

Metti un pinguino nel motore...

di Giuseppe Zanetti

"Linux is not portable (uses 386 task switching etc.), and it probably never will support any thing other than AT-hard disk, as that's all I have." - Linus Torvalds, 25 Agosto 1991.

Solo il 2.5% dei circa 4.8 miliardi di microprocessori prodotti nel 1998 è stata utilizzata all'interno di un personal computer. La maggior parte di essi è stata invece assorbita dal mercato dei cosiddetti sistemi "embedded" (traducibile in italiano come "incorporato"), termine generico che comprende sia le applicazioni di tipo industriale che l'utilizzo dell'informatica all'interno di oggetti come videoregistratori, ricevitori satellitari, telefoni cellulari, impianti di allarme, centralini telefonici, stampanti, apparati di networking, ... Per rendersi conto di quanto importante sia tale fascia di mercato, si pensi che una moderna automobile contiene almeno 20 diversi chip, mentre in una abitazione se ne possono trovare almeno il doppio, disseminati fra i diversi elettrodomestici o oggetti d'uso comune.

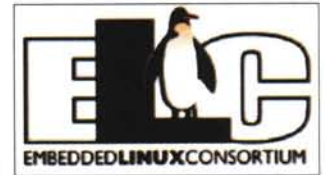
Si tratta quasi sempre di "appliances", ovvero di oggetti che esteriormente non rivelano l'aspetto tipico di un computer, pur basandosi sugli stessi principi di funzionamento (processori, memorie, e... software). Una cosa che differenzia nettamente tali apparecchi dai personal computer è che nella maggior parte dei casi non si tratta di strumenti "general purpose", bensì di oggetti studiati ed ottimizzati per far bene un solo compito preciso (ad esempio la visione di un programma televisivo via satellite, che si ottiene scompattando un flusso di dati in formato MPEG).

Nei casi più semplici il software di tali sistemi è costituito da un unico programma che sovrintende a tutte le funzionalità richieste, comprese quelle a basso livello (interfacciamento diretto con l'hardware, gestione delle interruzioni, ...). Tuttavia nei casi più complessi un singolo programma non è più sufficiente ed è necessario appoggiarsi ad un sistema operativo. Esistono, oltre a Linux, diversi prodotti utilizzati nei sistemi embedded: Windows CE, MSDOS, OS-9, µCOS, pSOS, EPOC, JavaOS, ... in questo articolo restringeremo il nostro punto di vista solamente alle tipologie di sistemi embedded di interesse per lo sviluppatore Linux.

Il più importante punto di riferimento per Linux embedded è <http://www.linuxdevices.com/>. Esiste anche un mensile cartaceo interamente dedicato all'argomento: si tratta di "Linux Embedded Journal", a cui è possibile abbonarsi nel sito <http://www.embedded.linuxjournal.com/>.



I vincoli in un sistema embedded



Nella realizzazione di un sistema embedded l'interfaccia utente svolge un ruolo particolarmente importante e deve essere la più semplice ed intuitiva possibile, in quanto l'oggetto deve essere utilizzabile anche da persone senza particolari competenze informatiche.

Al contrario che nei PC, non è detto che in un sistema embedded siano disponibili periferiche come tastiera, mouse o monitor, ma magari possono esservi solamente qualche bottone per l'input e dei LED o un display per segnalare all'utente l'esito delle operazioni. In alcuni casi le periferiche di input/output possono essere di tipo molto diverso rispetto a quelle a cui si è abituati nel mondo dei PC: in un centralino, ad esempio, l'input avviene mediante la tastiera DTMF del telefono e l'output spesso sotto forma di segnali sonori o vocali.

Una delle richieste tipiche di un sistema embedded è quella di avere una elevata affidabilità. E' improponibile che un centralino telefonico lasci a piedi una azienda o che si debba resettare il ricevitore del satellite nel bel mezzo della finale di coppa dei campioni.

La necessità di affidabilità implica alcune scelte progettuali importanti, come quella di non basare la memoria di massa su un hard disk, soggetto a possibili rotture, bensì su una memoria a stato solido di tipo Flash EPROM (Flash Disk, DiskOnChip o Compact Flash). Esse sono molto costose se paragonate ad un normale hard disk, e ciò porta alla necessità di utilizzarle con una certa parsimonia, allo scopo di tenere bassi i costi di produzione, a maggior ragione per prodotti destinati al mercato consumer.

Nonostante si tratti di memorie scrivibili, di solito vengono usate per quasi tutto il tempo a sola lettura, in modo da evitare che il filesystem si possa rovinare. Inoltre permettono solamente un numero limitato, anche se abbastanza elevato, di operazioni di scrittura.

Poiché Linux ha la necessità di scrivere in certi file o directory, ad esempio /var o /tmp, generalmente questi risiedono su un filesystem residente su ramdisk (un disco simulato usando la memoria RAM, che viene perso ad ogni reboot o spegnimento). Esso viene creato "al volo" e popolato dei file necessari mediante uno degli script avviati al boot da Linux.

Ovviamente è necessario prevedere il salvataggio su flash delle impostazioni, ad esempio rimontando la flash in lettura/scrittura per il tempo necessario all'operazione.

Spesso i sistemi embedded sono basati su versioni semplificate dei processori, privi ad esempio di coprocessore matematico - che viene comunque tranquillamente simulato nel kernel - o della MMU (Memory Management Unit). Quest'ultima componente è particolarmente importante, in quanto fornisce funzioni fondamentali per un sistema operativo multitasking, come la protezione della memoria per evitare che un processo vada a scrivere nella memoria di un altro processo e la possibilità di usare la memoria virtuale. Esiste una versione di Linux, uClinux, in grado di funzionare, seppur con molte limitazioni, senza MMU.

Nel Geode, un processore compatibile con l'architettura x86 molto utilizzato in applicazioni di tipo "set top box", il basso costo è ottenuto a scapito della cache di primo livello (soli 16 KB) e rinunciando alla cache L2.

