

ACPI ed altre tecnologie per il Power Management

Corso di Sistemi Operativi
AA 2001-2002
Prof. William Fornaciari
www.elet.polimi.it/~fornacia
fornacia@elet.polimi.it

Relazione a cura di
Francesco Cambiaso

Ver. 1.0 del 17/10/2001

Sommario

1. Introduzione.....	2
2. ACPI	3
2.1 Storia ed Obiettivi	3
2.2 - Struttura.....	5
2.3 - Requisiti di compatibilità	6
2.4 - Definizione degli stati del sistema	6
2.5 - Aree funzionali coperte - esempi	7
2.5.1 - Global System Management	7
2.5.2 - Device Management	8
2.5.3 - Performance Management	9
2.5.4 - Plug and Play - System Events	10
2.5.5 - Battery Management	10
2.5.6 - Thermal Management	10
2.6 - Specifiche hardware	12
2.7 - Specifiche software	15
3. Tecnologie di Power Management basate sull'ACPI.....	15
3.1 Instantly Available PC.....	15
3.2 SpeedStep	17
3.3 OnNow	17
4. Smart Battery System	20
5 Power Management in Windows CE.....	21
5.1 Power Management in Windows CE for Automotive.....	22
Bibliografia.....	24

1. Introduzione

Ridurre e gestire il consumo di potenza è un aspetto critico per molti sistemi elettronici. Sono stati quindi sviluppati meccanismi di Power Management (PM) per minimizzare il consumo di potenza, gestire i limiti termici del sistema e massimizzare la durata di eventuali batterie. Nel PM si considerano tradeoffs tra velocità del sistema, durata della batteria o consumo di potenza AC, potenza di calcolo disponibile, rumore ecc.

Una tendenza recente per i Personal Computers è l'uso del software per il PM. A questo scopo Intel, Microsoft ed altre aziende hanno proposto nel dicembre 1996 l' Advanced Configuration and Power Interface Specification (ACPI) (1), che definisce un'interfaccia standard per permettere ad un Sistema Operativo (OS) di comunicare con l'hardware. Si ottiene quindi quello che viene chiamato Operating System-directed Power Management (OSPM).

Le interfacce ed i concetti definiti dall'ACPI sono adatti potenzialmente per tutte le classi di computer: desktop, mobile, workstation e server. Tutte le ultime versioni di Windows (98, ME, 2000, XP) e di Linux (2) supportano l'ACPI. Windows CE, la versione dell'OS per computer palmari o dispositivi embedded in generale, ad oggi invece non supporta alcuno standard di PM: si limita a permettere agli Original Equipment Manufacturer (OEM) di effettuare una gestione ad hoc della potenza, attraverso Application Programming Interfaces (APIs) scritte per i vari componenti del sistema. Alcune versioni di Windows CE progettate per specifici campi applicativi, come ad esempio Windows CE for Automotive, adottano un modello di PM specifico, studiato tenendo conto delle particolarità e delle criticità proprie del contesto in cui verrà usato.

Altri importanti standard che in qualche modo sono collegati con l'ACPI sono l'OnNow di Microsoft, lo SpeedStep di Intel e lo Smart Battery System.

2. ACPI

2.1 Storia ed Obiettivi

L'ACPI è l'evoluzione di un precedente standard di PM, l'Advanced Power Management (APM). Il vecchio standard era basato principalmente sul BIOS, nel quale era presente del codice che veniva eseguito quando la CPU si trovava in uno stato di idle ed usava dei semplici timeouts per determinare le transizioni dei dispositivi in stati di low power. Lo svantaggio principale di questo standard nei confronti dell'ACPI è che l'OS non può sapere come l'APM si sta comportando.

L'ACPI è stato quindi progettato per raggiungere i seguenti obiettivi:

- Fare in modo che i costruttori dei sistemi possano implementare funzioni di PM con gli appropriati tradeoffs di costo/funzionalità, permettendo un'ampia gamma di soluzioni, dalle più semplici alle più sofisticate. Una larga adesione a questo standard infatti rende conveniente anche agli sviluppatori delle applicazioni includere il supporto al PM, con la conseguenza che si potranno creare nuovi usi per il PC e si potranno rendere più economici quelli già esistenti.
- Migliorare la robustezza e le funzionalità del PM. Infatti certe politiche di PM sarebbero troppo complicate per essere implementate nel BIOS ed inoltre, avendo

a disposizione a livello di OS informazioni sugli utenti, le applicazioni e l'hardware insieme, si possono adottare nuove politiche, come ad esempio gli algoritmi di Dynamic Power Management.

