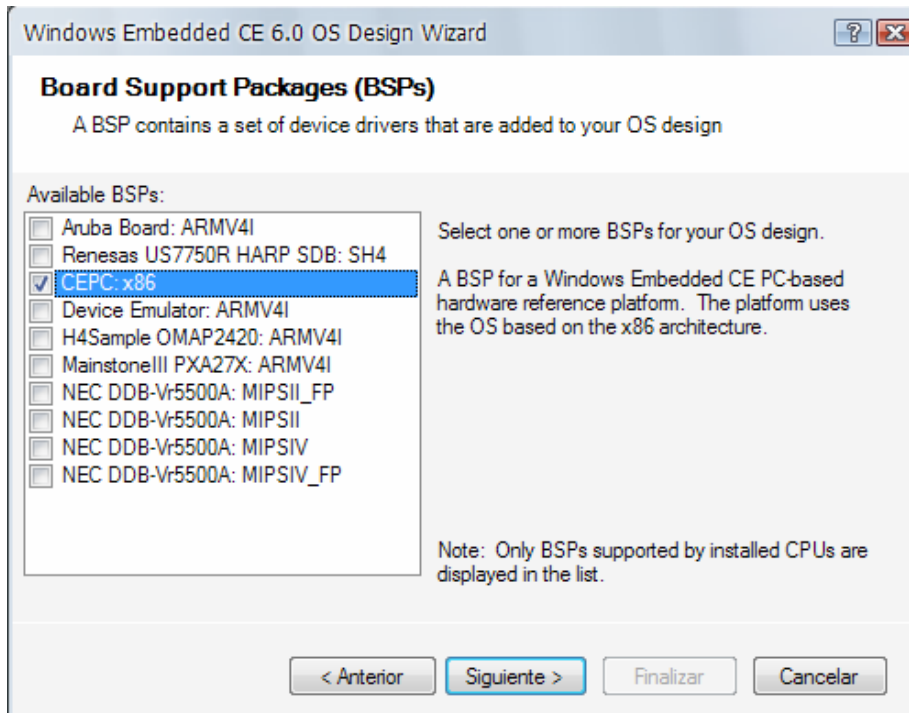


Figura 4.8. Selecció de BSPs.

Selecciona *CEPC: x86* i *Device Emulator: ARMV4I* per a poder implementar el SO que crearem. Cliqueja ara *Siguiente* per accedir a la pantalla *Design Templates* (figura 4.9).

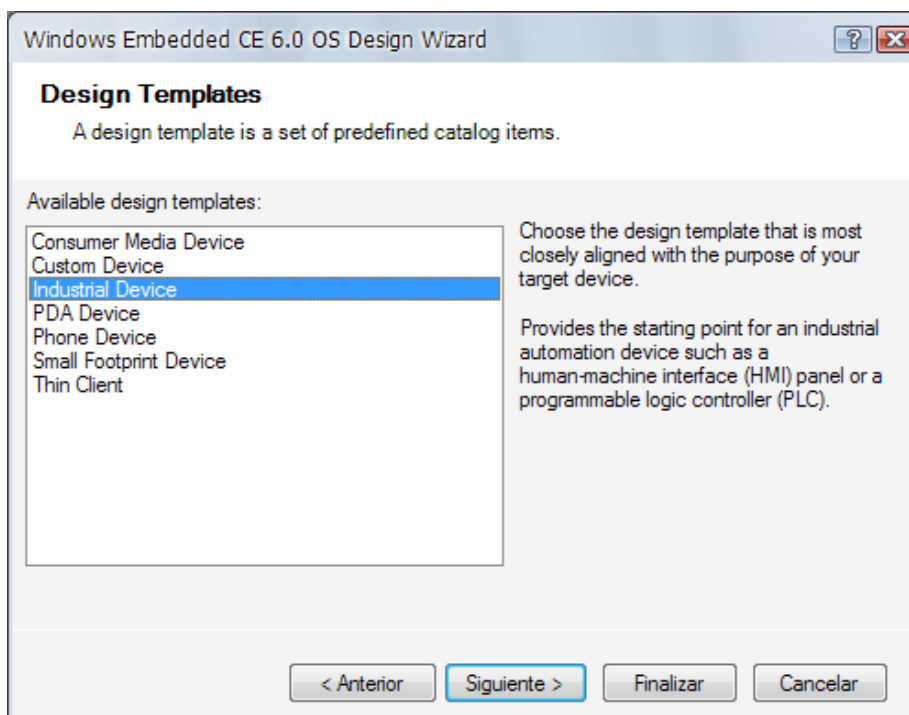


Figura 4.9. Selecció de la plantilla de disseny.

La pantalla *Design Templates* presenta diferents categories de plantilles de disseny. Fes clic a sobre de cada plantilla i a la part esquerra apareixerà una breu descripció de la mateixa. Per a la realització d'aquest exercici d'exemple selecciona la opció *Industrial Device*, i després fes clic a *Siguiente*.

Per a cada plantilla de disseny de SO, l'assistent *OS Design Wizard* proporciona una o varies variables de plantilla per a escollir (pantalla *Design Template Variants*, figura 4.10). Per a aquest exercici d'exemple, escull *Internet Appliance* i fes clic a *Siguiente*.

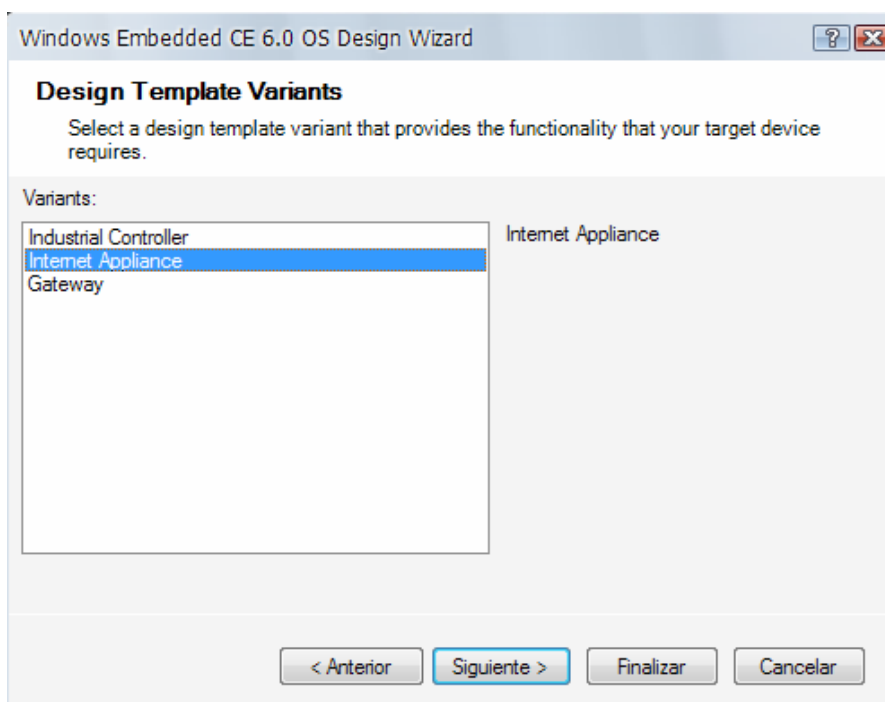


Figura 4.10. Selecció de les variants de plantilles.

La següent pantalla (figura 4.11) de l'assistent *OS Design Wizard* correspon a *Applications & Media*. Proporciona la opció d'incloure aplicacions, components multimèdia, i llibreries programables; que poden proporcionar al disseny una funcionalitat addicional.

El component de llibreries *NET. Compact Framework 2.0* dona suport al codi font d'aplicacions. *Internet Explorer 6.0* dona suport a la funció de navegador d'Internet. I els components *Media* de Windows proporcionen funcionalitats multimèdia bàsiques.

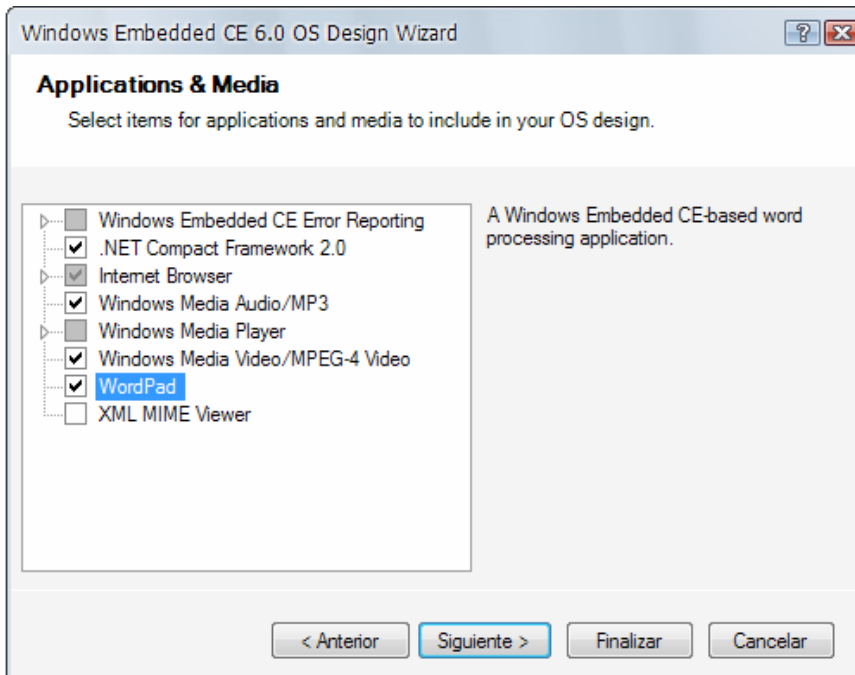


Figura 4.11. Selecció d'aplicacions multimèdia..

Seguim avançant al proper pas per a seleccionar les opcions addicionals proporcionades per l'assistent. Fes clic a *Sigüiente* per a passar a la selecció de components de comunicacions *Networking*. Expandeix els menús *Local Area Network (LAN)*, *Personal Area Network (PAN)*, i *Wide Area Network (WAN)*, tal com es mostra a la figura 4.12.

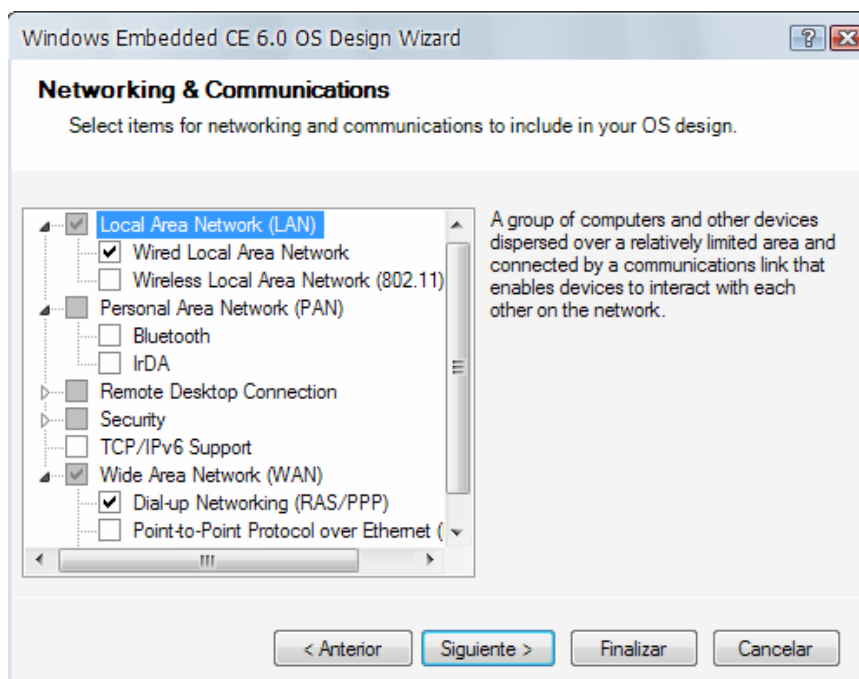


Figura 4.12. Selecció d'ítems per a les comunicacions.



Les funcionalitats *Networking* són de gran utilitat per a dispositius *embedded*, donat que proporcionen les següents funcions:

- **Components LAN (*Local Area Network*)**. Proporcionen les llibreries i utilitats necessàries per al suport a LAN, com Ethernet i RAS (*Remote Access Service*).
- **Components PAN (*Personal Area Network*)**. Proporcionen les utilitats i llibreries de suport necessàries per a PAN, com IrDA i Bluetooth.
- **Components *Remote Desktop Connection***. Proporcionen les utilitats i llibreries de suport necessàries per a establir sessions remotes *desktop*.
- **Components de Seguretat**. Proporcionen les utilitats i llibreries de suport necessàries per a l'ús de funcions de seguretat i encriptació.
- **Components TCP/IPv6**. Proporcionen les utilitats i llibreries de suport necessàries per al nou *stack TCP/IPv6*.
- **Components WAN (*Wide Area Network*)**. Proporcionen les utilitats i llibreries de suport per a comunicacions *Dial-up*, *Point-to-Point* i *Virtual Private*.

Els components addicionals de comunicació disponibles al catàleg de components poden ser afegits posteriorment, un cop creat el disseny de SO.

*Cliquejant la opció Siguiete*, s'arriba a la última pantalla de *OS Design Wizard*, que dona per finalitzat l'assistent.

L'assistent *OS Design Wizard*, un cop completats tots els passos, genera una carpeta de projecte amb subcarpetes i arxius del disseny de SO. Al primer pas de l'assistent, hem deixat el nom que dona el programa per defecte OSDesign1. VS2005 crea una carpeta amb el mateix nom a **WINCEROOT\OSDesigns\OSDesign1** i crea diverses subcarpetes i arxius. A continuació es descriuen les subcarpetes i arxius utilitzats.

- **Subcarpeta \OSDesign1\OSDesign1.** Conté l'arxiu del projecte de disseny del SO.
- **Arxiu \OSDesign1\OSDesign1.sln.** Arxiu de solucions de VS2005.
- **Arxiu \OSDesign1\OSDesign1.pbxml.** Arxiu de projecte de la plataforma (*Platform Builder*).
- **Directori \OSDesign1\OSDesign1\RelDir.** Conté els BSPs seleccionats pel projecte.

Nota: Al llarg de tot aquest apartat, dedicat a la creació de plataformes mitjançant Windows Embedded CE i les eines de desenvolupament de Visual Studio 2005, anomenarem **WINCEROOT** al directori arrel de Windows CE 6.0, tal com ja s'ha especificat anteriorment.

En aquest punt, l'explorador de solucions del *Workspace* t'indicarà quin BSP està actiu al teu projecte. Pots generar una imatge *runtime* per a cada un dels BSPs seleccionats, però has de seleccionar cada cop una d'ells com l'actiu, per a generar la imatge.

Per a canviar el BSP actiu, selecciona al menú *Generar* la opció *Administrador de configuració*. Un cop dins de la finestra de l'administrador de configuració (figura 4.13) selecciona el BSP a la pestanya *Configuración de soluciones activas*. Aquesta conté les següents configuracions:

- **CEPC x86 Debug.** És la configuració per a generar la depuració de la imatge executable, amb suport per a dispositius *target* basats en processadors x86.
- **CEPC x86 Release.** És la configuració per a generar el llançament de la imatge executable, amb suport per a dispositius *target* basats en processadors x86.

- **Device Emulator ARMV4I Debug.** És la configuració per a generar la depuració de la imatge executable, amb suport per a dispositius *target* basats en processadors ARM.
- **Device Emulator ARMV4I Release.** És la configuració per a generar el llançament de la imatge executable, amb suport per a dispositius *target* basats en processadors ARM.

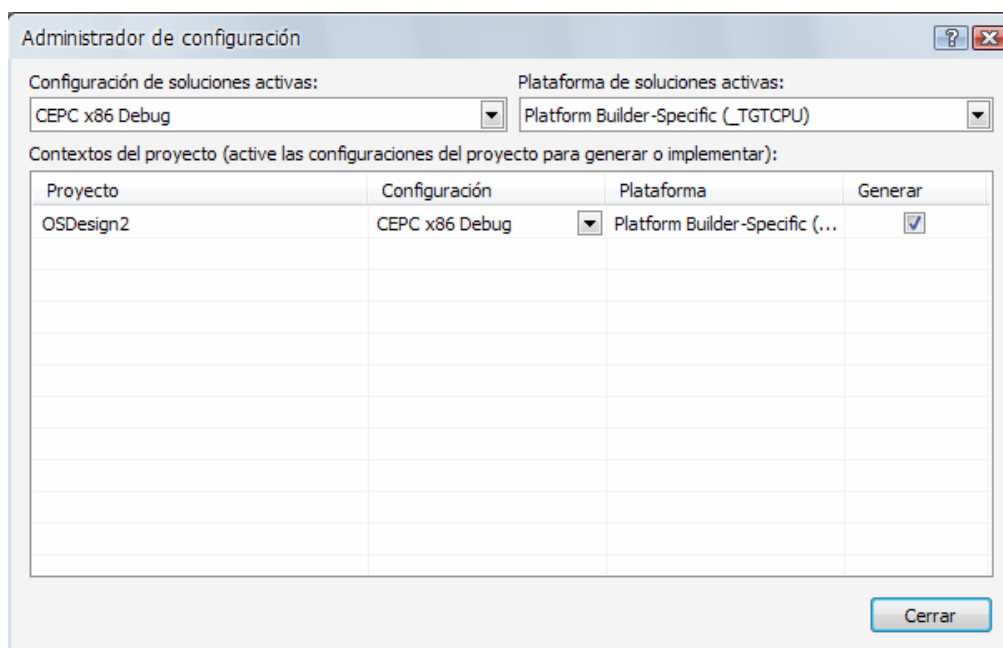


Figura 4.13. Administrador de configuracions.