Una delle motivazioni per cui i processori per il mercato embedded sono più limitati rispetto a quelli pensati per il mondo desktop è la necessità di risparmio energetico, sia perché molti di questi apparecchi funzionano a batterie, sia per non dover essere costretti ad inserire rumorosi ed ingombranti sistemi di raffreddamento (che a loro volta presentano consumi non trascurabili). Alcuni processori attuali per il mondo embedded (in particolare per il mercato dei PDA) permettono di variare la frequenza di clock ed il voltaggio di alimentazione in funzione del carico di lavoro della macchina: la potenza consumata da un processore in tecnologia CMOS dipende infatti da questi due parametri secondo la formula $P=CV^2f$. Esempi di queste tecnologie sono SpeedStep di Intel e LongRun, implementata dai processori Transmeta Crusoe, che permette di far consumare un solo watt ad una macchina a 500 MHz.

Un'altra necessità importante che si trova spesso nei sistemi embedded è quella del "real-time", ovvero la capacità di rispondere ad un evento entro un determinato periodo di tempo. Un sistema real-time è giustificato in tutti quei casi in cui esistono delle temporizzazioni critiche, ad esempio nel campo della telefonia.

Un esempio molto interessante è stato presentato lo scorso anno ad una fiera dedicata a Linux: immaginate una barra di otto LED che scorra da destra a sinistra e viceversa come un metronomo. Attivando in modo opportuno uno o più di tali LED è possibile, sfruttando la persistenza delle immagini sulla retina, visualizzare delle scritte. Nell'esperimento, realizzato con una versione standard (non real-time) del kernel di Linux, tutto funzionava a dovere fino a quanto la macchina compiva solamente quell'operazione. Lanciando un comando che causasse accessi al disco, ad esempio un "ls", il sistema era costretto a servire anche gli "interrupt" inviati dal controller IDE, col risultato che le scritte divenivano illeggibili. Utilizzando una



Interfaccia grafica di Qt embedded. (da inserire vicino a paragrafo "L'interfaccia grafica")



Linux Embedded Journal, una rivista dedicata allo sviluppatore Linux in ambiente embedded.

i fabbricanti di microprocessori forniscono delle "piattaforme di riferimento", ovvero delle specifiche su come costruire l'hardware basato sulle loro macchine. Quasi sempre esistono anche delle "development platform", che possono essere usate per una prima fase di sviluppo. Si tratta di schede basate sul processore che si utilizza nel progetto e contenenti i principali chipset che vengono utilizzati a corredo per fornire funzioni ausiliarie (interfacciamento al bus PCI, controller per la Ethernet o per il display LDC o SVGA, ...).

I vincoli progettuali appena descritti sono significativi e spesso parecchio pesanti per chi deve implementare un sistema embedded.

La nuova generazione di "appliances" "Internet ready"

Il profilo hardware di un tipico sistema embedded è molto limitato, tuttavia anche in questo campo si fanno sentire gli effetti della legge di Moore: di pari passo con le crescenti esigenze dell'utenza e all'ingresso prepotente dell'informatica in moltissimi apparecchi di uso quotidiano si sta assistendo alla nascita di apparecchiature con richieste hardware sempre maggiori, dotate di display LCD di qualità a colori e di memorie di massa anche di notevoli dimensioni. Cosa d'altronde ovvia, dato che si stanno trasferendo su sistemi dedicati opera-

versione real-time di Linux, esse restavano invece visibili anche in condizioni di carico elevato della macchina.

La realizzazione del software per un sistema embedded è complicata dal fatto che spesso si sviluppano contemporaneamente sia il software che l'hardware. Trattandosi di hardware costruito appositamente, è spesso necessario sviluppare nuovi device driver. In molti casi ci si può trovare a sviluppare un sistema senza avere la macchina su cui farlo girare oppure a sbattere la testa contro i banchi dell'hardware che altri stanno sviluppando. Per venire incontro al primo problema,



Un "single board computer": la scheda CPU-1231, in formato PC104, della Eurotech di Udine (<http://www.eurotech.it/>) è basata su un processore MediaGX a 266 MHz, compatibile con la famiglia x86. Essa dispone di controller IDE, floppy, SVGA, PS/2, 2 USB e FastEthernet integrati, oltre che di due porte seriali ed una parallela. Monta fino a 128 MB di RAM ed è possibile aggiungere un Disk-On-Chip di capacità fino a 128 MB. Per l'alimentazione è sufficiente una singola tensione di +5 V. E' possibile espandere il sistema mediante moduli in formato PC104.

zioni relativamente pesanti per le quali oggi si usa un PC abbastanza potente, come navigare in Internet o ascoltare file MP3. L'Internet Radio Kebango è ad esempio basata su un PowerPC a 80 MHz con 8 MB di RAM ed altrettanti di memoria di massa (flash). I sistemi operativi tradizionalmente utilizzati nei sistemi embedded non sono in grado di soddisfare appieno le nuove richieste, in



uCsimm (<http://www.uclinux.org/ucsimm/>) è la macchina più piccola (3,5"x1") in grado di fare girare una versione di Linux. Ha il formato di una SIMM ed è basata su un microcontrollore Motorola DragonBall 68EZ328 con 2 MB di flash e 8 MB di DRAM e contiene una scheda Ethernet 10Base-T ed una porta seriale ad alta velocità, oltre che un controller per display LCD in grado di offrire una risoluzione QVGA di 320x240 pixel.

quanto non comprendono funzionalità evolute di networking, compresa la possibilità di collegarsi ad Internet, o una gestione accurata della parte grafica. Occorrono pertanto sistemi operativi completi e moderni e Linux è una possibile soluzione, anche se non certamente l'unica: in questa fascia molto appetibile di mercato esiste infatti una concorrenza agguerrita e di ottimo livello (Windows CE, EPOC, ...).