- Rendere appetibile per tutta l'industria, anche dal punto di vista economico, l'implementazione del PM: l'OSPM e l'ACPI ridurranno l'ammontare di investimenti ridondanti in PM, che saranno concentrati nell'OS.

Un esempio concreto di ciò che può essere fatto con l'ACPI è il poter dividere nell'OS le operazioni di I/O in "normali" e "lazy". Queste ultime, ad esempio i salvataggi in background, possono essere messe da parte ed eseguite solo quando il dispositivo di I/O richiesto è acceso per qualche altra ragione. Una politica di questo tipo richiede di conoscere quando un dispositivo è acceso, quali richieste provenienti dalle applicazioni sono "lazy" e di assicurare che non vi siano operazioni in stato di starvation.

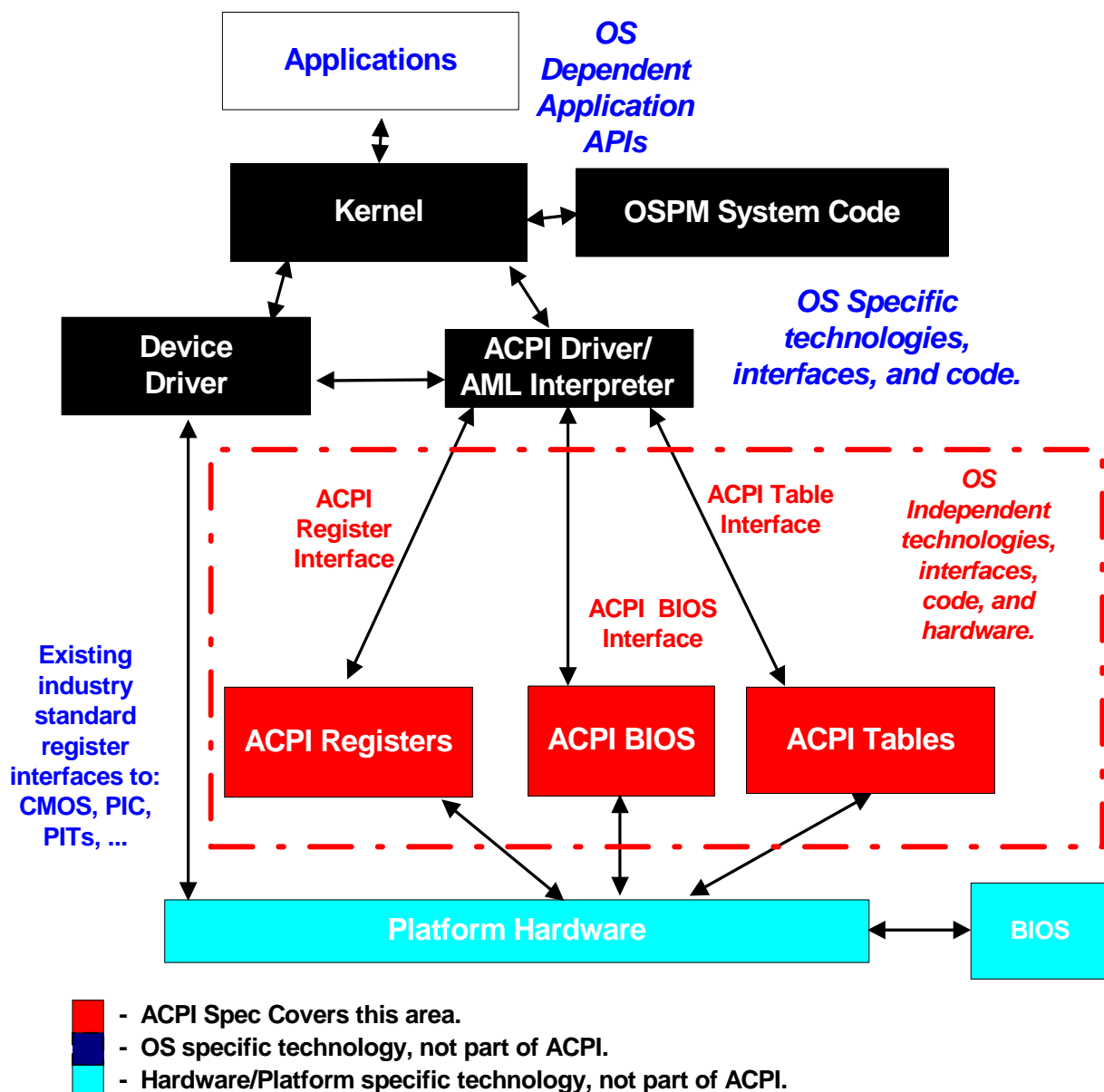
Un altro scenario può essere pensato per quelle applicazioni particolari, come ad esempio un risponditore telefonico automatico, che potrebbe richiedere all'OS il vincolo che qualsiasi stato di sleep in cui possa entrare il sistema possa permettere al sistema di svegliarsi e rispondere ad una telefonata in meno di un secondo. A questo punto, quando l'utente preme il bottone di spegnimento, il sistema sceglie lo stato di sleep più "profondo" consistente con il vincolo espresso dall'applicazione.