Ara anem a seleccionar una aplicació per al nostre projecte a la vista del catàleg d'articles. A la casella *Search* (recerca), escriu **Remote Display** i pitja l'Enter. Apareixerà dins de la carpeta *Debugging tools* l'article *Remote Display Application*.

Selecciona'l per a incloure'l al disseny. Aquesta aplicació proporciona una funció similar a l'escriptori remot de Windows XP, habilitant accés remot a l'escriptori de Windows Embedded CE executant l'aplicació client *cerhost.exe*.

Aquesta aplicació es troba a la ruta: **C:\WINCE600\PUBLIC\COMMON\OAK\BIN\i386\cerhost.exe**.

Amb aquesta aplicació, el dispositiu *target* pot ser accessible utilitzant la pantalla, el teclat i el display de l'estació de desenvolupament (el sistema *host*).

A la vista de l'explorador de solucions del *Workspace* pots observar l'estructura de carpetes de Windows Embedded CE, els arxius de configuració, els subprojectes de disseny de SO, i el Kit de desenvolupament de software SDK (*Software Development Kit*). També permet veure la carpeta de favorits, on pots afegir enllaços de les parts més freqüentment utilitzades de del codi font de Windows Embedded CE, tal com ens mostra la figura 4.14. A l'explorador de solucions pots visualitzar components, subprojectes, obrir finestres de creació, etc.

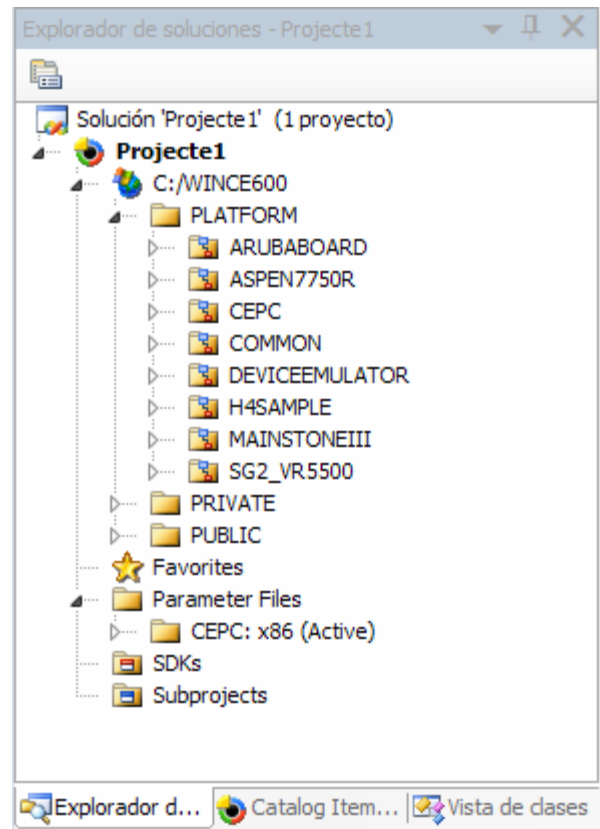


Figura 4.14. Estructura d'arxius d'un projecte de Visual Studio 2005.

La vista d'ítems del catàleg (*Catalog Items View*) del *Workspace* et permet afegir opcions remotes, mòduls, i components del disseny de sistemes operatius.

La pestanya *Filter* et permet veure tots els articles del catàleg (*All Catalog Items in Catalog*), els articles del catàleg seleccionats per l'usuari (*User – Selected Catalog Items*) i les dependències i els articles seleccionats per l'usuari (*User-Catalog Items and Dependencies*).

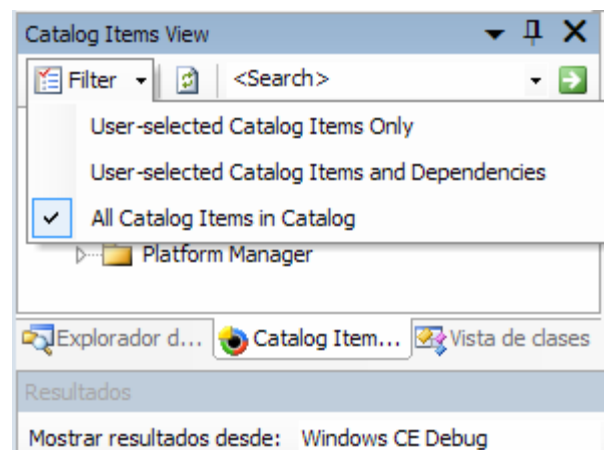


Figura 4.15. Filtre d'opcions

Les caselles buides dels articles de la vista d'ítems del catàleg són opcions no seleccionades. Cliquejant sobre aquestes caselles, se seleccionen les opcions. Si observes els articles, veuràs que hi ha opcions habilitades automàticament pel sistema per a resoldre dependències necessàries.

La vista de classes, ofereix un pràctic explorador per a treballar amb el codi font de subprojectes. L'examinador de crides et permet determinar ràpidament quines tasques fan una crida a una funció particular, gràcies a la seva capacitat de recerca. La finestra de definició de codi mostra la definició del codi de funció seleccionat a l'editor.

Pots afegir finestres addicionals al teu Platform Builder de Visual Studio 2005, seleccionant la opció desitjada al menú desplegable *Ver*, tal com es pot observar a la figura 4.16.

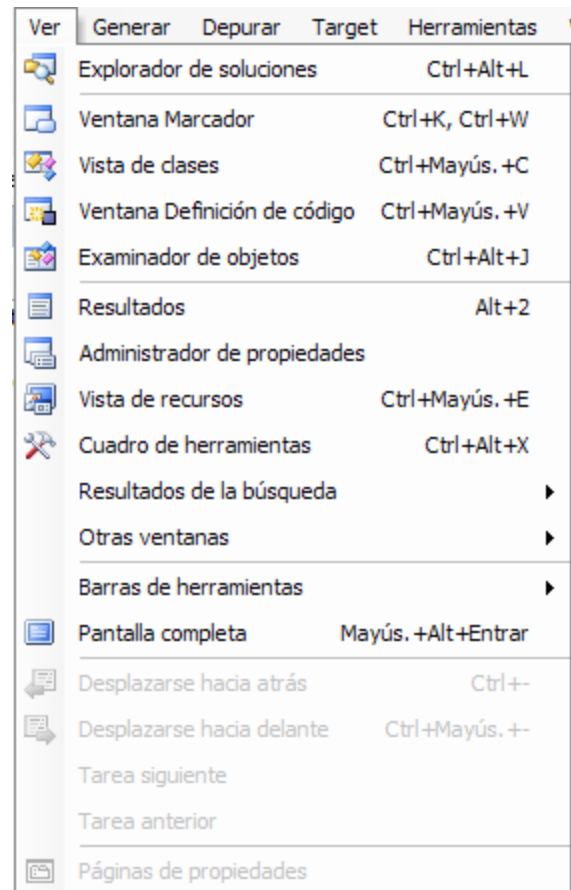


Figura 4.16. Menú *Ver*.

Ara veiem les opcions disponibles al menú principal. El menú *Proyecto*, tal com es mostra a la figura 4.17, et permet afegir subprojectes nous o ja existents al disseny del SO. A més pots afegir SDKs (Software Development Kits) nous o ja existents, i accedir a les propietats dels objectes seleccionats a l'explorador de solucions.

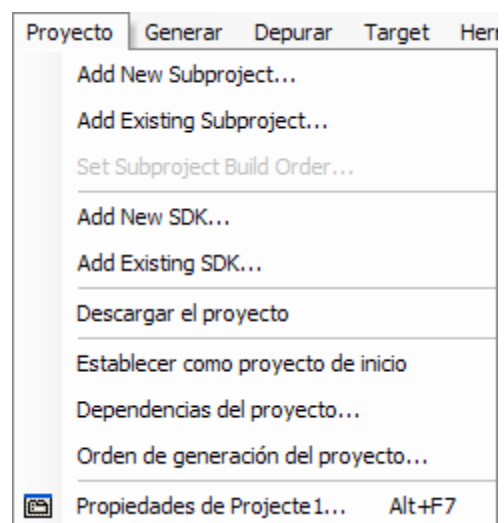


Figura 4.17. Menú *Proyecto*.

## Addició d'ítems del catàleg al disseny

Windows Embedded CE és un sistema operatiu componentitzat. L'eina *Platform Builder* inclou articles i dependències seleccionables per l'usuari, exclouent els ítems innecessaris. Si un ítem necessari no és inclòs manualment en el disseny del sistema operatiu i no és disparat per dependències incloses, no estarà a la imatge final del sistema.

Al catàleg d'ítems hi ha una casella buida on es pot escriure aquell ítem que es desitja trobar dins el *workspace*. Això és una eina útil que evita haver d'obrir totes les carpetes del catàleg per a trobar un ítem ja conegut.

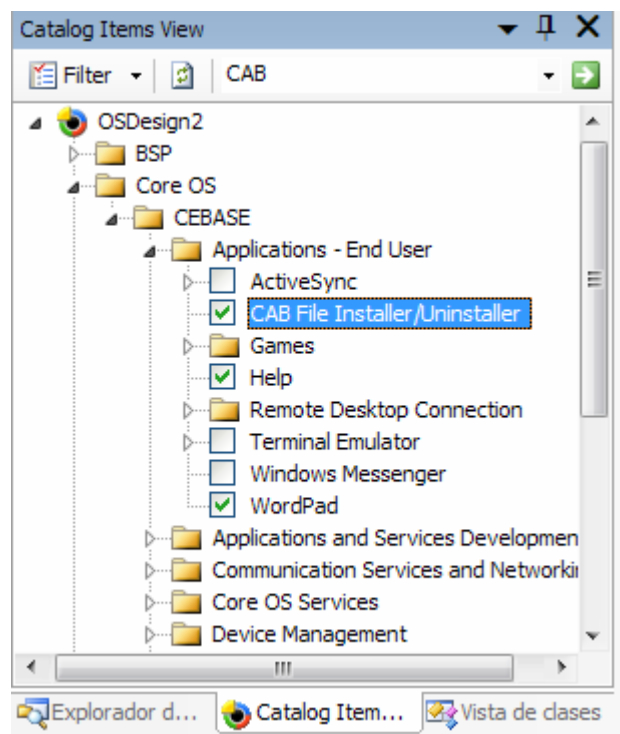


Figura 4.18. Ítems del catàleg.

Utilitza l'eina de recerca per a trobar i agregar els articles següents al disseny de SO:

- **CAB File Installer/Uninstaller.** Aquest article és necessari per a donar suport a la instal·lació d'aplicacions i desinstal·lar (veure figura 4.18).
- **USB Storage Class Driver.** És un driver requerit per a suportar dispositius d'emmagatzematge massiu USB.
- **Remote Display Application.** Permet accés remot a l'escriptori de Windows CE mentre s'executa l'aplicació cerhost.exe.

En aquest punt, tots els ítems del catàleg necessaris són inclosos al disseny d'exemple.

Si al *Explorador de Soluciones* està seleccionada l'arrel del disseny del SO seleccionat, fent clic a la opció *Propiedades* del menú *Proyecto* s'obrirà una finestra amb les propietats del disseny del SO, tal com mostra la figura 4.19.

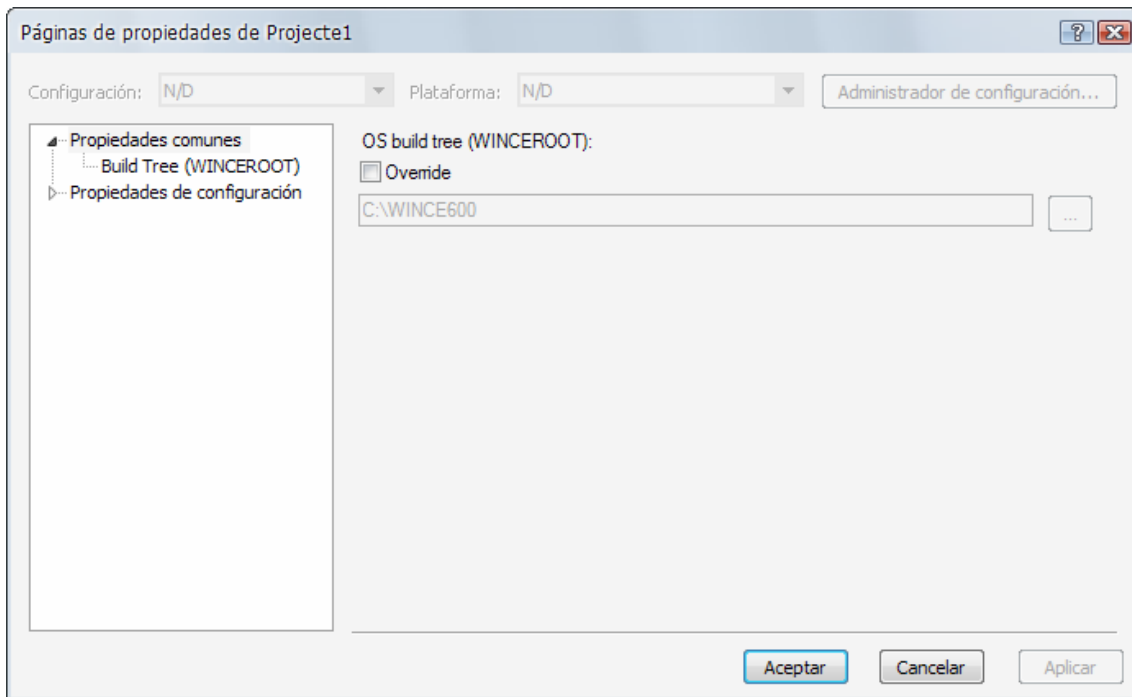


Figura 4.19. Propietats del disseny del projecte.

Les propietats comunes són aquelles que s'apliquen a tot l'entorn de disseny. Tenen una única configuració, que és l'especificació de l'arbre de construcció del SO on Windows Embedded CE 6.0 està instal·lat. Seleccionant *Propiedades de configuración*, apareix una llista on pots escollir el tipus de configuració per a veure i editar les propietats: *Active*, *Debug*, *Release*, o *Todas las configuraciones*.

Ara anem a veure les *Propiedades de configuración*. Dins del subapartat *General*, pots escollir el directori arrel en el qual quedaran guardats tots els arxius que es generin amb el projecte. A partir d'ara, anomenarem aquest directori arrel com a WINCEROOT (tal com ho fa el programa).

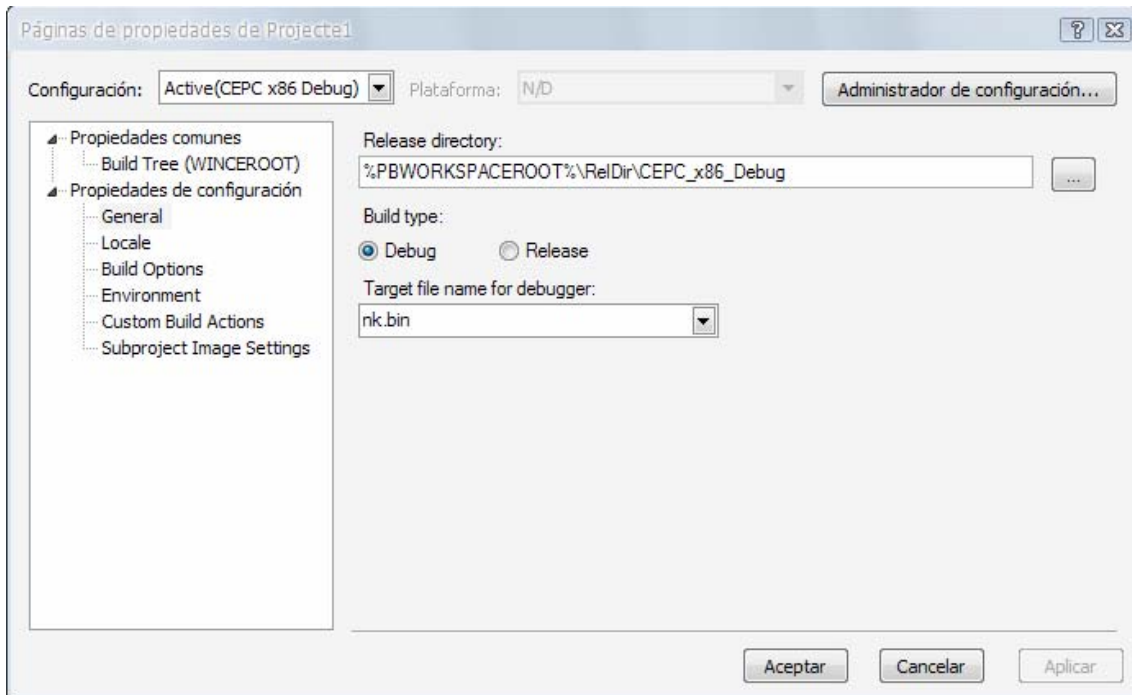


Figura 4.20. Propietats de configuració del disseny de SO.