Le distribuzioni di Linux embedded

Prima di vedere le principali distribuzioni di Linux per sistemi embedded, è necessario capire come avviene la fase di sviluppo per tali sistemi. Abbiamo già visto che, salvo casi molto particolari, generalmente si tratta di sistemi con un "footprint" molto limitato e dotati di memorie di massa piccole e a sola lettura. In tale situazione è normale pensare di distinguere fra la macchina "di sviluppo", dove si compilano i programmi e si prepara il sistema, dal prototipo che si sta sviluppando. Spesso la macchina di sviluppo è basata su una architettura diversa da quella di destinazione e la compilazione dei programmi avviene usando un "cross compilatore", ovvero un compilatore che genera codice per un'altra piattaforma.

La fase di sviluppo di solito porta come risultato ad un'immagine del filesystem del sistema finale, sotto forma di un file che deve essere successivamente scritto nella memoria a stato solido del prototipo. Tale operazione dipende dal tipo di hardware che si sta utilizzando: nel caso delle Compact Flash su bus PCMCIA è sufficiente usare un PC portatile per scriverle, mentre nel caso delle memorie DiskOnChip si deve usare un apposito adattatore. Esistono altri modi, come ad esempio quello di caricare il software via rete.

In ogni caso si tratta di operazioni più complicate e laboriose rispetto ad installare Linux su un PC. Lo scopo delle "distribuzioni embedded" è quello di facilitare tali operazioni e di fornire un ambiente testato e completo dei principali driver, lasciando che lo sviluppatore si concentri solamente sullo sviluppo dell'applicazione e dei driver specifici per il proprio hardware.

Esistono diverse distribuzioni di Linux embedded, che si differenziano nettamente per il modo di gestire e mantenere i "packages", per le piattaforme e funzionalità supportate, il tipo di interfaccia grafica usata, il supporto o meno del real-time, il tipo di licenza (commerciale o GPL), ...

Per dovere di bandiera, ricordiamo solamente ET Linux, realizzata in Italia nei Prosa Labs. Potete trovare la lista completa su <http://www.linuxdevices.com/articles/AT2760742655.html>.

Linux nei sistemi embedded

Come ho già sottolineato altre volte, fra le caratteristiche più interessanti di Linux vi sono l'estrema robustezza, la portabilità fra diverse piattaforme e la scalabilità. Al contrario di quanto affermato da Linus Torvalds dieci anni or sono, Linux è ora disponibile su diversi processori e piattaforme, fra cui alcune particolarmente adatte al mondo embedded, come PowerPC, ARM, StrongARM, MIPS, 68000 e persino per alcuni modelli di microcontroller. Una panoramica sulle famiglie di processori supportati da Linux si trova all'indirizzo, <http://linux-embedded.com/linuxports.php3>.

Linux, rispetto ad altri sistemi operativi tradizionalmente usati nell'embedded, offre caratteristiche ineguagliabili, ad esempio nel networking, ed ha il vantaggio di disporre di una notevole quantità di software e tool di sviluppo e di essere molto documentato e dotato di un know-how diffuso e facilmente accessibile. Ultimo aspetto, ma non meno importante, l'utilizzo di software libero permette di abbassare notevolmente i costi, in quanto non è necessario pagare delle royalty per ogni esemplare venduto. Nel caso di produzione su scala industriale, si tratta spesso di cifre non indifferenti.

Il kernel di Linux, specialmente la versione 2.4, fornisce alcune caratteristiche molto utili per la realizzazione di un sistema embedded. Vediamo le principali:

- small footprint** grazie alla modularità del kernel, è possibile creare delle versioni di Linux sufficientemente piccole da essere usabili in sistemi con risorse molto limitate
- CRAMFS** si tratta di un filesystem residente in memoria RAM (ramdisk), il quale, pur mantenendo i dati in forma compressa, permette l'accesso ai file in modalità mmap (che consente di tenere "swappata" su disco parte delle librerie shared). Non essendo basato su una memoria di massa, non è necessario prevedere il buffering del filesystem, con un aumento ulteriore delle prestazioni ed un'economia nell'uso della RAM
- initrd** si tratta della possibilità di fare un disco di boot di Linux che automaticamente crea "al volo" una ramdisk da usare come filesystem di root a partire da una immagine contenuta nel disco stesso
- Simple RAM Filesystem** ha il funzionamento simile a quello di una ramdisk, con la differenza che non viene simulato un device (per cui è necessario definire a priori una dimensione), bensì direttamente un filesystem, la cui dimensione può crescere a piacere a seconda dell'uso. Ciò consente di evitare sia sprechi sia il rischio di avere un filesystem troppo piccolo. Anche in questo caso non si ricorre al buffering
- ROMFS** Si tratta di un filesystem a sola lettura progettato per l'utilizzo in memorie ROM e per ciò molto semplificato e compatto
- DevFS** col kernel 2.4 non è più necessario creare nella directory /dev tutti i file speciali necessari ad accedere alle periferiche hardware, ma può essere in alternativa utilizzato un filesystem in cui il kernel crea automaticamente solo quelli relativi all'hardware effet

Memory Technology Devices (MTD)

Journaling Flash Filesystem (JFFS)

tivamente presente sulla macchina supporto delle memorie flash direttamente nel kernel. Esso è composto da una parte comune e da una serie di device driver specifici per i diversi tipi di hardware si tratta di un filesystem di tipo journaling ottimizzato per memorie flash. Esso consente di montare una memoria in lettura/scrittura in modo senza pericolo di perdita dei dati in caso di crash o spegnimento. Funziona sul kernel 2.0. Si veda <http://developer.axis.com/software/jffs/>.