Naturalmente nel progetto dell'ACPI è stato tenuto in considerazione il supporto per il legacy hardware, per permettere una transizione graduale all' hardware ACPI:

Hardware\OS	Legacy OS	OSPM/ACPI OS
Legacy hardware	A legacy OS on legacy hardware does what it always did.	If the OS lacks legacy support, legacy support is completely contained within the hardware functions.
Legacy and ACPI hardware support in machine	It works just like a legacy OS on legacy hardware.	During boot, the OS tells the hardware to switch from legacy to OSPM/ACPI mode and from then on, the system has full OSPM/ACPI support.
ACPI-only hardware	There is no power management.	There is full OSPM/ACPI support.

2.2 - Struttura

La figura mostra i componenti software e hardware presenti in un sistema OSPM/ACPI. È importante comprendere che l'ACPI non è una specifica né di software né di hardware, bensì una specifica di interfaccia, che comprende sia elementi software che hardware. Solo i componenti nel riquadro tratteggiato fanno parte delle specifiche dell'ACPI. Di questi quello principale sono le ACPI System Description Tables, fornite dal Firmware, la cui funzione è quella di descrivere l'interfaccia con l'hardware: a questo scopo possono contenere anche dello pseudocodice scritto in un linguaggio chiamato ACPI Machine Language (AML), che dovrà poi essere interpretato tramite una virtual machine dall'OS. La struttura delle interfacce verrà analizzata nel dettaglio successivamente.



2.3 - Requisiti di compatibilità

Per ogni classe di piattaforma hardware (mobile, desktop ecc.) verrà richiesto un opportuno subset delle interfacce specificate dall'ACPI. Anche dal lato OS sono richiesti certi elementi per assicurare la compatibilità con l'ACPI, fra cui, ad esempio

- trovare ed usare le ACPI System Description Tables
- interpretare l'AML
- interfacciarsi con il PM Timer
- implementare gli stati relativi al consumo di potenza (Power States) nello scheduler
- fornire APIs per influenzare le politiche di PM

2.4 - Definizione degli stati del sistema

Vengono definiti degli stati globali che si applicano all'intero sistema e sono visibili per l'utente:

Global system state	Software runs	Latency	Power consumption	OS restart required	Safe to disassemble computer	Exit state electronically
G0 Working	Yes	0	Large	No	No	Yes
G1 Sleeping	No	>0, varies with sleep state	Smaller	No	No	Yes
G2/S5 Soft Off	No	Long	Very near 0	Yes	No	Yes
G3 Mechanical Off	No	Long	RTC battery	Yes	Yes	No

All'interno dello stato G1 si distinguono diversi sottostati di sleep (S1-S5), che differiscono per la latenza di wake, per l'ammontare di contesto salvato e naturalmente per il consumo di potenza.

Anche per i singoli devices vengono definiti dei Power States, che in questo caso non sono generalmente visibili per l'utente e che vengono distinti anche per il tempo necessario al driver per rendere il device pienamente funzionante (D0).