La configuració dels paràmetres de l'apartat *Local*, et permet especificar aspectes com els idiomes suportats, les pàgines de codi, etc. Observa la figura 4.21.

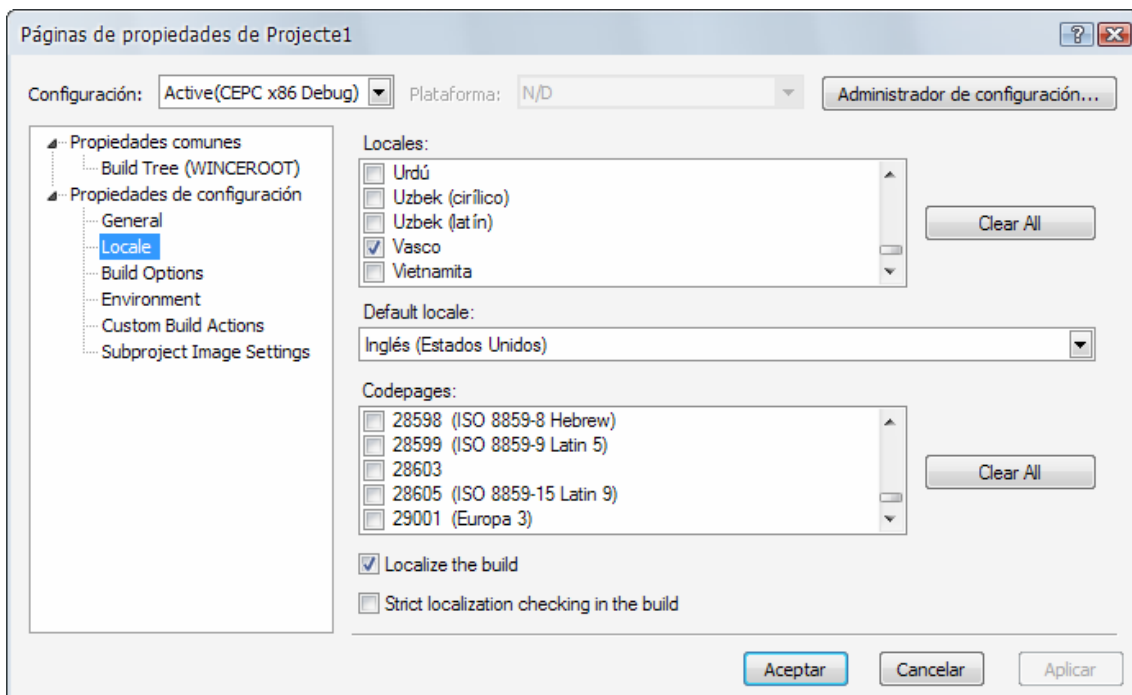


Figura 4.21. Opcions locals del disseny de SO.



Cliquejant sobre *Build Options*, apareix una finestra que et permet configurar les variables utilitzades més freqüentment per a controlar el procés de creació, tal com ens mostra la figura 4.22.

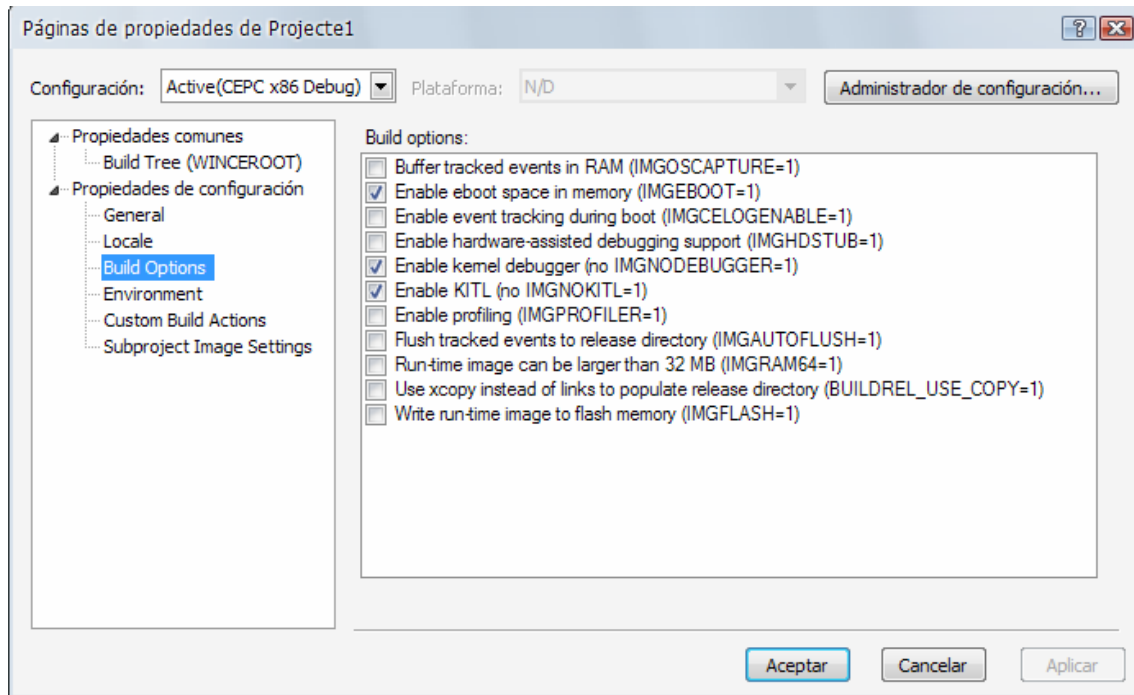


Figura 4.22. Opcions constructives del disseny de SO.

A continuació es descriuen les diferents opcions constructives seleccionables al menú *Propiedades de configuración*.

- **Build tracked events in RAM.** Variable: IMGOSCATURE. Afegeix el mòdul OSCapture.exe a la imatge. Durant la càrrega, el mòdul de SO comença a escriure esdeveniments de sistema dins la RAM.
- **Enable eboot space in memory.** Variable: IMGEBOOT. Reserva espai *bootable* a la memòria. Habilita al mòdul per a preservar les dades que poden ser llegides pel sistema durant la càrrega.
- **Enable event tracking during boot.** Variable: IMGCELOGENABLE. Afegeix l'arxiu CELog.dll a la imatge i inicialitza la col·lecció d'esdeveniments del sistema quan és carregat.

- **Enable hardware - assisted debugging support.** Variable: IMGHDSTUB. Habilita el suport a la depuració del hardware.
- **Enable kernel debugger.** Variable: IMGNODEBUGGER. Inclou suport a la depuració del *kernel*.
- **Enable KITL.** IMGNOKITL. Inclou suport a KITL (*Kernel Independent Transport Layer*).
- **Enable profiling.** Variable: IMGPROFILER. Inclou suport al perfil del *kernel*.
- **Enable ship build.** Variable: WINCESHIP. Quan aquesta opció és seleccionada, la imatge resultant de sistema operatiu no dona sortida a missatges de depuració. Quan aquesta opció no és seleccionada, el sistema operatiu proporciona detallats missatges per ajudar a la depuració. Aquesta opció està només disponible per a la construcció del llançament i està oculta a la configuració de la depuració.
- **Flush tracked events to release directory.** Variable: IMGAUTOFLUSH. Permet netejar l'apuntador d'esdeveniments del directori de llançament.
- **Runtime image can be larger than 32 MB.** Variable: IMGRAM64. Permet el suport per a imatges executables més llargues de 32 MB.
- **Use xcopy instead of links to populate release directory.** Variable: BUILDREL\_USE\_COPY. Copia arxius al directori de llançament en comptes de crear enllaços durs.
- **Write runtime image to flash memory.** Variable: IMGFLASH. Permet escriure de la imatge executable a la memòria flash després de la descàrrega.

Les opcions d'entorn del submenú *Environment*, figura 4.23, et permeten modificar els paràmetres constructius especificant variables d'entorn addicionals.

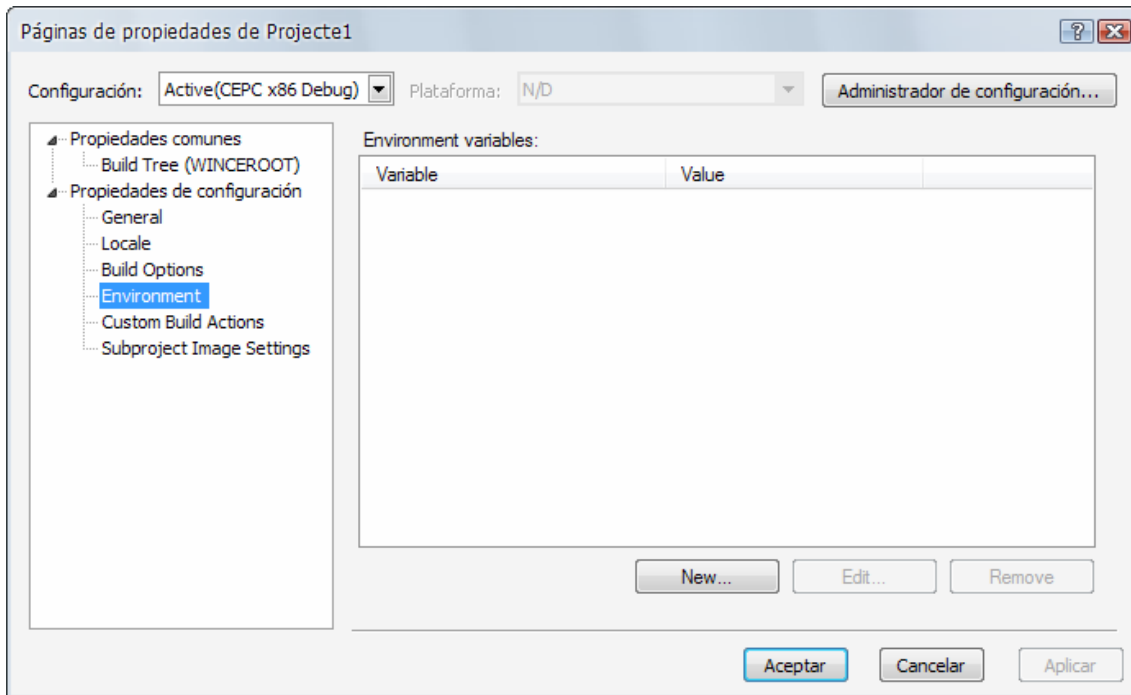


Figura 4.23. Especificació de variables d'entorn addicionals.

El paràmetre *Custom Build Actions* et permet realitzar accions constructives personalitzades durant certs instants de la creació, tal com es pot observar a la figura 4.24.

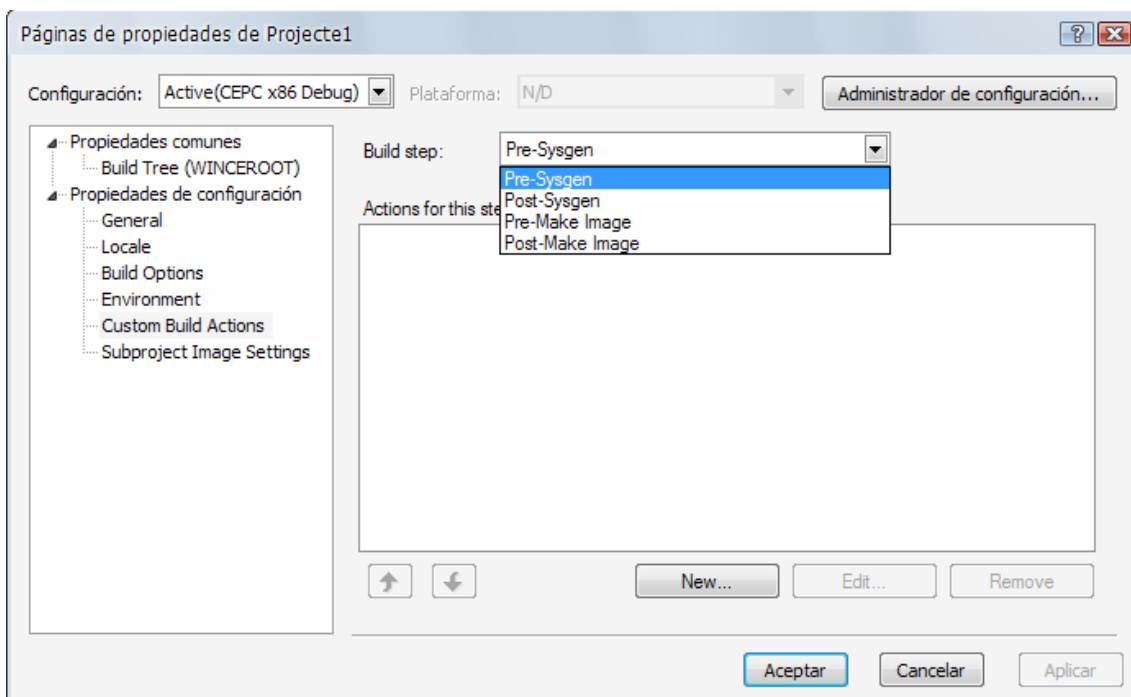


Figura 4.24. Accions constructives personalitzades.

Per a l'exercici d'exemple actual, selecciona les opcions:

- **Enable eboot space in memory.** Selecciona aquesta opció per a reservar espai de memòria per a l'arrencada del sistema, on es guardaran dades de paràmetres, com la resolució de pantalla o l'idioma del teclat; i permetre al SO llegir-les durant l'arrencada.
- **Enable KITL.** Selecciona aquesta opció per habilitar el KITL (*Kernel Independent Transport Layer*), l'enllaç de comunicació entre l'estació de desenvolupament i el dispositiu *target*, per a la depuració.
- **Run-time image can be larger than 32 MB.** Quan aquesta opció està activada, s'habilita la variable d'entorn IMGRAM64, que configura la imatge *runtime* per a utilitzar un sistema de memòria de 64 MB.

## Variables d'entorn

A més de les opcions constructives, es poden afegir variables addicionals d'entorn al disseny, des del menú de propietats de disseny del SO. Per exemple, si volguéssim excloure tots els components d'àudio de la imatge, per mitjà de la variable d'entorn BSPNOAUDIO, necessitem fer el següent:

[1] Selecciona al menú desplegable *Proyecto* la opció *Propiedades de OSDesign1*.

[2] A la part esquerra, expandeix les *Propiedades de Configuración* i cliqueja sobre *Environment*, per a desplegar les propietats d'entorn. Fes clic a *New* i escriu el nom de la variable i el seu valor associat, tal com indica la figura 4.25.

Nota: Per a habilitar el funcionament d'una variable se li ha d'adjudicar el valor 1.  
Exemple: BSP\_NOAUDIO = 1.

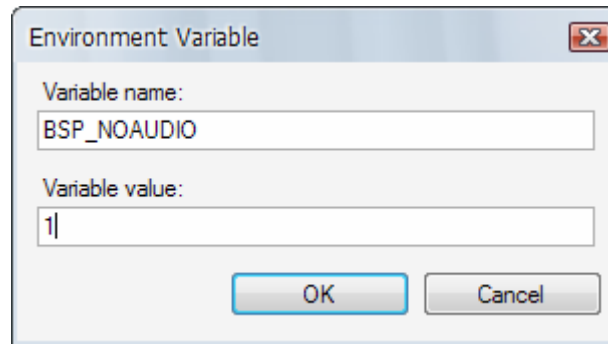


Figura 4.25. Variable d'entorn.

Les variables d'entorn juguen una paper important en com el disseny construeix, habilita i deshabilita certes característiques fonamentals i configura la forma en que la imatge executable del SO és creada. Per a més informació sobre les diferents variables d'entorn i les seves funcions associades, consulta l'apartat *BSP environment variables* de la documentació de Windows Embedded CE 6.0.

## Generació de la imatge executable de SO

El *Platform Builder* construeix, compila, i crea una imatge executable del SO. El procés de construcció d'una imatge és llarg. Depenent de l'estació de desenvolupament, aquest procés pot durar de 15 a 30 minuts, o inclús més.

Selecciona del menú desplegable *Generar* la opció *Generar solució*. Durant el procés de construcció de la imatge, la finestra de sortida de VS2005 mostra els mòduls i llibreries que s'estan construint en aquest moment (figura 4.26).