Poiché un kernel piccolo da solo non è sufficiente, sono state scritte anche delle versioni minimali dei programmi maggiormente utilizzati. Ad esempio il pacchetto busybox fornisce in un unico eseguibile molto compatto la maggior parte delle utility di base tipiche di un sistema UNIX come ls, cp, mv, sed, tr, ...

Esistono degli accorgimenti da parte del programmatore utili per contenere le dimensioni di un programma, come l'aver cura di eliminare dal codice eseguibile le informazioni relative ai simboli utilizzati per il debugging (comando "strip") e di eseguire una compilazione ottimizzata per dimensione invece che per velocità.

Linux Real Time

Esistono diversi progetti atti ad aggiungere a Linux le funzionalità tipiche di un sistema real-time. La lista delle principali distribuzioni di Linux che supportano le funzionalità di real-time può essere reperita su <http://www.linuxdevices.com/articles/AT8073314981.html>.

La difficoltà maggiore consiste nel fatto che certe operazioni interne al kernel non possono essere interrotte per rispondere a richieste esterne, che devono perciò essere messe in attesa del proprio turno. Così facendo il vincolo del real-time non può essere rispettato.

Per superare questo limite sono possibili diversi metodi: il primo consiste nel modificare il kernel in modo da renderlo completamente interrompibile o nel fare in modo che comunque le diverse operazioni vengano compiute entro un tempo stabilito. E' logico che non si tratta di una soluzione banale, in quanto il software che funziona a livello di kernel comprende molte funzionalità, fra cui i device driver. Un secondo approccio consiste nel fare girare il normale kernel di Linux sopra un "superkernel" real-time più semplice, come se si trattasse di un comune processo. In questo modo viene diminuita la complessità del problema, in quanto non è più necessario andare a modificare tutto il kernel di Linux. Questo metodo è usato ad esempio da RTLinux (<http://www.rtlinux.org/>).

L'interfaccia grafica

Tralasciando certe esagerazioni (frigoriferi o forni che dovrebbero servire per navigare in Internet invece che per sfor-

Le piattaforme Hardware

Nonostante sia la più conosciuta, solamente una piccola parte delle CPU attualmente presenti sul mercato è basata su una architettura di tipo x86. La maggior parte dell'hardware è invece costruita usando altre piattaforme come StrongARM, ARM, MIPS, ...

Col termine architettura, ad esempio "x86", non si indica una particolare macchina (l'Intel Pentium III), bensì una famiglia di processori, basati su specifiche comuni oppure su un unico "core" (es: SPARC, ARM) che viene dato in licenza a diversi fabbricanti che a loro volta aggiungono attorno a questo nucleo comune diverse funzionalità come la cache, il supporto per bus e per le periferiche di I/O, ... che possono essere inserite direttamente nel chip principale oppure essere fornite esternamente mediante chipset di supporto. Molto interessanti sono anche i cosiddetti system-on-chip

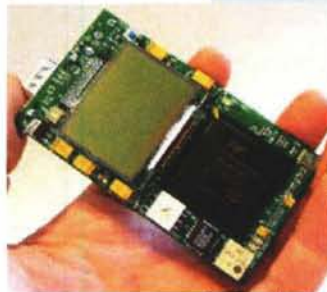
(<http://www.linuxdevices.com/articles/AT4313418436.html>), ovvero la possibilità di integrare su un singolo circuito integrato la maggior parte della logica di un calcolatore (CPU, memoria, porte di I/O, gestione dei bus esterni, ...).

Come nel software, anche nell'hardware esistono delle piattaforme più "aperte" di altre (come disponibilità di specifiche, reference platform, schemi, ...). Tale caratteristica non dovrebbe essere scordata nel caso si volesse iniziare un progetto embedded, in quanto assicura facilità di accesso alle informazioni, continuità di supporto ed indipendenza dalle scelte di un singolo produttore.

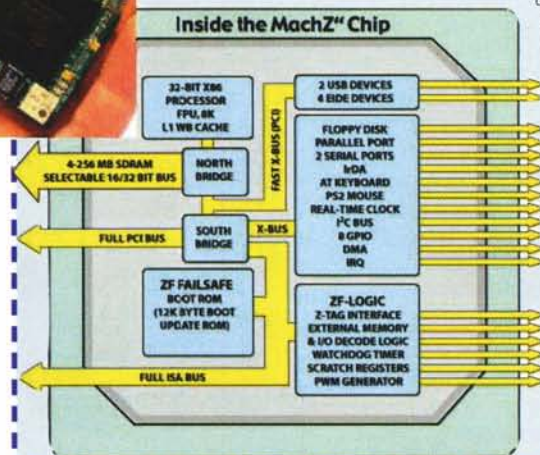
Ovviamente perché il proprio hardware sia supportato, non è sufficiente che Linux funzioni su una determinata architettura, ma occorre che vi sia anche il supporto per l'hardware ausiliario, che alle volte può differire da produttore a produttore.

Form factor

La maggior parte delle architetture supporta i più diffusi standard industriali, come i bus PCI o USB e perciò è possibile utilizzare hardware standard. Si tenga presente che il "form factor" usato nei sistemi embedded non è lo stesso dei PC desktop (AT, ATX, ...), bensì sono diffusi formati come il PC104 (<http://www.pc104.org/>) in cui la macchina è costituita da un wafer di schede hardware sovrapposte (delle dimensioni di 10.4x10.4 cm, da cui il nome). Altri formati molto usati in ambito industriale sono PC104+ (PC104 con bus PCI), CompactPCI ed EBX. Spesso inoltre i fabbricanti di "appliances" si costruiscono in casa schede custom con forme non standard.



MachZ, un esempio di system-on-chip a bassissimo consumo (238 milliwatt a 128 MHz!), basato su un core Cyrix "586", fabbricato dalla National Semiconductor per conto della ZF Linux Devices. Comprende in un singolo chip: CPU, cache memory, PC core logic; boot ROM, funzioni di I/O, "north bridge" (interfaccia per il bus PCI bus), "south bridge" (interfaccia per il bus ISA), e logica di controllo della DRAM interfaccia. Per costruire un sistema funzionante è sufficiente aggiungere solamente la DRAM. In alto a sinistra la demo board.