Device State	Power Consumption	Device Context Retained	Driver Restoration
D0 - Fully-On	As needed for operation	All	None
D1	D0>D1>D2>D3	>D2	<D2
D2	D0>D1>D2>D3	<D1	>D1
D3 - Off	0	None	Full initialization and load

Data la particolare importanza del processore all'interno di un sistema, anche all'interno dello stato globale G0 si distinguono diversi sottostati che differiscono per il consumo di potenza del solo processore (C0-C3). Solo nello stato C0 il processore esegue istruzioni. Le transizioni verso stati a consumo minore di potenza avverranno quando l'OS si trova in uno stato di idle e si tornerà nello stato C0 dopo avere ricevuto un interrupt dal timer dello scheduler.

Sono infine definiti i Performance States (P0-Pn), che sono sottostati di C0 per il processore e di D0 per gli altri devices. Quando un device oppure il processore si trova nello stato Px ha performance minori rispetto allo stato P(x-1) ma consumerà meno potenza. Questo può permettere di poter scalare la frequenza di funzionamento di un dispositivo in funzione delle prestazioni volute o della potenza che si è disposti a consumare. Una tecnologia che sfrutta proprio questa possibilità offerta dall'ACPI è la SpeedStep di Intel.

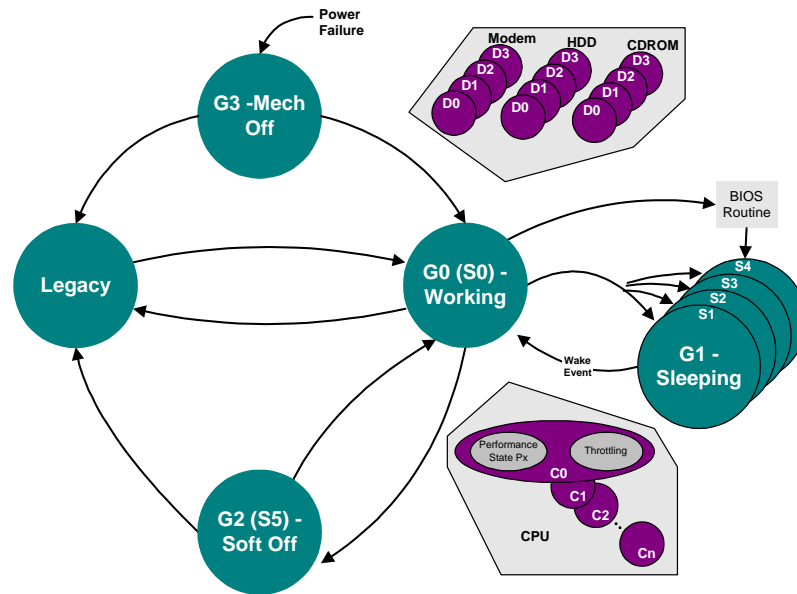
2.5 - Aree funzionali coperte - esempi

System PM, Device PM, Processor PM e Device/Processor Performance Management sono le funzioni di PM che possono essere svolte gestendo gli stati sopra esposti. Oltre a queste l'ACPI fornisce all'OS anche altre opportunità per una efficiente gestione del sistema, in particolare vengono rese più facili e robuste la gestione del Plug and Play, degli eventi di sistema, di eventuali controllori embedded, delle batterie e degli aspetti termici del sistema.

Una volta in modalità ACPI il controllo di tutti questi aspetti è completamente in mano all'OS e né il Firmware né altro software potrà manipolare la configurazione del sistema, il controllo della potenza o delle performance ecc., a meno di casi particolarmente critici nei quali, ad esempio, la latenza con la quale interverrebbe l'OS potrebbe danneggiare fisicamente il sistema.

2.5.1 - Global System Management

Per quanto riguarda il PM globale l'OS controlla le transizioni fra gli stati nel modo descritto dal seguente diagramma :