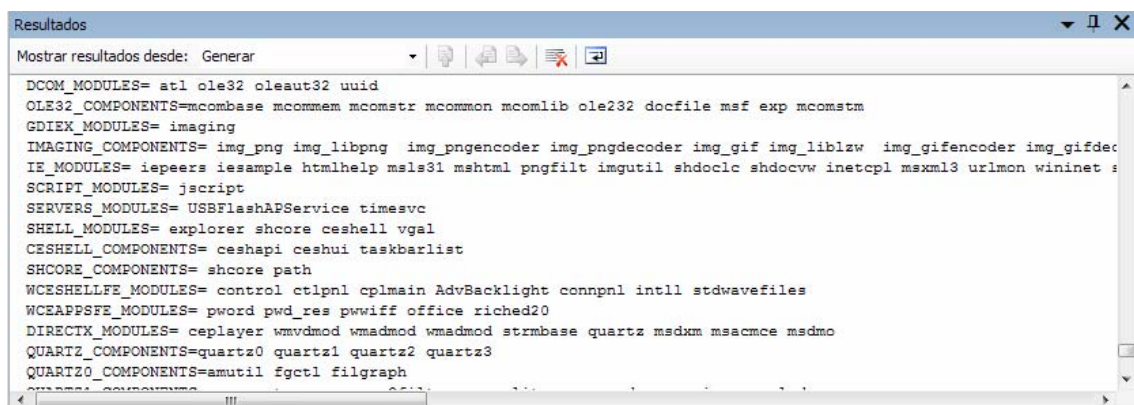
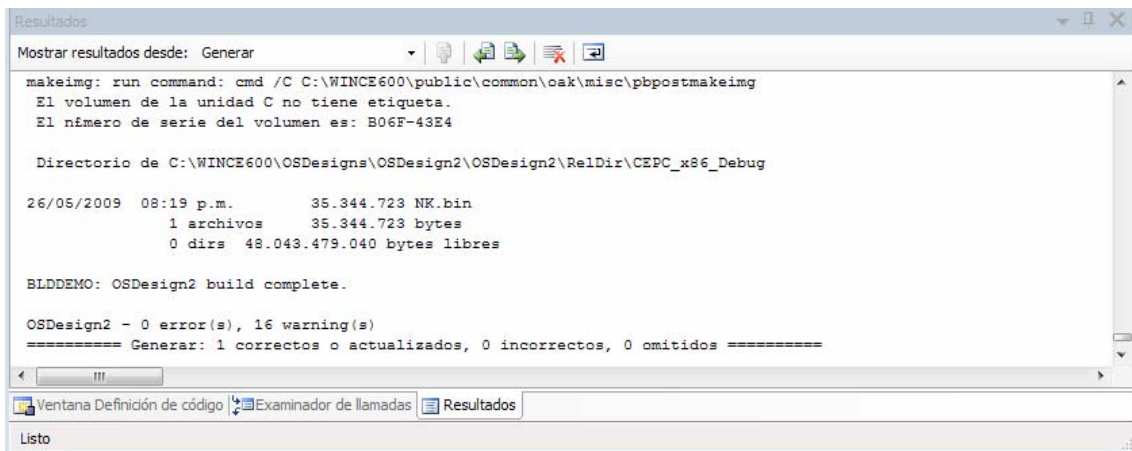


Figura 4.26. Finestra de sortida de Visual Studio 2005.

Finalitzada la creació, la finestra de sortida de VS2005 mostra els resultats de la mateixa, tal com mostra la figura 4.27.



```

Resultados
Mostrar resultados desde: Generar
makeimg: run command: cmd /C C:\WINCE600\public\common\oak\misc\pbpostmakeimg
El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: B06F-43E4

Directorio de C:\WINCE600\OSDesigns\OSDesign2\OSDesign2\RelDir\CEPC_x86_Debug

26/05/2009 08:19 p.m.      35.344.723 NK.bin
      1 archivos      35.344.723 bytes
      0 dirs  48.043.479.040 bytes libres

BLDDemo: OSDesign2 build complete.

OSDesign2 - 0 error(s), 16 warning(s)
==== Generar: 1 correctos o actualizados, 0 incorrectos, 0 omitidos =====

```

Figura 4.27. Resultats de sortida.

Quan el procés de creació s'ha completat amb un o varis errors, la creació entra en fallida i no es genera la imatge del sistema operatiu. Per a poder crear una imatge executable de sistema operatiu, el procés de creació no ha de tenir cap error. Tal com es pot observar a la figura 4.28, el procés de creació de la imatge té 16 missatges d'advertència i cap missatge d'error, completant-se així amb èxit.

Els arxius generats per la creació de la imatge (d'aquest exercici d'exemple) són emmagatzemats a la carpeta: **C:\WINCE600\OSDesigns\OSDesign1\OSDesign1**.

La imatge executable del SO (d'aquest exercici d'exemple) queda guardada a la carpeta: **OSDesign1\RelDir\DeviceEmulator\_ARMV4I\_Release**.

Dins d'aquesta carpeta es genera la imatge executable de SO, *nk.bin*, juntament amb més de 1.000 objectes.

A més de mostrar informació i advertiments durant el procés de creació, la informació de la finestra de sortida queda enregistrada als arxius següents:

- **WINCEROOT\Build.log** . És un arxiu molt llarg, que registra totes les activitats realitzades durant la creació.

- **WINCEROOT\Build.wrn.** Aquest arxiu anota tots els missatges d'avertència (*warning messages*), que també són inclosos a l'arxiu Build.log.
- **WINCEROOT\Build.err** Aquest arxiu registra tots els missatges d'error (*error messages*). Quan la creació té èxit, aquest arxiu no arriba a crear-se.

Ara s'ha de realitzar els mateixos passos per a generar la imatge de *CEPC x86 Release*. Selecciona la opció *Propiedades de OSDesign1* del menú *Proyecto*. Seguidament escull la opció *CEPC x86 Release* i cliqueja *Close*.

Ara veure'm un exemple de com agregar *drivers* a la imatge executable de SO generada per aquest BSP. Afegirem els *drivers* de les targetes Ethernet següents:

- NE2000 - compatible ISA card
- NE2000 - compatible PCI card
- NE2000 - compatible PCMCIA card
- Realtek RTL8139

Cliqueja sobre *Environment* per accedir a la configuració de les variables d'entorn i realitzar les següents accions:

- [1] Cliqueja sobre *New* per agregar una nova entrada a la pantalla *Environment Variables*.
- [2] A la casella de nom de la variable, introdueix *BSP\_NIC\_RTL8139*.
- [3] A la casella de valor de la variable, introdueix 1.
- [4] Cliqueja *OK* per a completar el procés.
- [5] Repeteix els anteriors passos per a les variables d'entorn *BSP\_NIC\_NE2000\_PCI* i *BSP\_NIC\_NE2000\_ISA*.
- [6] Cliqueja *Aplicar* seguit de *Aceptar* per a continuar.

La finestra de variables d'entorn de la pàgina de propietats del disseny mostrarà finalment l'aspecte que ens mostra la figura 4.28.

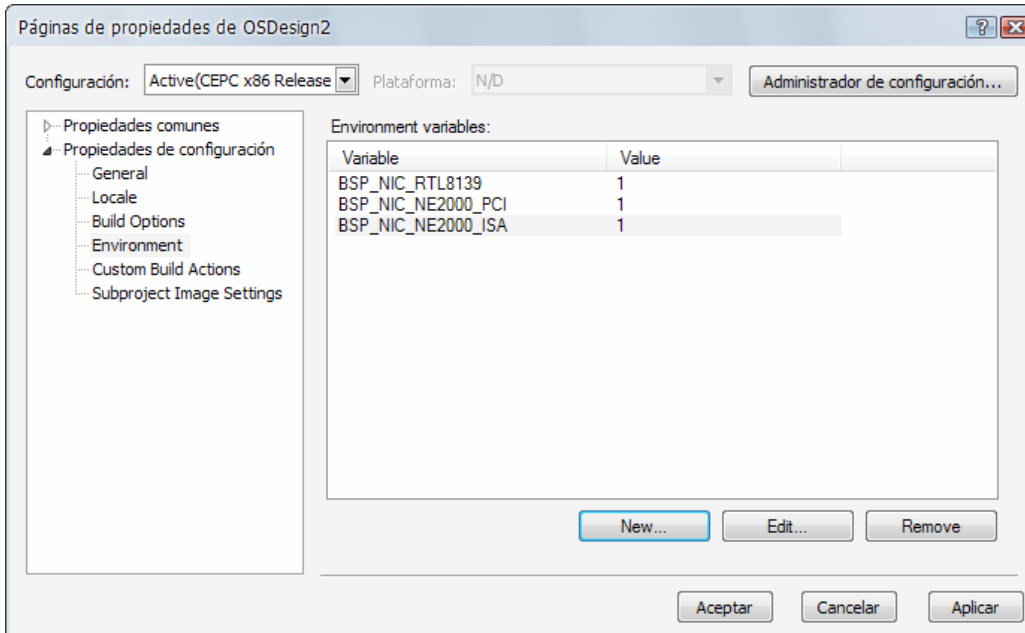


Figura 4.28. Finestra de variables d'entorn.

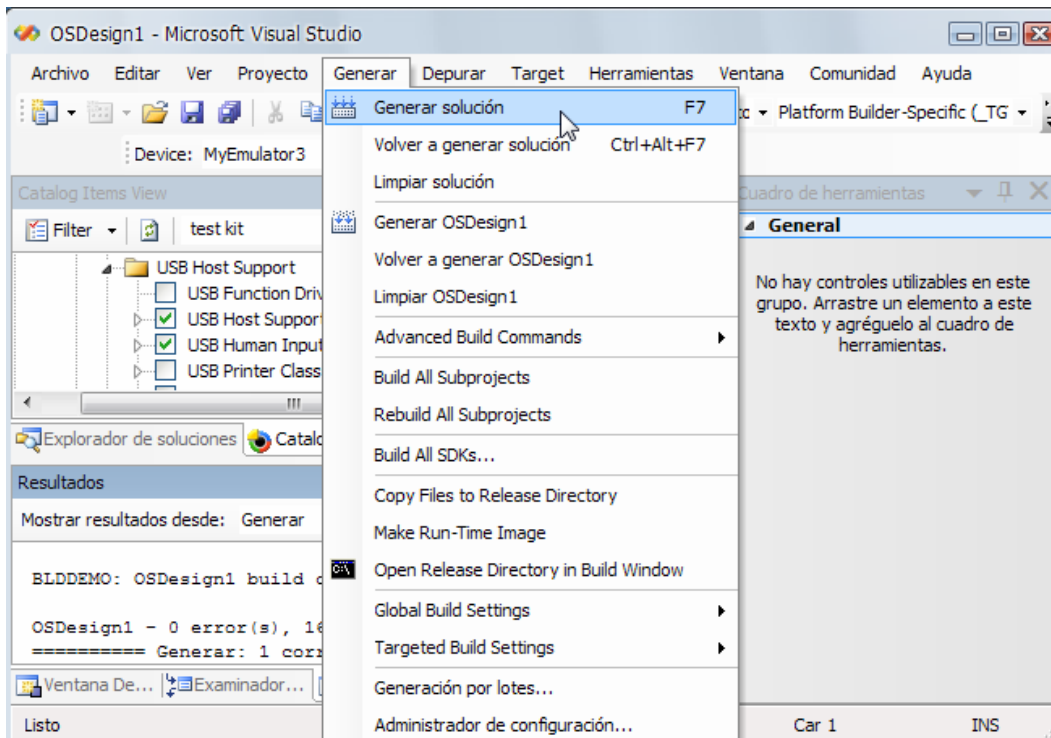


Figura 4.29. Menú Generar de VS2005.



Per a crear finalment la imatge executable de SO del BSP *CEPC x86*, selecciona del menú *Generar* la opció *Generar solució* (figura 4.28) i la imatge resultant quedarà guardada a la carpeta: **OSDesign1\RelDir\CEPC\_x86\_Release**.

#### 4.2.8. Connexió amb el dispositiu Target

A l'apartat 4.2.7 hem vist el procés de creació de les imatges de SO de Windows Embedded CE. En concret, per als BSPs *DeviceEmulator\_ARMV4I* i *CEPC\_x86*. Per a carregar aquestes imatges als dispositius *target*, necessitem establir connexions entre la *workstation* i aquests. En aquest apartat veurem el procés de connexió i descàrrega de les imatges als dispositius *target*.

Per a la connexió de l'estació de desenvolupament amb els dispositius *target* existeixen tres alternatives:

- Ethernet
- USB
- Sèrie

També hi ha disponible un emulador de dispositius, inclòs al VS2005. Aquest ofereix la possibilitat de simular el comportament dels SO dissenyats, sense la necessitat de transferir-los al *target*. És una eina molt útil per al test i depuració dels dissenys, durant el procés de desenvolupament.

L'emulador és instal·lat, com a una part més de Windows Embedded CE 6.0, juntament amb el *Platform Builder* quan es selecciona el suport a processadors ARM a les opcions d'instal·lació del programa. Per a utilitzar l'emulador com a dispositiu *target* per a la depuració d'un programa, la imatge ha de ser creada per a donar suport al BSP *DeviceEmulator*, emprant l'eina d'emulador de dispositius (DMA).

Tot seguit veurem un exemple del procés de connexió amb l'emulador mitjançant les eines de connectivitat que ofereix Microsoft Visual Studio 2005.

Obre l'administrador de configuracions del menú *Generar* i selecciona la opció *DeviceEmulator ARMV4I Release*. A continuació fes clic a *Cerrar* per a guardar els canvis.

## Creació d'un perfil de dispositiu Target

Es poden establir múltiples perfils de dispositiu per a ajudar a reduir el temps de desenvolupament al treballar amb diversos projectes. Cada perfil és configurat per a donar suport a un dispositiu *target*. Quan s'escull com a mitjà de descàrrega de la imatge una connexió Ethernet, el servei de connectivitat utilitza el controlador d'Ethernet MAC (*Media Access Control*) per a identificar el dispositiu i només descarregar la imatge configurada per a treballar amb el seu perfil actiu.

Ara crearem un perfil per a treballar amb l'Emulador i l'anomenarem **MyEmulator**. Selecciona al menú *Target* el submenú *Connectivity Options* per a obrir l'assistent *Target Device Connectivity Options* (figura 4.30).

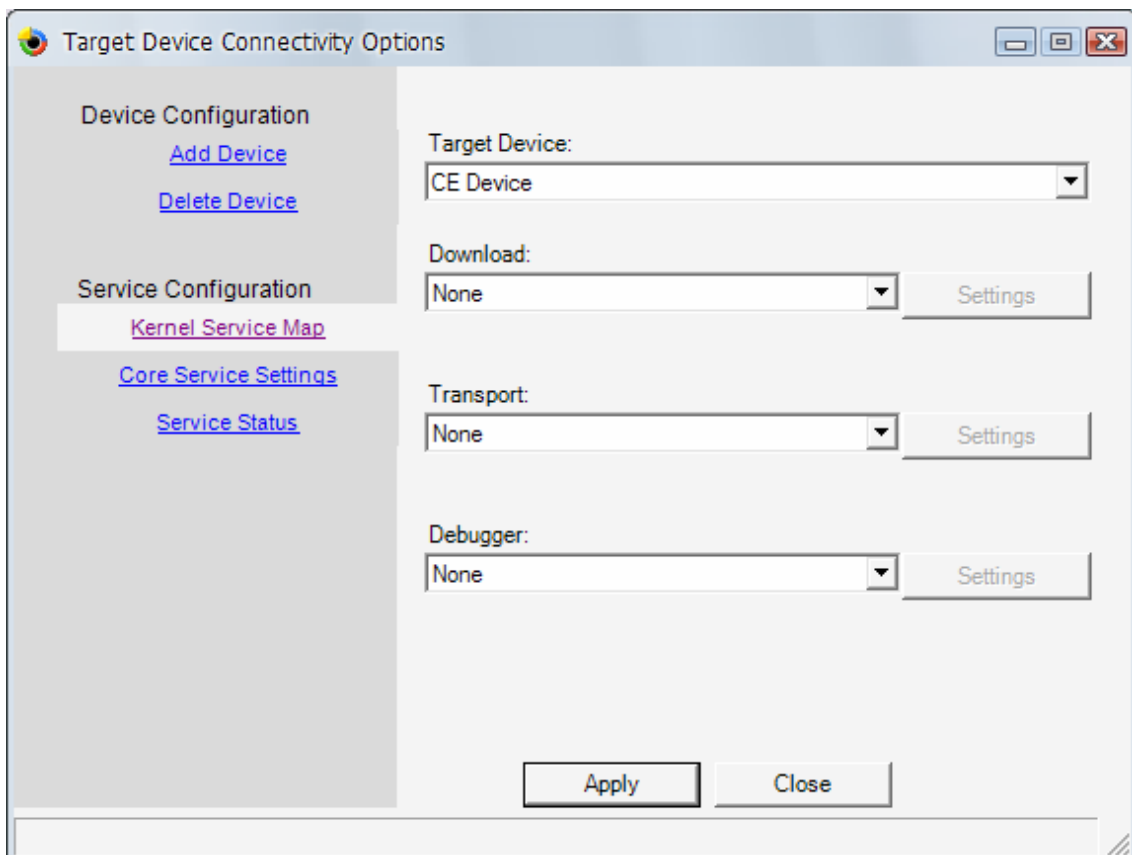


Figura 4.30. Target Device Connectivity Options.