"north bridge" (interfaccia per il bus PCI bus), "south bridge" (interfaccia per il bus ISA), e logica di controllo della DRAM interfaccia. Per costruire un sistema funzionante è sufficiente aggiungere solamente la DRAM. In alto a sinistra la demo board.

Cool! It works with Linux!

In un solo anno il consorzio che raggruppa i maggiori sviluppatori di sistemi embedded su Linux (ELC, Embedded Linux Consortium) è passato da zero a più di 125 associati, fra i quali non mancano nomi importanti come Agilent Technologies, Hewlett-Packard, Alcatel, IBM, Lineo, QNX, ZiLOG, 3com, Motorola, National Semiconductor, Samsung, ...

Anche in Italia esistono diverse aziende che supportano e sviluppano prodotti basati su Linux embedded: Prosa (<http://www.prosa.it/>), Ascensit (<http://www.ascensit.com/>), Raider (<http://www.raider.it/>), Profuso (<http://www.profuso.com/>), 4P (<http://www.4p-online.com/>), Eurotech (<http://www.eurotech.it/>), solo per citarne alcune.

Vediamo una carrellata di sistemi embedded realizzati usando Linux (maggiori dettagli su <http://www.linuxdevices.com/articles/AT4936596231.html>). In molti casi si tratta solamente di prototipi, ma in altri di prodotti già disponibili sul mercato.

PDA e palmtop computer

Agenda VR3 - questo PDA, dotato di un display LCD da 160 x 240 pixel (2.25" x 3.25"), è basato su un processore a 32 bit NEC VR4181 a 66 MHz con 8 MB di RAM e fino a 8 MB di flash disk. Dispone di una porta standard RS232, di una seriale ad alta velocità e di interfaccia IrDA. La versione del sistema operativo utilizzata è Linux-VR.



G.Mate Yopy - fra i molti PDA funzionanti con Linux, quello di casa Samsung si prospetta come uno dei più interessanti, anche per il prezzo finale, che dovrebbe essere molto concorrenziale. Lo Yopy permette di usare le applicazioni tipiche di un PDA, come accedere ad Internet e leggere la propria e-mail, ma può essere anche usato come un registratore audio oppure per ascoltare file MP3, per vedere filmati MPEG, o come videogioco. Il cuore della macchina ruota attorno ad una CPU Intel StrongARM a 206 MHz. Offre un display a colori da 4", una porta RS232, una USB e uno slot per Compact Flash Type II.



Compaq iPAQ - pur non essendo preinstallato, è possibile utilizzare Linux sul PDA di Compaq grazie ad un progetto di porting nato sul sito handhelds.com e sponsorizzato dalla stessa casa americana. L'iPAQ è basato su una CPU StrongARM a 206 MHz con 32 MB di RAM e 16 di flash e dispone di un display LCD a colori di 240 x 320 pixel, di porta seriale sincrona e asincrona, IrDA, USB e slot PCMCIA.

4P DAT500 - si tratta di un prodotto interamente sviluppato in Italia. DAT500 è un palmtop computer adattato per applicazioni professionali di mobile

computing come point-of-sale (POS), raccolta di ordini e dati o gestione degli inventari. Essendo destinato all'utilizzo "sul campo", si tratta di un prodotto estremamente robusto ed affidabile (è ad esempio certificato per il funzionamento sicuro su aerei in volo). In aggiunta a tutte le funzioni tipiche di un palmtop (GUI, touchscreen, riconoscimento della scrittura), DAT500 dispone di alcune periferiche incorporate adatte al tipo di applicazioni per cui è pensato: lettore di codice a barre, lettore di carte magnetiche, interfaccia per la lettura/scrittura di smart-card, stampantina (termica o ad impatto). Non mancano le porte RS232 e IrDA e due slot per schede PCMCIA Type II. La CPU attualmente utilizzata è un AMD Elan SC300 (equivalente ad un 386) a 33 MHz, con 4 MB di RAM statica e 512 KB di Flash. Il sistema Linux risiede su una Flash PCMCIA ATA esterna da 8 MB. Per i primi mesi del 2001 è prevista una nuova versione basata su un system-on-chip StrongARM SA1110 a 206 MHz con 64 MB di DRAM, dotata di funzioni multimediali e possibilità di comunicazione wireless (GSM-GPRS, CDPD, CDMA). La versione di Linux installata (kernel 2.2.10) è derivata da Caldera OpenLinux, mentre come sistema grafico viene utilizzato OpenGUI (<http://www.tutok.sk/fastgl/>). Per ulteriori informazioni si veda il sito della 4P Mobile Data Processing, <http://www.4p-online.com/4p/dat500.html>.



Internet Appliances e telefonia

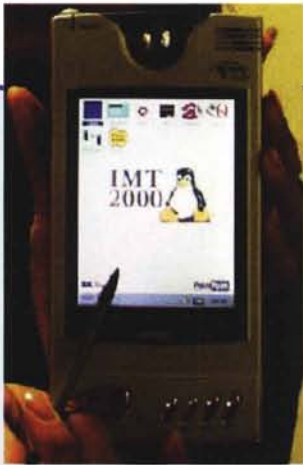
Aplio/PRO IP Phone - si tratta di una appliance basata su APLIO/Trio (un system-on-chip dotato di 4 MB di RAM e 2 MB di flash disk), che permette di effettuare chiamate telefoniche via Internet usando un protocollo voice-over-IP. Il collegamento alla rete avviene, a seconda del modello, mediante un modem interno oppure via Ethernet. Il sistema operativo utilizzato deriva dalla distribuzione uClinux. Trattandosi di un telefono, non è presente ovviamente una interfaccia grafica.