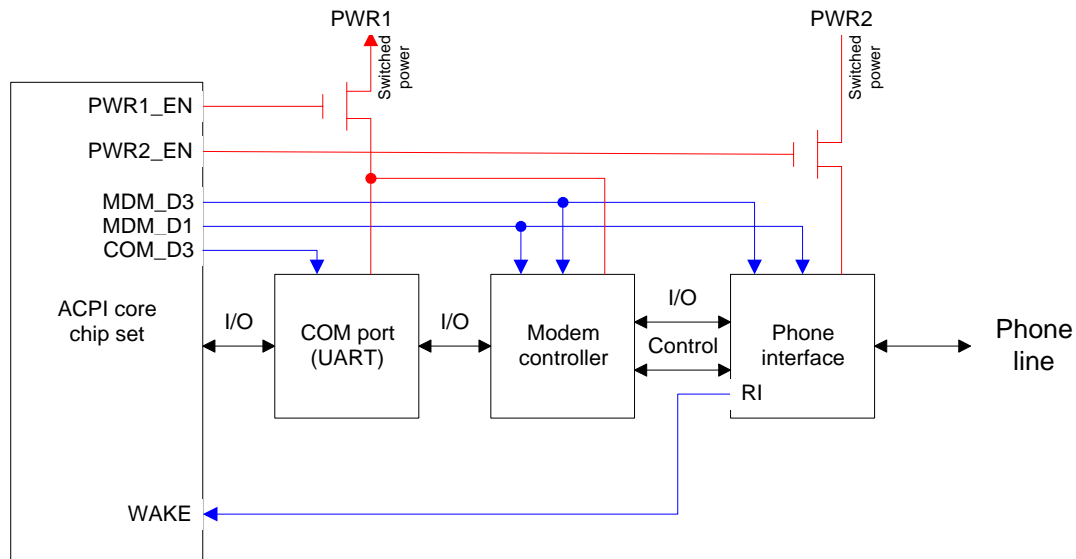
Solitamente si alterna fra stati di working e di sleeping, ed all'interno degli stati di working si “gioca” sui tradeoffs tra velocità, potenza, calore e rumore emessi dai vari devices.

All'interno di un sistema OSPM possono assumere un nuovo significato i bottoni di accensione/spegnimento: se ne potrà inserire uno che ha l'effetto di portare istantaneamente il sistema in stato G3, togliendo la tensione, ed un altro che non farà altro che mandare una richiesta al sistema, alla quale il l'OS risponderà secondo una politica che deriva dalle preferenze espresse dell'utente e dai dati delle applicazioni, spesso portando il sistema in uno stato di sleeping oppure soft-off.

2.5.2 - Device Management

Per quanto riguarda i singoli devices l'ACPI, attraverso le Description Tables, ne descrive all'OS le capacità e fornisce i controlli per settare o leggere i Power States. Inoltre fornisce uno schema generale affinché i devices possano svegliare il Computer. Dovendo l'OS mandare i comandi ai devices attraverso il bus al quale sono connessi, l'ACPI sfrutta, se esistenti, gli standard e le specifiche di PM a livello di bus. Ad oggi esistono standard per bus PCI, PCMCIA/CardBus, USB e IEEE 1394.

Il seguente esempio illustra brevemente come l'ACPI permette all'OS di controllare il modem la cui struttura hardware è schematizzata in figura.



Leggendo la entry relativa al modem nelle strutture dati dell'ACPI l'OS ne determinerà le capacità. In questo caso nella entry vi saranno le seguenti informazioni:

- Il modem supporta D0, D1 e D3
- D0 richiede potenza da PWR1 e PWR2
- D1 richiede potenza da PWR1
- Per svegliare il computer il modem non necessita di potenza
- Metodi di controllo per settare/leggere i Power States, scritti in AML
- Descrizione delle risorse del modem, principalmente registri

Una politica dell'OS potrebbe prevedere, nel caso un'applicazione metta il modem in answer mode, una transizione verso lo stato D1. Per effettuare questa transizione l'OS dovrà prima controllare che nessun altro device abbia bisogno di PWR2. Chiamerà il metodo di controllo `_OFF` associato alla risorsa PWR2 per spegnerla (nel metodo saranno contenuti i comandi per disabilitare PWR2_EN). Quindi chiamerà il metodo `_PS1` per mettere il modem nello stato D1. Questo metodo abiliterà il segnale MDM_D1, che dirà al controller del modem di entrare in modalità low-power.

Particolare attenzione deve essere posta per il PM dei bus. Rispetto agli altri device si aggiunge infatti la necessità di tenere traccia dello stato di tutti i dispositivi sul bus e di far compiere al bus stesso transizioni verso stati che siano consistenti con quelli dei devices che vi sono connessi. Ad esempio, lo stato del bus non dovrebbe essere più basso dello stato più alto fra quello dei devices che vi sono connessi, per non comprometterne il corretto funzionamento. Naturalmente vincoli di questo tipo vanno adattati alle specifiche caratteristiche dei vari bus e dei vari devices.