Ara selecciona la opció *Add Device* i s'obrirà la finestra de la figura 4.31. Introdueix el nom a la casella *Target Device* i cliqueja el botó *Add*.

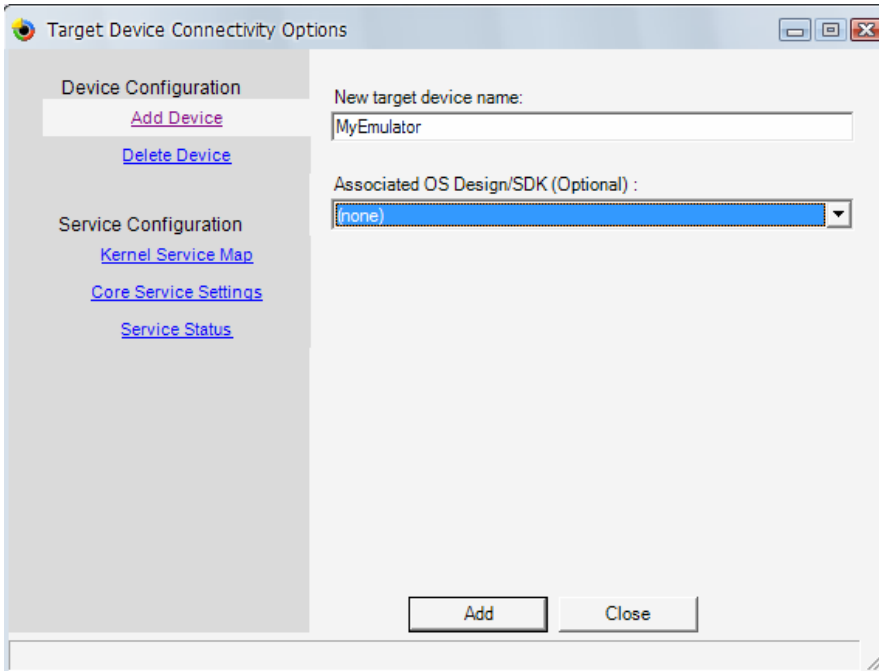


Figura 4.31. Entrada d'un nou dispositiu.

Selecciona ara la opció *Device Emulator (DMA)* a la caselles *Download* i *Transport*. A la casella *Debugger* deixa la opció seleccionada per defecte *KdStub*.

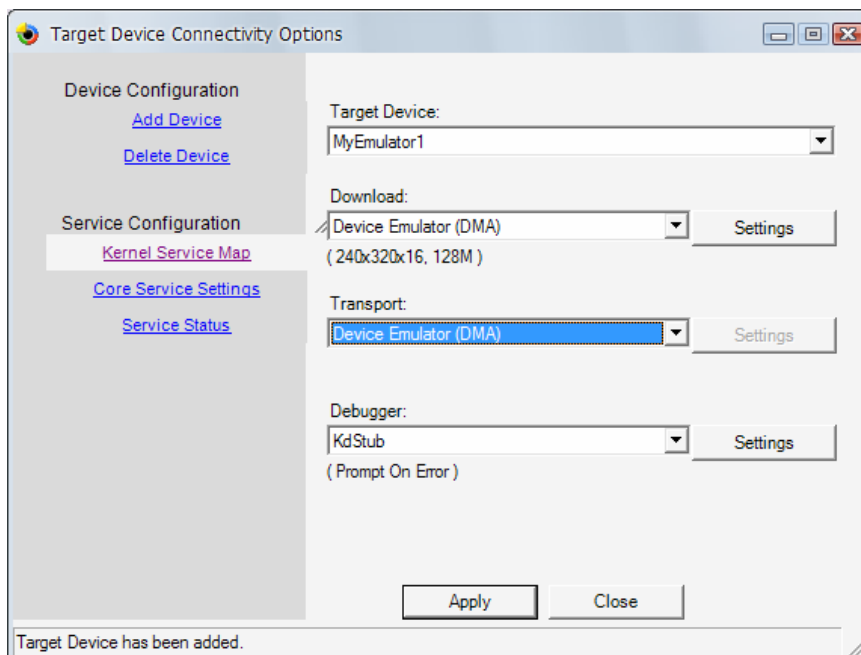


Figura 4.32. Configuració d'un nou dispositiu.

Cliqueja ara el botó *Settings* situat a la dreta de la casella *Download* per accedir a la pantalla de propietats de l'emulador (figura 4.33).

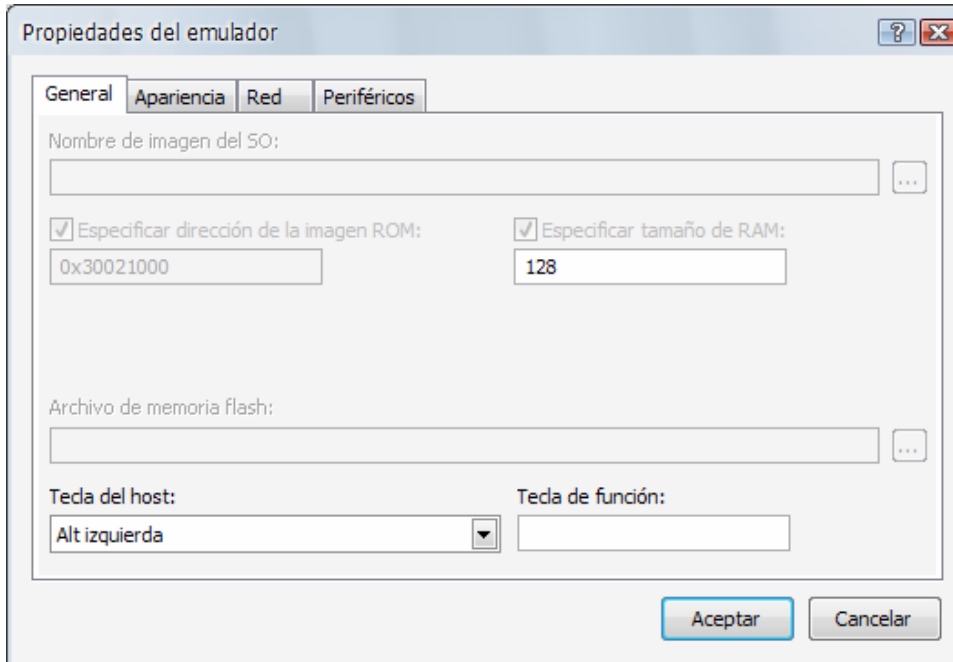


Figura 4.33. Propietats de l'emulador.

La imatge executable final pot ser més gran de 32 MB. L'emulador, per tant, ha de ser configurat per a tenir almenys 64 MB de RAM de suport a la imatge.

A la pantalla *General* selecciona la casella per habilitar la opció *Especificar tamaño de RAM* a 128.

La pantalla del dispositiu pot ser configurada a diferents resolucions. En aquest exercici, introdueix una resolució de 640 x 300. Fes clic sobre la pestanya *Apariencia* i introdueix aquesta resolució.

També hem de configurar la interfície de comunicacions. Això ho pots fer cliquejant sobre la pestanya *Red*. Habilita la pestanya *Habilitar adaptador de red PCMCIA NE2000* i deixa la opció per defecte *Tarjeta de red conectada*.

Amb aquesta configuració de xarxa, l'emulador pot connectar-se a la xarxa a través de la connexió activa de l'estació de desenvolupament.

Finalment cliqueja sobre el botó *Apply* seguit del botó *Close* per a finalitzar l'assistent.

## Descàrrega de la imatge executable a l'emulador

Per a descarregar la imatge *runtime* a l'emulador selecciona del menú *Target* la opció *Attach Device*. En aquest moment apareixerà la finestra de progrés de la descàrrega, la com mostra la figura 4.34.

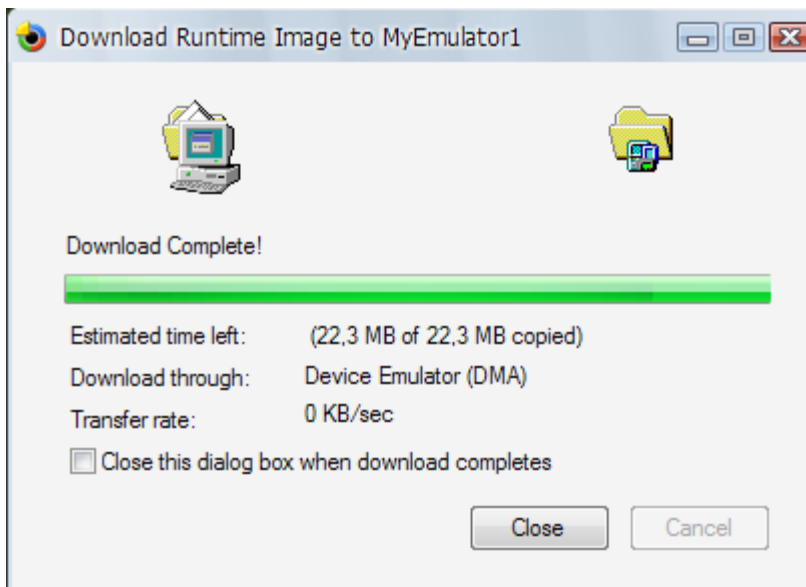


Figura 4.34. Finestra de progrés de la descàrrega de la imatge.

Finalment, apareixerà la pantalla de l'emulador (figura 4.35).

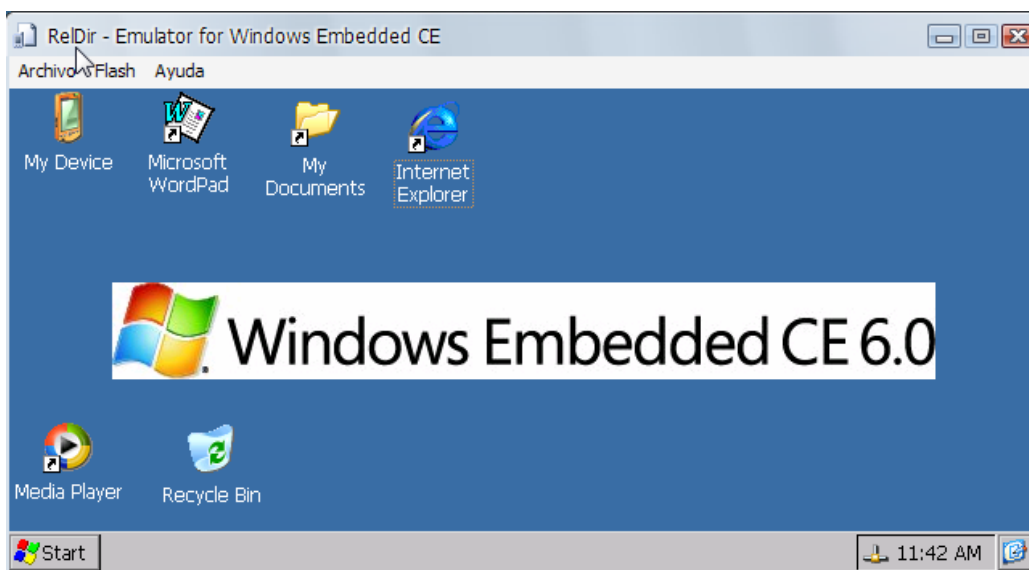


Figura 4.35. Emulador de dispositius de VS2005.

## Connexió amb el dispositiu CEPC

CEPC és una abreviació de computador PC funcionant amb Windows Embedded CE. El BSP CEPC està dissenyat per a proporcionar suport bàsic per a tots els dispositius construïts amb processadors x86 de 32 bits, amb un mínim de 32 MB de memòria.

Aquest CEPC pot proporcionar una imatge executable de SO que permeti el funcionament de Windows Embedded CE al teu vell PC. Els requeriments mínims del PC per al funcionament de la imatge CEPC són:

- 486 o més de processador x86.
- 64 MB o més de memòria RAM.
- Disquetera *bootable* (disponible a la majoria de PCs antics).
- Al menys un port sèrie (disponible a la majoria de PCs antics).
- Sortida VGA que suporti com a mínim una resolució de 640 x 480.

Idealment, si el PC disposa d'una targeta de xarxa compatible amb NE2000 o Realtek 8139, la descàrrega de la imatge es podrà efectuar més ràpidament. Si això no és possible, s'efectuarà una transferència per mitjà del port sèrie. La transferència a través del port sèrie es realitza amb l'Hyperterminal, inclòs a les versions Windows anteriors al Vista. Aquesta és bastant més lenta. La descàrrega de 20 MB d'imatge pot durar aproximadament 30 minuts.

## Creació d'un disquet d'arrencada CEPC

El primer que s'ha de fer és crear un disquet d'arrencada amb l'arxiu `bootloader.exe` i altres arxius associats. Un arxiu d'imatge de disquet, anomenat *CepcBoot.144*, i un programa d'utilitat, anomenat *makeimagedisk.exe*, per a l'extracció de la imatge al disquet, són inclosos com a part de Windows Embedded CE 6.0 durant la instal·lació. Aquests dos arxius es troben al directori: **\Program Files\Microsoft Platform Builder\6.00\cepb\utilities**.

Fes els següents passos per a crear el disquet d'arrencada CEPC:

1. Introdueix un disquet formatejat a la disquetera de la teva estació de desenvolupament.
2. Executa el programa *makeimagedisk.exe*.
3. A la pantalla de *MakeImageDisk*, cliqueja *Open* per a obrir la finestra de selecció d'arxiu. Obre el fitxer *CepcBoot.144* contingut al directori **\Program Files\Microsoft Platform Builder\6.00\cepb\utilities**.
4. Fes clic al botó *Start* per a crear la imatge CEPC al disquet.

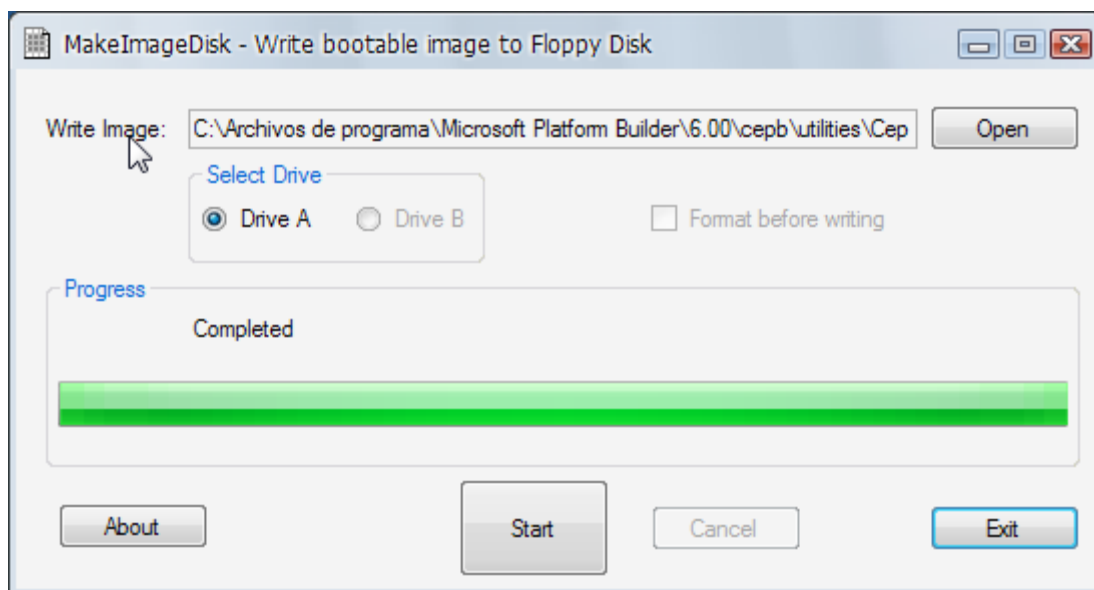


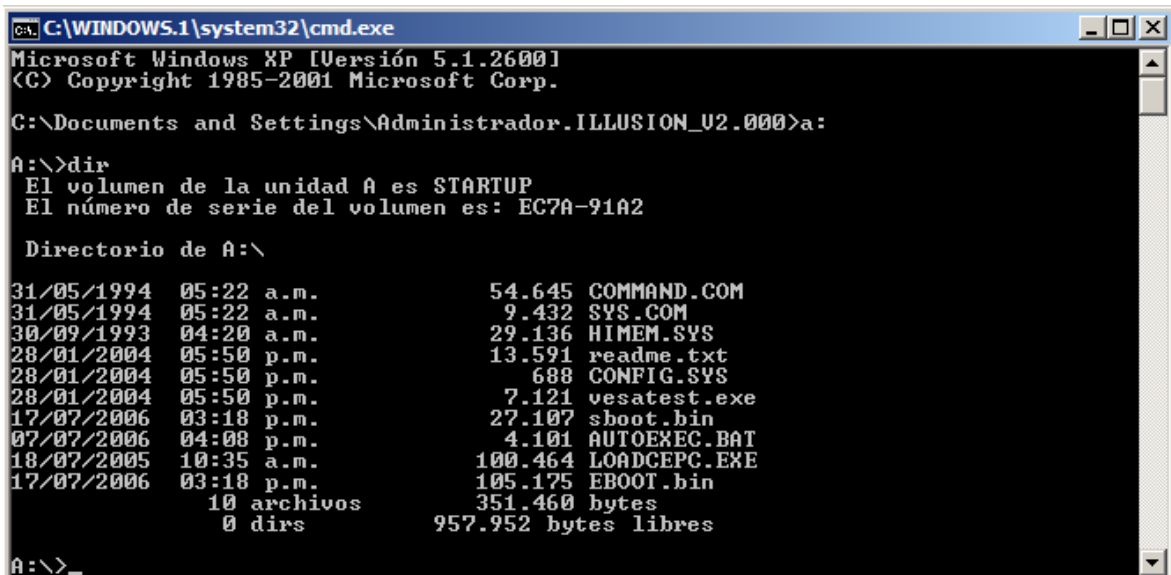
Figura 4.36. Aplicació *makeimagedisk* de VS2005.