Ericsson cordless webpad/phone - il prodotto della casa svedese è un wireless webpad con integrato un telefono. Per comunicare utilizza la tecnologia senza fili via radio "bluetooth". Può essere usato sia per le funzioni tipiche di un computer come navigare sul web o spedire e-mail, sia come un telefono cordless evoluto (il tastierino DTMF viene disegnato direttamente sul display LCD). È basato su un processore Intel StrongARM e su una distribuzione embedded di Linux derivata da Red Hat. La GUI utilizzata è il Qt/Embedded GUI toolkit di Trolltech (gli stessi di KDE) e il browser è Opera.



SK Telecom IMT2000 WebPhone - due apparecchi in uno: si tratta infatti della combinazione fra un PDA in grado di collegarsi ad Internet (grazie al browser Opera) e un telefono cellulare CDMA. È



basato su una CPU StrongARM SA1110 a 206 MHz, con 32 MB di RAM ed altrettanti di memoria flash. L'output grafico avviene su un display LCD da 4" ed è presente una mini telecamera. La versione di Linux utilizzata è Tynux embedded Linux di PalmPalm, mentre la parte grafica viene gestita mediante Qt/Embedded.

Gateway Connected Touch Pad - le due particolarità di questa appliance per l'accesso ad Internet sono quella di essere il primo prodotto commerciale basato sul processore Transmeta Crusoe e di essere distribuita da America Online, il più grande Internet Provider del mondo. Possiede un ampio display touch-screen a colori da 10.4" ed è uno dei pochi prodotti embedded basati su X Window e su Gecko come browser.

TV, audio ed entertainment

Nokia Media Terminal - questo set-top-box è in grado di aggiungere parecchie nuove funzioni alla TV di casa, ovviamente a patto che nascano dei provider che forniscano i contenuti: possibilità di accesso a trasmissioni radio e TV digitali, video-on-demand, "cached" TV, ovvero la possibilità di rivedere pezzi di un programma mentre si sta registrando il seguito. Ad esse si aggiungono le tipiche funzioni di accesso ad Internet e la possibilità di usare giochi su CD e applicazioni web-based. L'hardware, di tipo PC, è costruito attorno ad un Celeron a 366 MHz con 32 MB di RAM con hard disk da 20 GB. Il software è una versione (probabilmente non embedded) di Linux, con X Window e Mozilla.

Indrema Entertainment System (IES) - anche se l'aspetto ricorda più quello di un videoregistratore, in realtà si tratta di una console da videogiochi basata su una CPU di tipo x86 di fascia elevata, con 64 MB di RAM e 50 GB di hard disk. A scelta la macchina viene fornita con

un modem o con scheda Ethernet. Il processore video è appropriato allo scopo del prodotto, essendo basato su tecnologia NVIDIA. La versione embedded di Linux usata è DV Linux.

TiVO Personal TV - questo set-top-box, basato su un PowerPC 403GCX a 54 MHz con 16 MB di RAM, viene venduto come un "personal video recorder", ma in realtà si tratta di un vero e proprio oggetto del desiderio!

Usato come un comune videoregistratore, permette di registrare in modo digitale fino a 30 ore di programmi televisivi su un capiente hard disk, anche in modo automatico (ad esempio ad orari predeterminati quotidianamente o settimanalmente) e saltando automaticamente la pubblicità.

Quello che solo TiVO può fare in più è ad esempio vedere un programma registrato in precedenza mentre se ne sta registrando un altro, oppure rivedere alla moviola quante volte si vuole la partenza del Gran Premio mentre il resto della gara continua a venire



registrata.

Ma la vera forza di TiVO è quella di poter godere la televisione senza essere schiavi della stessa. Non è ancora la vera "TV-on-demand" ma iniziamo ad esserci abbastanza vicini. Con TiVO si può: rientrare a casa anche con mezz'ora di ritardo e non perdere neppure un secondo di film; rispondere al telefono; distrarsi; vedere <http://www.space1999.net/> mentre alla tele danno la puntata; andarsi a fare un panino; nutrire il pinguino; andare al bagno quante volte si vuole; e infine... mettersi comodi comodi sul divano senza la preoccupazione di addormentarsi.

Kerbango Internet Radio - è una radio in grado di ricevere, oltre ai normali programmi FM, anche le migliaia di emittenti di tutto il mondo che trasmettono via Internet. Il collegamento può avvenire o attraverso una linea telefonica, oppure via Ethernet. E' basato su un PowerPC Motorola a 80 MHz con 8 MB di RAM e altrettanti di flash. La versione embedded di Linux utilizzata è MontaVista Hard Hat Linux.



Empeg car audio player - permette di portare in auto un archivio di 600 ore di musica MP3 ad elevata qualità, che viene registrata su un hard disk da 2"1/2. Può essere collegato via seriale o USB ad un PC per trasferire file prelevati da Internet. E' basato su un system-on-chip StrongARM a 220 MHz, con 12 MB di RAM e 1 MB di flash disk per il boot del sistema operativo. Come opzione è disponibile la possibilità di ricevere radio FM.



PhatNoise PhatBox - ha vinto il premio come migliore prodotto al terzo Summit annuale su MP3. Permette di scaricare brani MP3 da Internet e di ascoltarli in auto collegandolo all'autoradio. E' basato su un system-on-chip Cirrus Logic EP7212 a 74 MHz.

Diamond RIO audio receiver - si tratta di un lettore di MP3 in grado di far distribuire musica con qualità da CD in qualunque angolo della casa, grazie ad un collegamento con un PC che funge da server attraverso un comune doppino telefonico. E' basato su un system-on-chip Cirrus Logic Maverick.



Adomo home information system

Lo scopo del progetto AdomoWing è quello di creare un sistema completo per applicazioni domestiche, fra cui la distribuzione di musica o l'accesso ad Internet dalle diverse stanze della casa. Il progetto ruota attorno ad un server centralizzato collegabile ad Internet ed in cui risiedono i file, ad esempio la musica in formato MP3, e ad una serie di thin client specifici per le diverse funzioni (browser, riproduttore di suoni, ...).