A més dels arxius de sistema DOS, el disquet creat conté els següents arxius:

- **Eboot.bin.** Arxiu de càrrega via Ethernet per a CE.
- **Loadcepc.exe.** Arxiu de càrrega de DOS per a CE.
- **Sboot.bin.** Arxiu de càrrega via Sèrie per a CE.

- **Vesatest.exe.** Utilitat de test útil per les pantalles PC que no suporten el mode VESA
- **Config.sys.** Arxiu de configuració de DOS, responsable de llançar el *driver* del dispositiu altres llibreries del sistema durant l'inici del DOS.
- **Autoexec.bat.** Arxiu DOS que s'executa durant l'arrencada.

Comprova els arxius que conté el disquet. Una opció és entrar a la línia de comandaments de Windows i fer un dir per a fer aquesta comprovació (tal com mostra la figura 4.37).



```

C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Versión 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Administrador.ILLUSION_U2.000>a:
A:\>dir
El volumen de la unidad A es STARTUP
El número de serie del volumen es: EC7A-91A2

Directorio de A:\

31/05/1994  05:22 a.m.          54.645  COMMAND.COM
31/05/1994  05:22 a.m.           9.432  SYS.COM
30/09/1993  04:20 a.m.          29.136  HIMEM.SYS
28/01/2004  05:50 p.m.          13.591  readme.txt
28/01/2004  05:50 p.m.           688  CONFIG.SYS
28/01/2004  05:50 p.m.           7.121  vesatest.exe
17/07/2006  03:18 p.m.          27.107  shoot.bin
07/07/2006  04:08 p.m.           4.101  AUTOEXEC.BAT
18/07/2005  10:35 a.m.         100.464  LOADCEPC.EXE
17/07/2006  03:18 p.m.         105.175  EBOOT.bin
           10 archivos          351.460 bytes
           0 dirs             957.952 bytes libres

A:\>

```

Figura 4.37. Línia de comandaments de Windows XP.

Ara ja s'ha creat el disquet d'arrencada per a preparar el dispositiu per a la càrrega de la imatge de Windows Embedded CE.

Tal com ja s'ha explicat, existeixen diferents alternatives per a la descàrrega de la imatge al dispositiu. Seguidament realitzarem un exercici d'exemple per la transferència de la imatge a un dispositiu x86 via Ethernet, mitjançant el protocol de xarxa DHCP.



El DHCP (*Dynamic Host Configuration Protocol*) és un protocol de xarxa que permet als nodes d'una xarxa IP obtenir els seus paràmetres de configuració automàticament. Es tracta d'un protocol de tipus client/servidor en el qual generalment un servidor posseeix una llista d'adreces IP dinàmiques i les va assignant als clients conforme aquestes van estant lliures, sabent en tot moment qui ha estat en possessió d'aquesta IP, quant temps l'ha tingut i a qui ha estat assignada després.

Per a realitzar la transferència és necessari un router de xarxa i dos cables TIA/EIA 568-B.2. El router i el tipus de cable especificat és l'habitualment subministrat per a les connexions domèstiques d'ADSL.

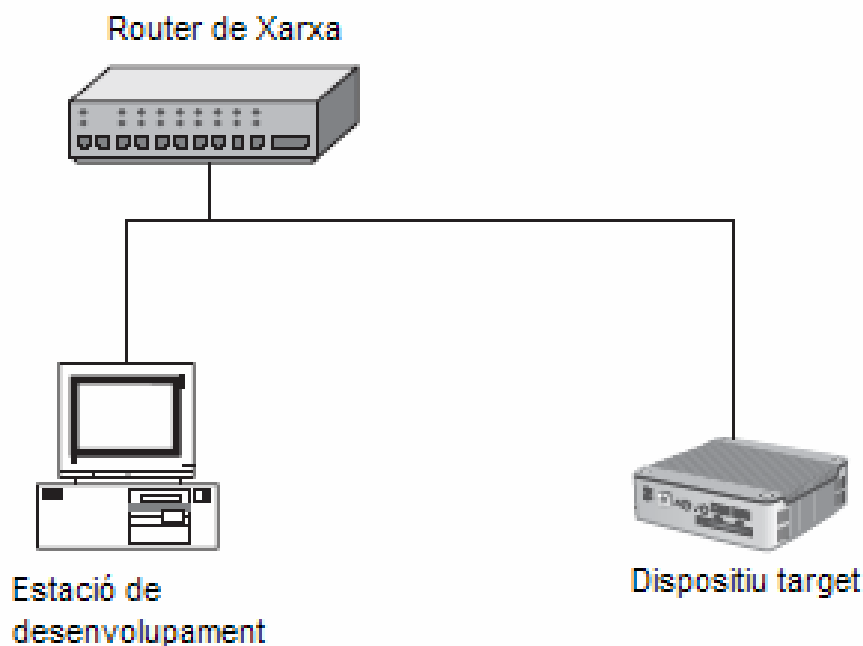


Figura 4.38. Esquema de connexió física amb el dispositiu target via Ethernet.

També es pot realitzar una comunicació amb un sol cable crossover Ethernet establint manualment una IP estàtica. Per a més informació sobre aquest sistema consulta el llibre "Microsoft Embedded CE 6.0" de edicions Wrox, escrit per Samuel Phung; d'on s'ha estret gran part de la informació continguda en aquest apartat

Treballant amb el mateix projecte creat als exercicis anteriors, has de seleccionar la configuració activa BSP\_CEPC\_Release a l'administrador de configuracions.

Per a crear un perfil de dispositiu target per a la connexió amb el dispositiu seguix els següents passos:

1. Al menú *Target*, cliqueja el submenú *Connectivity Options*.
2. Dins de la finestra *Connectivity Options* fes clic sobre *Add Device*.
3. Introdueix un nom per al perfil del dispositiu, per exemple el model del dispositiu *target*. En el cas d'aquest projecte, la connexió es realitza amb un PC industrial Advantech PCM-9575. Cliqueja *Add* per a continuar.
4. Dins de la finestra *Connectivity Options* selecciona la opció *Ethernet* a les pestanyes *Download* i *Transport*, tal com mostra la figura 4.39.

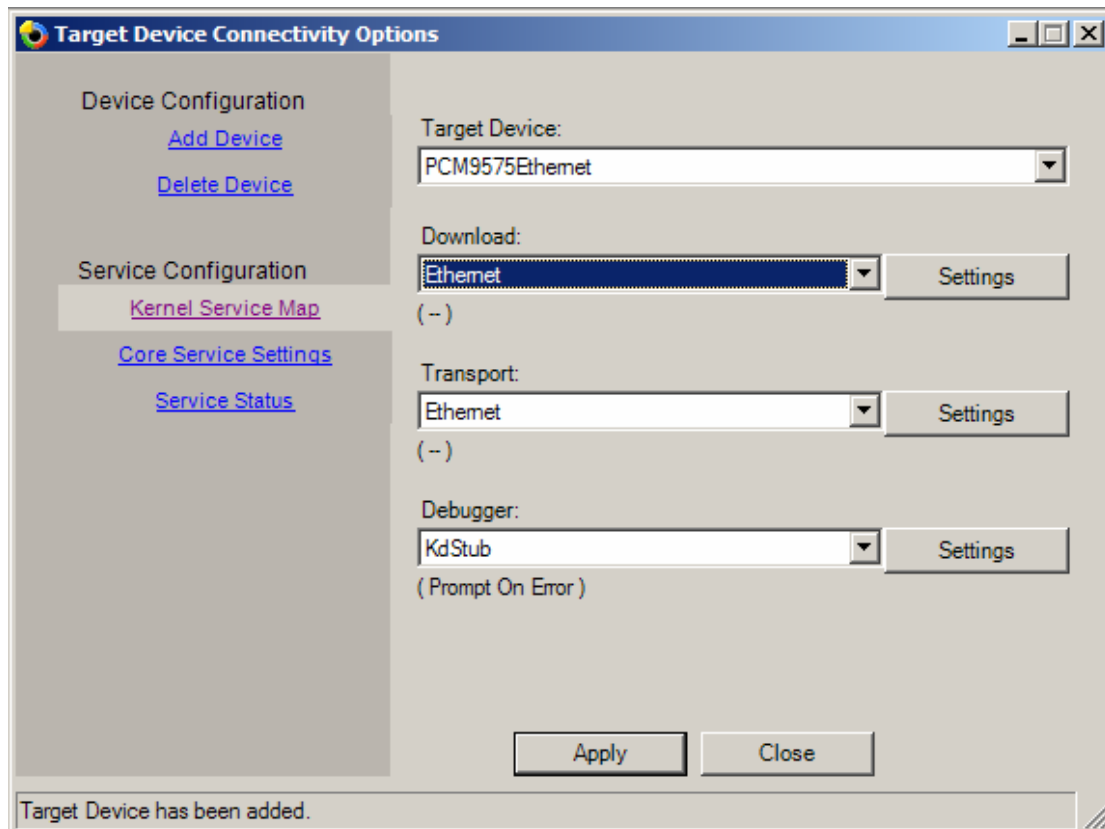


Figura 4.39. Finestra d'opcions de connectivitat de VS2005.

5. Cliqueja el botó *Settings* situat a la dreta de la pestanya *Download*. En aquest moment s'obre la finestra *Ethernet Download Settings* (figura 4.41).

En aquest punt, el dispositiu target no està associat encara a cap perfil. En els següents passos es connectarà el dispositiu *target* amb l'estació de desenvolupament per a identificar-lo i assignar-li un perfil de dispositiu.

6. Encén el dispositiu *target* x86 amb el disquet d'arrencada anteriorment creat introduït a la disquetera.

Nota: És possible que hagi d'accedir al BIOS del dispositiu per a establir la disquetera com a primer dispositiu d'arrencada del sistema.

Un cop arrencat l'equip, accedeix als arxius d'arrencada del disquet i els executa. En aquest punt apareix una pantalla amb diferents opcions.

7. Selecciona la opció 2: *Load OS image from development station with DHCP service*.

En aquest punt, el processador executa la línia de codi de l'arxiu `autoexec.bat`:

```
Loadcepc /L:1024x768x32 /C:0  
/e:0:0:0 eboot.bin
```

El dispositiu target comença a enviar dades pel cable sèrie i apareix un nom assignat al dispositiu a la finestra *Active target devices* de la pantalla *Ethernet Download Settings* (figura 4.40).

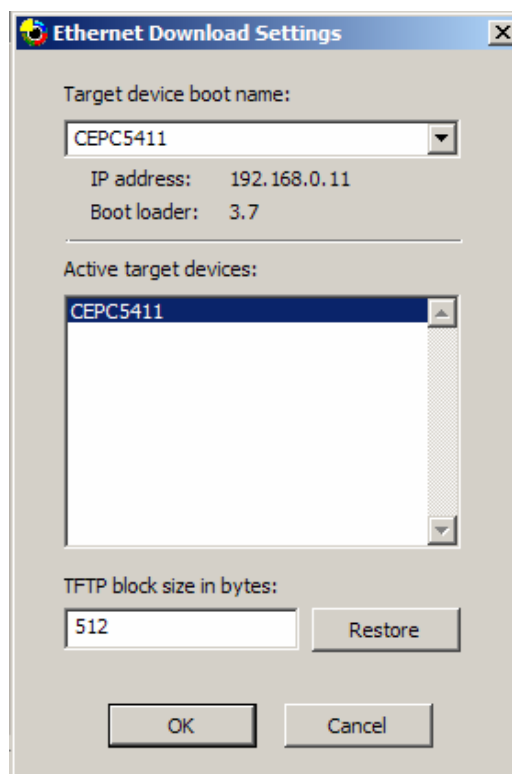


Figura 4.40. *Ethernet Download Settings*.

En el cas del PC Advantech PCM-9575 utilitzat en aquest projecte apareix el nom de dispositiu: CEPC5411.

8. Selecciona el dispositiu detectat de la finestra *Active target devices* i cliqueja *OK*. Finalment fes clic a *Apply*, seguit de *Close* per finalitzar l'assistent.
9. Ara anem a connectar el dispositiu. Selecciona al menú *Target* la opció *Attach Device*.

En aquest moment s'estableix la comunicació i s'efectua la transferència de la imatge executable de SO al dispositiu target. Pots utilitzar les eines remotes de depuració, que es mostren a continuació, per a comprobar el correcte funcionament del dispositiu target, mentre es mantinguin connectats l'estació de desenvolupament i el dispositiu target.

#### **4.2.9. Eines i entorns de depuració**

A la majoria dels casos, en el desenvolupament de sistemes *embedded* hi ha diferents opcions i mètodes per assolir els objectius fixats per al disseny. Durant el procés de desenvolupament, es prenen decisions basades en criteris diversos: rendiment del hardware, consum de recursos, compatibilitat amb altres plataformes, etc.

Aquest procés pot ser llarg i complex i un bon entorn de depuració d'errors i testejat del software pot ajudar molt.

Windows Embedded CE Platform Builder i Visual Studio 2005 proporcionen un entorn de desenvolupament eficient, amb eines per a la depuració, que ajuda a minimitzar el temps emprat en la resolució d'errors i el testejat dels dissenys.

A més dels recursos per a la depuració que facilita Microsoft, existeixen eines de depuració facilitades per alguns fabricants de hardware *embedded*. Aquestes, no obstant, requereixen de la instal·lació d'un software addicional, que no està integrat dins de l'entorn de desenvolupament de Windows Embedded CE.

## Depuració d'un disseny de SO

Un disseny de SO de Windows Embedded CE pot ser configurat per a generar una imatge executable en mode depuració (*debugging-mode*) o mode llançament (*release-mode*). A la majoria dels casos, una configuració en *release-mode* pot ser suficient per a disposar de la majoria de funcions per a la depuració. Això és important, ja que una imatge de SO generada en mode depuració pot ocupar el doble d'espai que la imatge del mateix projecte generada en mode llançament. A més, requereix de més recursos computacionals, i triga més a generar-se. Això és degut a que una imatge generada en mode depuració crea molts més missatges de depuració que si és generada en mode llançament.

Com has pogut observar als apartats anteriors del tema 2.2; les imatges dels dissenys de SO realitzats als exercicis d'exemple, s'han generat en mode *release* amb KITL (*Kernel Independent Transport Layer*).

Els missatges de depuració addicionals proporcionen més detalls sobre els components i mòduls de la imatge; però probablement només siguin necessaris quan es dissenya amb un llenguatge de baix nivell, molt proper al hardware del dispositiu. Per a la resta de casos, la imatge generada en mode *release*, amb KITL, aporta el missatges de depuració necessaris per al dissenyador.

## Errors del disseny

Un dels problemes més habituals en el desenvolupament amb Windows Embedded CE és que no es completi el procés de generació de la imatge. Aquest problema pot ser causat per: una entrada incorrecta a l'arxiu de configuració (de la imatge, subprojecte o plataforma), un arxiu no trobat, o una referència de directori incorrecta d'un arxiu necessari.

Si la creació de la imatge no s'ha pogut completar degut a errors durant el procés, hauràs de trobar la causa dels mateixos als arxius:

- **Build.err.** Conté la informació dels errors i només es genera si el procés no es completa amb èxit.

- **Build.log.** Anota totes les activitats realitzades durant el procés de creació de la imatge.
- **Build.wrn.** Conté informació sobre els advertiments. Si un procés no genera cap *warning*, aquest arxiu no es crea.

Aquests arxius es troben al directori `WINCEROOT\` i contenen informació sobre la última imatge creada. Un nou procés de creació eliminarà els arxius *build.err*, *build.log* i *build.wrn* del procés anterior.

És important tenir en compte aquest detall, donat que en un projecte gran és recomanable fer una còpia de seguretat dels tres arxius, en cada procés de creació de la imatge. Això permet comparar els arxius *build.err*, *build.log* i *build.wrn* de diferents sessions de creació i detectar problemes.