AdomoWing Internet thin-client - è il primo esempio di client domestico. Si tratta di un apparecchio "general purpose" (adattabile ad esempio per essere usato come riproduttore di MP3). E' basato su un system-on-chip Motorola Coldfire a 90 MHz. E' dotato di LED di segnalazione e delle connessioni esterne tipiche di un computer (uscita VGA,

porte per mouse e tastiera PS/2, IrDA, microfono e casse). Il collegamento al server domestico avviene usando un'interfaccia LAN di tipo wireless. Il software è basato sulla distribuzione embedded uClinux e su una versione ridotta di X Window (Tiny-X).

AdomoPad - il secondo componente della linea Adomo è un webpad portatile che consente l'accesso ad Internet e offre le funzioni di telefono e PDA. Grazie alla tecnologia wireless è possibile spostarsi per la casa rimanendo collegati alla rete domestica. Come per AdomoWing, anche in questo caso si tratta di un thin-client che per funzionare necessita di accedere al server.



Webpad

FrontPath ProGear - si tratta di un webpad wireless dotato di un display LCD da 10.4" e di un sistema di riconoscimento della scrittura.



Screen Media FreePad - è un webpad completo ed allo stesso tempo molto semplice da utilizzare. Esso fornisce le funzioni di collegamento ad Internet, PDA e telefonia, compresa una comoda segreteria telefonica. E' dotato

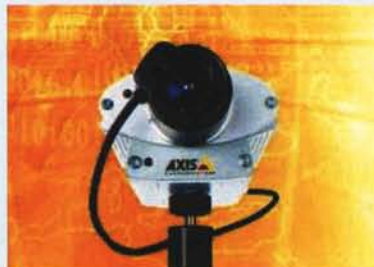
di un display LCD touch-screen da 10.4", un lettore di smart card integrato e di interfaccia USB. Il collegamento wireless alla linea telefonica avviene mediante il protocollo DECT tipico dei cordless europei. L'hardware è



basato su un MediaGX a 166 MHz con 32 MB di RAM e un flash disk da 16 MB. Il sistema grafico utilizzato è Nano-X (progetto Microwindows) ed il browser è Opera.

Altri apparecchi

Axis 2120 network camera - si tratta di una telecamera, basata su una CPU RISC a 100 MHz e sulla versione uClinux di Linux, che al posto di generare un segnale video si collega direttamente al doppino della rete Ethernet. Per realizzare una webcam o un sistema di sorveglianza remota è sufficiente fissare la telecamera al muro mediante l'apposita staffa e collegarla alla rete elettrica ed alla LAN. Non serve altro, in quanto il firmware contiene già un web server ed un software di "motion detection" in grado di rilevare la presenza di intrusi.



Sony SNT-V304 Video Network Station - è un sistema di distribuzione di immagini video via LAN basato sulla tecnologia della Axis e con funzionalità simili al prodotto precedente (la sorgente video viene in questo caso fornita da una fonte esterna).

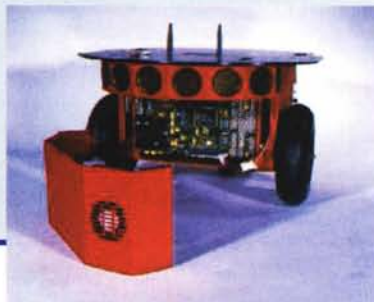
InterJak 100 Filanet intelligent Internet service appliance - si tratta di una appliance a basso costo e semplice da installare che fornisce ad una piccola LAN da ufficio funzionalità di gateway verso Internet, firewall, VPN, gestione della e-mail, file e printer sharing. La configurazione avviene da un PC mediante un browser. Il sistema usato è uClinux su un system-on-chip basato su core ARM 940T con 64 MB di RAM e 8 MB di flash. Le connessioni con l'esterno comprendono due porte USB, due Ethernet ed una interfaccia FireWire.

Nixdorf BEETLE Point-of-Sale - il POS della casa tedesca è basato su un processore MediaGX e su una versione embedded di Linux fornita da Red Hat.

IBM Linux Wrist Watch - il prototipo di computer da polso di IBM è un primo esempio di "wearable computer". Può essere usato per leggere la propria e-mail e come semplice PDA. E' costruito attorno ad un system-on-chip Cirrus EP7211 con 8 MB di RAM e 8 MB di flash ed è dotato di interfaccia IrDA e radio wireless. Dimenticavo... permette anche di sapere l'ora!



ActiveMedia Pioneer 2 Mobile Robot - si tratta di un robot da usare a scopo di ricerca e sviluppo di algoritmi per la robotica. Contiene un single-board computer basato su una CPU Pentium e su Linux.



nare dolci o "frigoriferare"), è indubbio che per molte applicazioni una buona interfaccia grafica ed un browser sono ormai necessari. I software attualmente usati in ambiente PC non si prestano molto bene per essere utilizzati all'interno di dispositivi embedded che tipicamente dispongono di soli 8-16 MB di flash disk e di 16-32 MB di RAM, in quanto troppo pesanti:

X Window System: 5 MB RAM, 16 MB disco
 GNOME: 14 MB RAM, 95 MB disco
 KDE: 11 MB RAM, 96 MB disco
 Mozilla: 12 MB RAM, 26 MB disco

Si tratta purtroppo di uno dei punti in cui maggiormente Linux ha dimostrato il proprio svantaggio rispetto ad altri concorrenti, in cui la gestione della grafica è un tutt'uno col sistema operativo. X Window infatti è un programma a parte rispetto al kernel del sistema e non è certamente un peso piuma per quanto riguarda le richieste hardware, essendo un prodotto molto complesso e basato su più strati di software. La pesantezza dell'interfaccia grafica è uno dei motivi principali che hanno causato il ritardo nell'uscita di molti prodotti embedded basati su Linux. Tuttavia nell'ultimo anno la situazione è molto cambiata, in quanto sono stati sviluppati diversi progetti di interfacce grafiche alternative e più leggere rispetto a X, che, assieme alle innovazioni inserite nel kernel 2.4, hanno riportato Linux al passo con la concorrenza, se non ad un livello superiore. Si tratta per la maggior parte di progetti sponsorizzati da grosse aziende, i cui benefici in molti casi ricadono su tutta la comunità di sviluppatori, essendo distribuiti come prodotti Open Source.