També es poden extreure conclusions a partir dels missatges generats a la finestra de sortida de VS2005. Aquesta ens va mostrant, durant el procés de creació de la imatge, les accions que s'estan portant a terme i els missatges d'error i advertència (en cas de que hi hagi).

Però quan es tracta de problemes complexos, és necessari conèixer el procés de creació de la imatge, per saber on i quan es generen els errors.

## Procés de creació de la imatge

El procés de creació de la imatge està format per cinc fases:

- [1] **Pre-SYSGEN.** El sistema de creació de la imatge compila el codi proporcionat per Microsoft i el diposita al directori *Public*.

- [2] **SYSGEN (System Generation)**. La fase de generació del sistema filtra els components seleccionats per al disseny d'una llista de components disponibles i enllaça les llibreries de components amb mòduls DLL o EXE.
- [3] **Post-SYSGEN**. Es compila el codi de BSPs ,*drivers* de dispositius, i codi d'arrencada del sistema.
- [4] **Build Release**. Durant aquesta fase, els arxius creats a les dues fases anteriors es copien en un únic directori, el directori de llançament.
- [5] **Creació de la imatge executable**. Es genera la imatge executable Windows Embedded CE.

Durant el procés de creació de la imatge, el *Platform Builder* genera els següents arxius:

- **DIRS**. L'arxiu DIRS identifica els subdirectoris del projecte que contenen codi font.
- **Sources**. Conté les macros requerides per a construir el codi font.
- **Makefile**. Conté les variables requerides per a per a compilar i enllaçar el codi font.
- **BIB**. L'arxiu BIB (*Binary Image Builder*) conté informació sobre la configuració del sistema de memòria i entrades de control en que eles fitxers.

## Eines de depuració remotes

El *Platform Builder* de Windows Embedded CE proporciona una sèrie d'eines remotes per ajudar al dissenyador a la depuració dels projectes. Aquestes eines poden accedir a la imatge de SO que s'està executant al dispositiu *target*, de forma remota desde l'estació de desenvolupament. Aquestes eines es descriuen a continuació:

- **File Viewer.** Permet veure, de forma remota, els arxius del dispositiu *target*; d'igual manera que ho permet l'explorador de Windows de forma local.
- **Heap Walker.** S'utilitza per a examinar la disposició i continguts a la memòria de cada procés funcionant al dispositiu *target*.
- **Zoom.** Captura la imatge de la pantalla del dispositiu *target*, en format bitmap (.bmp). La imatge capturada es pot emmagatzemar a la *workstation*; facilitant així la documentació dels projectes.
- **Process Viewer.** Mostra tots els processos en execució del dispositiu *target*, i els seus mòduls associats.
- **Registry Editor.** Mostra el registre del dispositiu *target* i possibilita l'addició, eliminació i modificació de les entrades del registre. És similar a l'editor del registre d'un Windows *desktop*.
- **System Information.** Mostra informació del sistema, configuracions i propietats del dispositiu *target*.
- **Performance Monitor.** Eina gràfica per a la mesura del rendiment del sistema. Rastreja l'activitat del sistema i crea un arxiu d'informació de la mateixa.
- **Spy.** Permet veure els missatges entregats a processos en execució del dispositiu *target*.
- **Kernel Tracker.** Proporciona una representació gràfica dels esdeveniments de les aplicacions i la imatge executable que estan succeint al dispositiu *target*.
- **Call Profiler.** És una eina de configuració del perfil d'aplicacions del dispositiu *target*.

Les eines remotes de depuració de VS2005 es poden executar seleccionant dins el menú *Target*, el submenú *Remote Tools*.



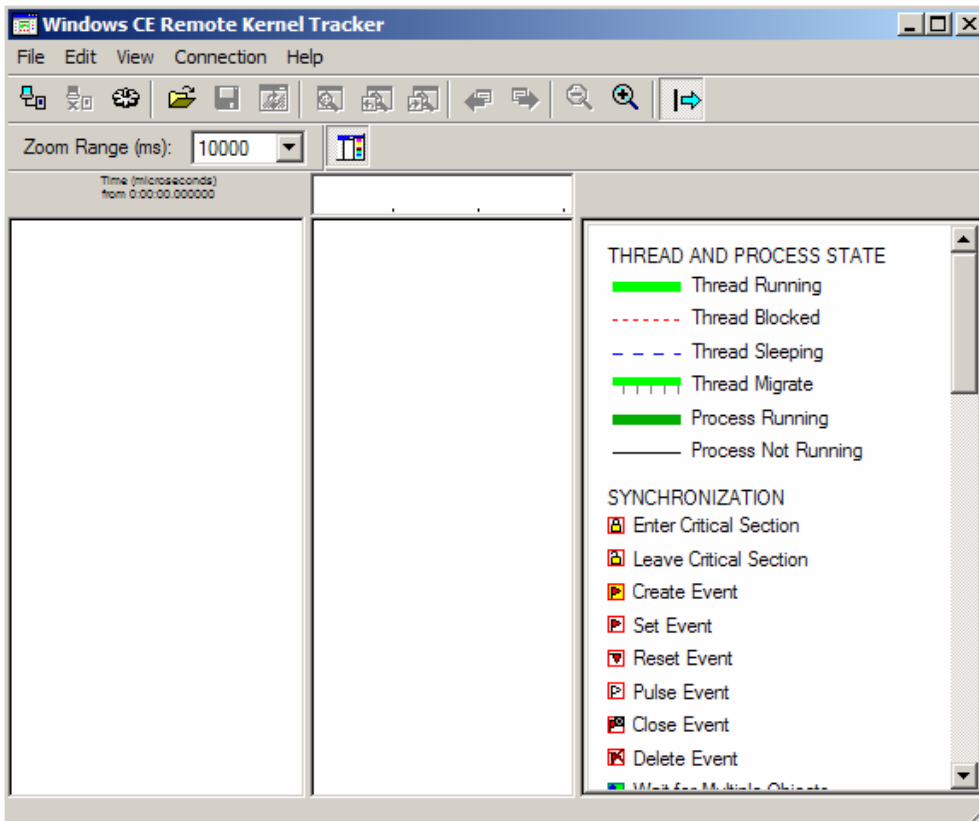


Figura 4.41. Pantalla principal de Kernel Tracker.

## Windows Embedded CE 6.0 Test Kit

També és pot depurar un projecte mitjançant el CETK (*CE Test Kit*), que proporciona Windows Embedded CE 6.0. Aquesta utilitat proporciona un entorn de depuració independent, que ens permet testejar els BSPs i *drivers* de dispositius.

Per a obrir el CETK desde Windows selecciona al menú *Inicio* ⇨ *Todos los Programas*  
⇨ *Windows Embedded CE 6.0* ⇨ *Windows Embedded CE 6.0 Test Kit*.

## Depuració sèrie

El port sèrie és un altre mitjà disponible per al suport a la depuració. Això es fa possible connectant un cable sèrie null modem RS-232 entre el port sèrie del dispositiu *target* i el port sèrie de la *workstation*.

Mitjançant l'eina *Hyperterminal* del teu Windows *desktop*, pots rebre a la teva *workstation* missatges de depuració provinents del dispositiu *target*.

## 4.3 – Creació de la imatge executable de SO per al PCM-9575

### 4.3.1. Introducció

Per a la realització d'una aplicació real al projecte s'ha emprat el PCM-9575, un PC industrial del fabricant Advantech, disponible a l'Escola Universitària Politècnica de Mataró.

La seva placa base amb un factor de forma EBX 5,25", integra un computador en una sola targeta. Es tracta doncs d'un SBC (*Single Board Computer*) que suporta ampliacions amb mòduls ISA/PCI de PC/104.

El seu processador de baix consum Eden 667MHz utilitza tecnologia CMOS amb 128 KB L1 i 64KB L2 de memòria cache i pot treballar sense ventilador a temperatures superiors als 60° C i un consum inferior als 14 Watts. A més, l'Eden 667MHz es compatible amb sistemes operatius i aplicacions de l'arquitectura x86, la més estesa del mercat *desktop*.

Es tracta doncs d'un PC industrial que integra grans funcionalitats amb una requisits de consum i grandària raonablement òptims.



Figura 4.42. PCM-9575 board.

### 4.3.2. Característiques principals del PCM-9575

Les principals característiques del PC industrial PCM-9575 es descriuen a continuació:

- Processador embedded de baix consum VIA Eden/Ezra.
- Placa base amb factor de forma EBX amb suport per a PC/104-Plus i ranures PCI
- Suport a panells TFT LCD de 36 bits, amb resolució màxima de 1024 x 768.
- Suport a interfície de dos canals LVDS.
- Interfície Ethernet 100/10Base-T compatible amb IEEE 802.3U
- Funció TV-out amb suport als formats NTSC i PAL
- 4 ports sèrie (tres RS-232 i un RS-232/422/485).
- Interfície USB 1.10
- Timer Watchdog amb 62 nivells amb reset del sistema o IRQ
- 2 canals IDE per a HDD o CD-ROM.

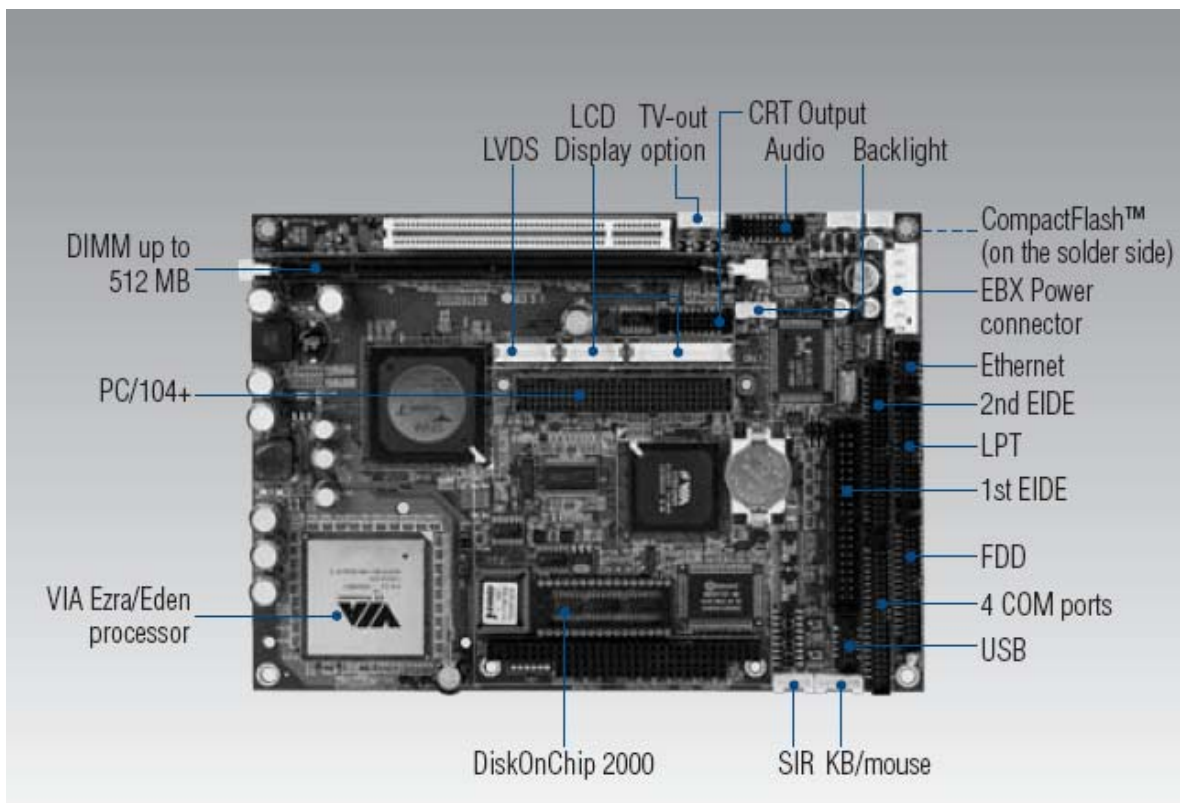


Figura 4.43. Ports i característiques del PCM-9575.

### 4.3.3. Configuració del PCM-9575

Per a poder transferir la imatge executable de SO s'ha de configurar el PC Advantech PCM-9575 de forma que aquesta comunicació i transferència sigui possible. El primer pas que s'ha de fer és identificar el hardware que compon el PCM-9575 i conèixer els requisits per a poder transferir la imatge de SO a aquest dispositiu.

Els requisits per a establir la connexió i transferència de la imatge de SO creada amb el BSP (*Board Support Package*) anomenat CEPC (PC amb sistema operatiu Windows CE) són els següents:

- CPU x86 486 o superior (o processadors compatibles).
- Mínim 64 MB de memòria de sistema
- Disc dur IDE arrencable amb DOS al canal primari IDE configurat com a mestre
- Disquetera i disquet per arrencar el PC amb DOS
- Port sèrie RS-232 o targeta Ethernet compatible amb NE2000 o Realtek-8139.

Si el PC Embedded aconsegueix les condicions anteriors, com és el cas del PCM-9575, es pot utilitzar el BSP\_CEPC per a crear la imatge. En cas contrari s'ha d'investigar quin BSP és idoni per al PC Embedded. És probable que algun BSP instal·lat per defecte a Windows Embedded CE 6.0 ja sigui adequat per al PC industrial. En cas contrari, és important obtenir el BSP concret; a través de la pàgina web <http://www.microsoft.com/windowseembedded/en-us/msdn/bspsearch.aspx> o a través del fabricant.

D'entrada el PCM-9575 no incorpora disquetera, però sí el connector a la placa SBC per a la connexió d'una. No es tracta d'un connector estàndard d'un PC *desktop*. Es tracta d'un connector per a cable pla FFD 34 (2.0). El cable s'ha comprat al distribuïdor oficial d'Advantech a Barcelona per a possibilitar la realització d'aquest projecte.

Un cop realitzades totes aquestes comprovacions ja es pot realitzar la creació de la imatge executable personalitzada i la seva transferència al PCM-9575.

#### 4.3.4. Transferència de la imatge al PCM-9575

La transferència s'efectua amb la imatge del BSP\_CEPC creada i explicada detalladament al capítol 4.2. Aquesta es pot efectuar de diferents maneres, però en aquest projecte s'ha efectuat mitjançant una connexió Ethernet, amb el protocol DHCP. La metodologia per a la transferència de la imatge mitjançant DHCP també està detalladament explicada al capítol 4.2 d'aquest projecte, amb una sèrie de variants que s'especifiquen a continuació.

Nota: Per a reduir la grandària de les explicacions, el dispositiu *target* PCM-9575, utilitzat en aquest projecte, es designarà CEPC (PC amb sistema operatiu CE).

#### Configuració del disc dur amb BIOS Boot Loader

Per a preparar el disc dur per a que el PC Embedded arranqui amb la imatge executable de Windows Embedded CE 6.0 són necessaris dos disquets formatejats. Amb les utilitats *Setupdisk.144* i *Bootdisk.144* es crearan uns discs d'arrencada amb DOS que ens permetran configurar l'arrencada del disc dur, mitjançant l'aplicació Biosloader. Per a realitzar aquesta operació segueix els següents passos:

1. Executa l'arxiu *MakeImageDisk.exe* contingut a la carpeta **\_Program Files\Windows CE Platform Builder\6.00\CEPB\Utilities**.  
Amb aquesta utilitat, pots crear discos d'arrencada amb arxius d'imatge. Aquesta utilitat està inclosa al paquet de Windows Embedded CE 6.0.
2. Introdueix un disquet formatejat a la disquetera de la teva estació de treball. Obre l'arxiu *Setupdisk.144* contingut a la carpeta **\_WINCEROOT\<nom\_del\_projecte>\CEPC\src\bootloader\BIOS Loader\DiskImages** amb la utilitat *MakeImageDisk*, i cliqueja *Start*.  
Amb això es crea un disquet d'arrencada que prepara el disc dur del PC embedded amb el BIOS boot loader.

3. Introdueix el disquet creat a la disquetera del CEPC i reinicia el CEPC.
4. El CEPC es s'inicia, i entre els diferents arxius que té el disquet, està l'arxiu *fdisk*. Utilitza la utilitat *fdisk* per a eliminar totes les particions existents al disc dur, i crear una nova partició activa.
5. Quan la utilitat et demani especificar un nom d'etiqueta per a la nova partició, pressiona ENTER per a no especificar cap nom d'etiqueta.

L'etiqueta del volum no ha de contenir cap caràcter.

El *BIOS boot loader* reconeix particions en format FAT12 i FAT16. La utilitat format del Setupdisk.144 formateja la partició en format FAT16.

6. Reinicia el CEPC.
7. Executa format <lletra de la unitat>:

Això formateja el disc dur.

8. Quan la utilitat et demani especificar un nom d'etiqueta per a la nova partició, pressiona ENTER per a no especificar cap nom d'etiqueta.

L'etiqueta del volum no ha de contenir cap caràcter.

9. El programa Mkdisk.bat està contingut dins el disquet Setupdisk.144. Escriu mkdisk seguit de la lletra que correspon a la unitat del disc dur i dos punts. Per exemple: **mkdisk c:**.

Si executes el Mkdisk.bat sense especificar la unitat, aquest danyarà el disquet i hauràs de començar de nou pel pas 1.

En aquest punt, el disc dur està ja preparat per a carregar una imatge de Windows Embedded CE.

L'arxiu *boot.ini* conté el nom de la imatge a carregar. Per defecte, el tag *BinFile* especifica *NK.bin* com a arxiu d'imatge, i el tag *BakBinFile* especifica *Eboot.bix* com a segona imatge. Si l'arxiu *Eboot.bix* no està comprimit, té l'extensió d'arxiu *.bin*. Per defecte, el programa Mkdisk copia l'arxiu *Eboot.bix* al disc dur.

10. Utilitza l'*Eboot.bix* per carregar la imatge de Windows Embedded CE a través d'Ethernet. Després utilitza la imatge carregada de Windows Embedded CE per a copiar la imatge executable sencera al disc dur.

## Creació i transferència de la imatge executable al disc dur

Per a copiar la imatge executable al disc dur, introdueix un *driver* i un arxiu de sistema del disc dur a la imatge executable. Quan la imatge executable resideix al disc dur, aquesta pot arrencar al dispositiu target sense necessitat de connexió amb l'estació de desenvolupament.

Per a crear i transferir una imatge executable al disc dur, segueix els següents passos:

1. Afegeix suport a dispositius ATAPI a la imatge executable.

A la vista del catàleg d'ítems del VS2005, introdueix a la casella de recerca *IDE Storage Block Driver* i pressiona ENTER.

Apareixerà l'ítem corresponent. Selecciona'l per a incloure'l a la solució de la imatge.

2. Afegeix també l'ítem *FAT File System* a la solució de la imatge. Pots buscar-lo amb l'eina de recerca d'ítems o dins del catàleg a **Core OS\Windows Embedded CE Devices\File Systems and Data Store\Storage Manager**.
3. Genera la imatge executable. Menú *Generar*, submenú *Generar solució*.

4. Reinicia el CEPC i descarrega la imatge executable de l'estació de desenvolupament al CEPC.
5. Un cop completada la descàrrega, copia l'arxiu *Nk.bin* al directori de llançament del disc dur.

Per a simplificar la recerca de l'arxiu de la imatge executable al CEPC, si la imatge inclou interfície d'usuari, a l'explorador d'arxius (*File Explorer*) de Windows Embedded CE, dins el menú *View*, escull *Folder Options*, i elimina la opció *Hide File Extensions*.

6. Reinicia de nou el CEPC amb el disquet creat dins la disquetera.
7. Canvia els directoris al disc dur del CEPC, especificant *<lletra\_de\_la\_unitat>:*.
8. Executa el comandament de DOS *dir* i verifica que l'arxiu *NK.bin* està present al disc dur.
9. Si el nom de l'arxiu ha estat canviat o està utilitzant un nom curt, com *NK~1.bin*, renombra l'arxiu com a *NK.bin* escrivint *ren NK~1.bin NK.bin*.

Finalment la imatge executable de sistema operatiu queda instal·lada al disc dur primari del PCM-9575 i ja té funcionalitat com a sistema operatiu; podent-se agregar addicionalment aplicacions amb una funcionalitat concreta.

Si desitges ampliar coneixements respecte a Windows Embedded CE 6.0 pots consultar el llibre "Professional Microsoft Windows Embedded CE 6.0" de edicions Wrox, escrit per Samuel Phung; d'on s'ha estret gran part de la informació continguda al capítol 4.2.

També pots extreure informació útil a la pàgina web <http://msdn.microsoft.com/> on s'expliquen metodologies i passos a donar per a la creació i transferència d'imatges.



## 5 CONCLUSIONS

El principal objectiu que perseguia d'aquest projecte era la introducció al coneixement dels sistemes embedded; donada la seva creixent importància al món industrial. La gran quantitat de matèries que s'ha d'impartir a la carrera d'Enginyeria Tècnica Industrial, en un volum d'hores determinat, no han permès un estudi complet d'aquests dispositius. No obstant, el fet de destinar el Projecte Final de Carrera a aquesta temàtica ha permès una introducció al món embedded.

Es tracta d'una filosofia d'aplicació tecnològica àmpliament estesa a l'àmbit industrial, amb desenes de pàgines web i llibres destinats a aquesta matèria. A més, els sistemes embedded estan en continu creixement dins el mercat.

Durant el desenvolupament d'aquest projecte s'han pogut adquirir una gran quantitat de coneixements que es poden resumir en els següents punts:

- Plataformes hardware de sistemes embedded, formats utilitzats i arquitectures de plaques bases i processadors; així com de les principals interfícies de comunicacions.
- Principals sistemes operatius i eines software disponibles al mercat per aplicacions embedded.
- Estat de l'art del mercat embedded, amb els principals fabricants, tant de software com de hardware.
- Metodologies i criteris en l'elecció dels components que han d'integrar un sistema embedded, per a una aplicació concreta; com l'elecció del format de la placa base, el processador i el sistema operatiu.
- Metodologia de creació d'imatges executables de sistema operatiu amb el constructor de plataformes de Windows Embedded CE 6.0.

L'objectiu d'aquest projecte és oferir les bases de coneixement necessàries, coma punt de partida, per a desenvolupar complexes aplicacions embedded. Per tant, aquest és un projecte obert a ser ampliat posteriorment per alumnes d'enginyeria, amb interès per la matèria.

Les possibilitats de continuació i ampliació del projecte són molt àmplies. Les meves propostes d'ampliació i millora del projecte són:

- Ampliació de funcionalitats del sistema operatiu Windows CE com connectivitat USB o Wireless LAN i suport a multisessió.
- Programació d'aplicacions mitjançant Visual Studio 2005 per al seu funcionament sobre un sistema operatiu Windows CE personalitzat. Un exemple d'aplicació podria ser la captació de senyals digitals o analògiques d'un mòdul d'E/S o la comunicació en xarxa de diversos dispositius embedded.
- Personalització de la interfície d'usuari del sistema operatiu, programació del llançament d'aplicacions durant l'arrencada i configuració del registre.
- Elaboració d'un informe exhaustiu dels PC embedded comercialitzats a l'estat espanyol, amb les característiques principals dels equips i els seus fabricants i proveïdors. Això permetria fer una actualització de l'informe de l'article "*PC embedded*", del nº 316 de la revista "Automática e Instrumentación", del Març de 2001, escrit per Julián Horrillo.

## 6 GLOSSARI

**ADC:** Un ADC (*Analog-to-Digital Converter*) o convertidor analògic-digital és un dispositiu electrònic que converteix senyals analògics en senyals digitals.

**API:** Una API (*Application Programming Interface*) és el conjunt de funcions i procediments (o mètodes, si es refereix a programació orientada a objectes) que ofereix certa biblioteca per a ser utilitzats per un altre software com una capa d'abstracció.

**ARM:** Es denomina ARM (*Advanced RISC Machines*) a una família de microprocessadors RISC de 32 bits, emprats especialment a sistemes embedded. Tenen un gran rendiment, un baix consum; i estan suportats per una gran quantitat de fabricants.

**Backplane:** Un backplane és una placa de circuit imprès que connecta diversos connectors en paral·lel, de tal manera que cada pin d'un connector estigui connectat al mateix pin relatiu de la resta de connectors, formant un bus informàtic. S'utilitza com a columna vertebral per a connectar diverses targetes, que juntes formen una computadora.

**CEPC:** Abreviació de "PC amb sistema operatiu Windows CE" que s'utilitza als manuals i el software de Microsoft Windows Embedded CE.

**DAC:** Un DAC (*Digital-to-Analog Converter*) o convertidor digital-analògic és un dispositiu electrònic que converteix senyals digitals en senyals analògics.

**Desktop PC:** El terme *desktop*, escriptori en anglès, s'empra en informàtica per a definir aquells computadors i perifèrics PC dissenyats com equips de sobretaula i orientats a aplicacions d'ús ofimàtic o domèstic.

**DLL:** Les biblioteques d'enllaç dinàmic DLL (*Dynamic Link Library*) són arxius amb codi executable que es carreguen sota demanda del programa per part del sistema operatiu.

**Embedded:** Terme anglès que significa integrat, encastat. En informàtica industrial un sistema embedded és un sistema que desenvolupa una aplicació concreta i està subordinat a un sistema superior.

**GRAF CET:** El GRAFCET (*Gràfica de Control d'Etapes de Transició*) és un diagrama funcional normalitzat, que permet fer un model del procés a automatitzar, contemplant entrades, accions a realitzar, i els processos intermedis que provoquen aquestes accions.

**LAN:** Una xarxa d'àrea local o LAN (*Local Area Network*) és la interconnexió de diversos ordinadors i perifèrics amb una extensió limitada. La seva aplicació més estesa és la interconnexió d'ordinadors personals i estacions de treball en oficines, fàbriques, etc., per a compartir recursos i intercanviar dades i aplicacions.

**Motherboard:** La *motherboard* o placa mare, anomenada també en anglès *mainboard* o *system board*, és una placa de circuit imprès PCB, que conté complexos sistemes electrònics; i s'utilitza generalment com a interfície lògica en els computadors moderns, com els PC.

**MMU:** La unitat de gestió de la memòria o MMU (*Memory Management Unit*) és un dispositiu hardware format per un grup de circuits integrats, responsable de la gestió dels accessos a la memòria per part de la unitat de processament central (CPU).

**OEM:** Abreviatura de l'anglès *Original Equipment Manufacturer*, fabricant d'equips originals. Empreses o persones que adquireixen dispositius a l'engròs per a implementar en computadores o equips de forma personalitzada que presenten amb el seu propi nom.

**OSI:** El model de referència d'Interconnexió de Sistemes Oberts OSI (*Open System Interconnection*) és un model de xarxa descriptiu per a la definició d'arquitectures d'interconnexió de sistemes de comunicacions.

**PCB:** Una placa de circuit imprès o PCB (*Printed Circuit Board*), és un mitjà per a sostenir mecànicament i connectar elèctricament components electrònics, a través de rutes o pistes de material conductor, gravats des de fulles de coure laminades sobre un substrat no conductor.

**PCI:** El PCI (*Peripheral Component Interconnect*) és un bus estàndard per a connectar dispositius perifèrics directament a la placa base.

**PC/104:** Estàndard que defineix un format de plaques electròniques, amb els seus respectius busos, utilitzat principalment a sistemes PC embedded.

**PLC:** Un PLC (*Programmable Logic Controller*) és un dispositiu electrònic utilitzat principalment en l'automatització industrial que controlen la lògica de funcionament de màquines, plantes i processos industrials.

**Rack:** Un rack és un bastidor destinat a allotjar equipament electrònic, informàtic i de comunicacions. Les seves mesures estan normalitzades perquè sigui compatible amb equipament de qualsevol fabricant.

**RISC:** L'anomenada RISC (*Reduced Instruction Set Computer*) és una arquitectura computacional de processadors que té com a principals característiques:

- Instruccions de grandària fixa i presentades en un reduït número de formats.
- Només les instruccions de càrrega i emmagatzematge accedeixen a la memòria per a dades.

L'objectiu de dissenyar màquines amb aquesta arquitectura és possibilitar la segmentació i el paral·lelisme en l'execució d'instruccions i reduir els accessos a memòria. Les màquines RISC protagonitzen la tendència actual de construcció de microprocessadors. PowerPC, DEC Alpha, MIPS i ARM són exemples d'alguns d'ells.

**RTOS:** Un sistema operatiu de temps real o *Real Time Operating System* és aquell té un comportament temporal determinista a l'hora de completar les instruccions. Pot respondre de forma previsible davant una petició d'interrupció, i ofereix una major estabilitat al sistema.

**SBC:** Un SBC (*Single Board Computer*) és un computador complet integrat en una sola targeta de circuit imprès, que incorpora el processador, la memòria, les E/S i tots els dispositius necessaris per al seu funcionament.

**System-on-chip (SOC):** Integració de tots els components d'un computador o de qualsevol sistema electrònic en un sol xip. Aquest xip conté funcions digitals, analògiques i, en molts casos, de radiofreqüència, totes integrades en el mateix circuit integrat IC (*Integrated Circuit*). La principal aplicació d'aquesta tecnologia és en el camp dels sistemes embedded.

**Target:** S'anomena *target* o dispositiu embedded aquell compost per una placa base amb un hardware determinat, destinat a realitzar unes aplicacions concretes.

**Workstation:** S'anomena *workstation* o estació de treball al computador emprat per al desenvolupament d'aplicacions. En el desenvolupament de sistemes embedded, l'estació de desenvolupament o *development workstation* s'utilitza per a la creació d'imatges personalitzades de SO o d'aplicacions, i a la transferència d'aquestes als dispositius *target*.

## **7 BIBLIOGRAFIA**

### **7.1- Llibres**

- [1] “*Real-Time Concepts for Embedded Systems*”, Qing Li, ed. CMP Books, 2006.
- [2] “*Embedded hardware, know it all*”, J. Labrosse, ed. Newnes, 2008.
- [3] “*Embedded software, know it all*”, Jack Ganssle, ed. Newnes, 2008.
- [4] “*Serial port complete: programming and circuits for RS-232 and RS-485 links and networks*”, Jan Axelson, ed. Illustrated, 1998.
- [5] “*Instrumentación virtual: Adquisición, procesado y análisis de señales*”, Antoni Mànuel, Edicions UPC, 2002.
- [6] “*Designing Embedded Hardware*”, John Catsoulis, ed. O’Reilly, 2003.
- [7] “*Introduction to Real-Time Operating Systems*”, Robert Betz, 2001.
- [8] “*Professional Microsoft Windows Embedded CE 6.0*”, Samuel Phung, ed. Wrox, 2009.
- [9] “*Windows Embedded Ce 6.0 Fundamentals*”, Stanislav Pavlov i Pavel Belevsky, Microsoft, 2008.

### **7.2- Revistes i catàlegs**

- [1] Revista **Automática e Instrumentación** nº 316 (Març 2001). Article “*PC embedded*”, Julián Horrillo.
- [2] Catàleg Arrow Ibèrica 2009.

### 7.3- Pàgines web

- [1] Embedded System Design. Comunitat de desenvolupament embedded.  
<http://www.embedded.com/> .
  
- [2] Embedded Computing. Portal amb notícies, productes i eines de desenvolupament de sistemes embedded.  
<http://www.embedded-computing.com/> .
  
- [3] Embedded devices. Enginyeria dedicada al disseny de sistemes embedded.  
<http://www.embeddeddevices.com.au/> .
  
- [4] Consorci PC/104. Informació sobre les especificacions PC/104 i llistat de fabricants.  
<http://www.pc104.org/> .
  
- [5] The Infrared Data Association. Consorci de comunicacions IrDA amb informació sobre els estàndards, comunitat de desenvolupament i fabricants.  
<http://www.irda.org/> .
  
- [6] Ampro Computers. Empresa líder en la fabricació de SBC i productes embedded.  
<http://www.ampro.com/> .
  
- [7] Diamond Systems Corporation. Empresa proveïdora de SBC, especialment dels estàndards PC/104.  
<http://www.diamondsystems.com/> .
  
- [8] ASUSTeC Computer Inc. Fabricant mundial de components electrònics i informàtics, especialment plaques base i targetes gràfiques per a PC.  
<http://www.asus.com/> .



- [9] iEi Technology Corp. Inc. Fabricant mundial de components informàtics industrials.  
<http://www.ieiworld.com/> .
  
- [10] Microsoft Windows Embedded. Secció de la pàgina web de Microsoft dedicada als productes Windows Embedded.  
<http://www.microsoft.com/windowseembedded/> .
  
- [11] Linux Online! . Portal del popular sistema operatiu amb documentació, descàrregues, distribuïdors, etc.  
<http://www.linux.org/> .
  
- [12] El Rincón de Linux para hispanohablantes. Tal com el seu nom indica, un portal amb tota la informació i suports de Linux en espanyol.  
<http://www.linux-es.org/> .
  
- [13] Monta Vista Linux. Pàgina del distribuïdor d'una de les versions *real-time* de Linux.  
<http://www.hardhatlinux.com/> .
  
- [14] Wind River. Espai web de l'empresa productora de software, amb el RTOS VxWorks com a producte estrella.  
<http://www.windriver.com/> .
  
- [15] QNX. Pàgina web de la competència directa de Windriver en RTOS.  
[www.qnx.com](http://www.qnx.com) .
  
- [16] MSDN. Portal de suport de Microsoft. En aquesta web es pot obtenir informació sobre el software de Microsoft, com Windows Embedded CE.  
<http://msdn.microsoft.com/> .
  
- [17] Datasheet Catalog. Portal buscador de documentació tècnica en format pdf.  
<http://www.datasheetcatalog.org/> .

- [18] Google. El buscador web més popular del món.  
<http://www.google.es/> .
- [19] Wikipedia. Enciclopèdia virtual, gratuïta i lliure d'Internet.  
<http://www.wikipedia.org/> .