Eliminare X non è certamente semplice, anche perché i driver delle diverse schede video sono gestiti internamente ad una delle sue componenti, l'X server. Esistono tuttavia altre librerie che permettono di accedere alla grafica, ad esempio SVGAlib, ma si tratta comunque di un approccio poco efficiente, non standard e non bene integrato fra le varie applicazioni. Per risolvere questo problema, a partire dal kernel 2.2 è stato introdotto in Linux il supporto per il framebuffer, ovvero la possibilità di far gestire la grafica direttamente dal kernel, mediante device driver caricabili a run-time. Ciò offre un approccio omogeneo per l'accesso alla grafica da parte dei diversi programmi, tuttavia si tratta sempre di una interfaccia a livello troppo basso e non adatto a sviluppare applicazioni, specialmente in ambiente multitasking. Occorrono perciò ulteriori strati di software e librerie che implementino funzioni ad alto livello, come la possibilità di disegnare primitive grafiche o addirittura finestre.

Una via che sembra molto promettente è quella di fare delle versioni che non si appoggino su X delle due librerie grafiche più utilizzate, ovvero la Qt di KDE, e la GTK+ di Gnome. Questo dovrebbe permettere un porting assolutamente banale su piattaforma embedded di molto del software oggi disponibile su PC, compresi i due ambienti di desktop e i vari browser.

La descrizione dei principali progetti di interfacce grafiche "leggere" può essere reperita agli URL <http://www.linuxdevices.com/articles/AT9202043619.html> e <http://www.linuxdevices.com/files/elec-oct00/anderson/img1.html>

Il browser

Se una persona acquista un prodotto in grado di navigare in Internet, è suo diritto pretendere di poter visitare in modo ac-

Un sistema embedded fatto in casa...

Linux embedded non significa solamente "prodotti per sviluppatori": esistono diverse distribuzioni di Linux (<http://www.linux-devices.com/articles/AT2760742655.html>) specializzate per compiti particolari ed utilizzabili con un po' di impegno anche da un utente comune.

Linux Router (<http://www.linuxrouter.org/>) ad esempio permette di trasformare un PC in un router o in un firewall. FREESCO (<http://www.freesco.org/>), invece, oltre a queste funzioni permette anche di creare dei thin-server dedicati ad offrire servizi come DHCP server, web server, print server, e permette di fare un mini POP in grado di gestire fino a 2 modem in ingresso.

Queste minidistribuzioni di solito permettono di selezionare i componenti desiderati e di assemblare un floppy a sola lettura che inserito nel PC esegue il boot di un sistema Linux con le funzionalità richieste.

Vi sono poi delle minidistribuzioni come LOAF (Linux On A Floppy, <http://loaf.ecks.org>) o μ Linux (<http://sunsite.auc.dk/mulinux/>) che permettono di avere dei sistemi Linux "quasi" completi in un unico floppy e possono essere usati ad esempio per provare Linux o per trasformare un vecchio 486 con 4-8 MB di RAM e senza hard disk in un thin-client, oppure in un terminale o in un Network Computer, ad esempio usando VNC (<http://www.uk.research.att.com/vnc/>) o il client della Citrix (<http://www.citrix.com/download/>).

cettabile quasi tutti i siti. Perciò un browser per essere al passo con i tempi deve attualmente supportare almeno le seguenti caratteristiche: HTML 4, CSS, Java, Javascript, plugin per i formati più diffusi. I browser per Linux con queste caratteristiche non erano fino a poco tempo fa abbastanza leggeri o sufficientemente robusti per essere usati in progetti embedded. Tuttavia anche in questo caso le cose sono cambiate molto velocemente e vi sono alcuni progetti importanti che si prefiggono di realizzare dei browser allo stesso tempo leggeri e completi. Due prodotti promettenti sembrano essere Conqueror (il filemanager/browser incluso in KDE) e la versione per Linux di Opera. Tuttavia la strada che sembra al momento offrire maggiori prospettive è quella del progetto Mozilla (<http://www.mozilla.org/>), l'evoluzione Open Source di Netscape. In realtà la parte più interessante è il suo motore di rendering, Gecko, che può essere alleggerito dall'interfaccia grafica ed utilizzato come un prodotto a parte.

Conclusioni

L'era del personal computer sta lasciando a poco a poco spazio a quella del "post computer", in cui le funzionalità e l'intelligenza non dipenderanno più da un singolo oggetto, ma da molti apparecchi in grado di dialogare fra loro e all'occorrenza di "spostarsi" assieme l'utente. Questo grazie in particolare alle tecnologie ed alle reti di telecomunicazioni, specialmente quelle senza fili (UMTS, GPRS, bluetooth, ...).

Alcuni oggetti sono destinati a perdere la propria identità, ad esempio fra alcuni anni sarà difficile distinguere fra un televisore, un PDA o un telefono cellulare, in quanto tali funzionalità potranno essere ottenute dallo stesso apparecchio.

In questo cambiamento Linux sicuramente giocherà un ruolo fondamentale: secondo Rick Lehrbaum, cofondatore di [linuxdevices.com](http://www.linuxdevices.com), nel 2004 più del 50% del mercato del software embedded sarà detenuto da Linux. 