2.5.3 - Performance Management

In precedenza si è visto che tecnologie come SpeedStep di Intel sfruttano i Performance States del processore. Quelli che seguono sono esempi di come i Performance States possono essere sfruttati anche per altri devices :

- un hard disk oppure un controller per la RAM possono fornire diversi livelli di throughput (a cui ovviamente corrispondono diversi livelli di consumo)
- un display LCD può fornire diversi livelli di luminosità
- un sottosistema audio può fornire diversi livelli di volume o qualità del suono
- un dispositivo con interfaccia Bluetooth può variare il suo range operativo da 10 a 100 metri aumentando la potenza di trasmissione da 1 a 100 mW (si tratta in questo caso solamente di una possibile futuro scenario, in quanto ad oggi nè l'ACPI nè Windows supportano Bluetooth)

2.5.4 - Plug and Play - System Events

Durante il boot del sistema vi è la fase di enumerazione dei device da parte dell'OS, che semplicemente ne legge l'hardware ID nelle Description Tables. Ogni device enumerato dall'ACPI include metodi di controllo che forniscono informazioni riguardanti quali risorse (numeri di IRQ, indirizzi di I/O) possono essere usate dal device e quali effettivamente usa. Queste informazioni possono essere usate per rendere più robusta la gestione del Plug and Play, che veniva precedentemente gestito secondo le specifiche PNPBIOS in modo così poco affidabile da guadagnarsi il nickname di Plug and Pray.

L'ACPI definisce un modello di eventi unico usato per eventi di Plug and play, di PM e di tipo termico. E' basato su 2 registri, l'event status register, nel quale viene settato il bit corrispondente all'evento accaduto e l'event enable register, che serve per mascherare alcuni eventi ed evitare che per essi venga generato un System Control Interrupt (SCI). Il sistema risponde ad uno SCI lanciando un metodo di controllo che usa codice AML per informare l'OS di quale evento sia accaduto.

2.5.5 - Battery Management

L'ACPI permette una gestione completa delle batterie di un sistema, a condizione che la loro interfaccia sia compatibile con lo standard Smart Battery oppure che siano presenti metodi di controllo scritti in AML. In entrambi i casi l'OS avrà a disposizione informazioni circa lo stato di carica, e la notifica di eventi come l'inserimento o la rimozione o l'inizio della fase di ricarica. Nel caso vi siano più batterie presenti e l'OS può effettuare una "sintesi logica" di un'unica batteria composita, nel caso quest'operazione non sia già stata effettuata dall'interfaccia delle batterie.

Uno degli aspetti più importanti è la gestione dei "Low Battery Levels": l'utente dovrà essere avvisato quando l'energia disponibile scenderà sotto determinati livelli con opportuni warning, l'OS dovrà portare il sistema in uno stato di sleep (S1-S5) prima della scarica completa e dovrà anche prevedere una procedura di "Emergency Shutdown" che minimizzi i danni all'integrità del sistema (ad esempio corruzione di files) nel caso che il livello di carica scenda a valori molto prossimi allo zero.

2.5.6 - Thermal Management

L'ACPI gestisce gli aspetti termici di un sistema dividendolo in una o più "Thermal Zones". Per ognuna di queste, nel caso i sensori rilevino aumenti eccessivi di temperatura, l'OS potrà agire con 2 metodi di raffreddamento:

- il "Passive cooling", che mira a ridurre la temperatura riducendo il consumo di potenza di un device al costo di una perdita di performance
- l'"Active cooling", che invece agisce aumentando il consumo di potenza, ad esempio azionando una ventola di raffreddamento od aumentandone i giri al minuto.

Naturalmente è opportuno che sia l'utente ad esprimere la preferenza fra performance e risparmio energetico e/o livello di rumorosità del sistema.

Per quanto riguarda il passive cooling l'OS ha la possibilità di implementare politiche anche molto sofisticate. Ad esempio, nel caso si voglia applicare il passive cooling ad un processore, si potrà ricorrere alla seguente equazione empirica per determinare l'ammontare della variazione di performance della CPU necessario per abbassare la temperatura di una zona termica:

$$\Delta P [\%] = _TC1 * (T_n - T_{n-1}) + _TC2 * (T_n - T_t)$$

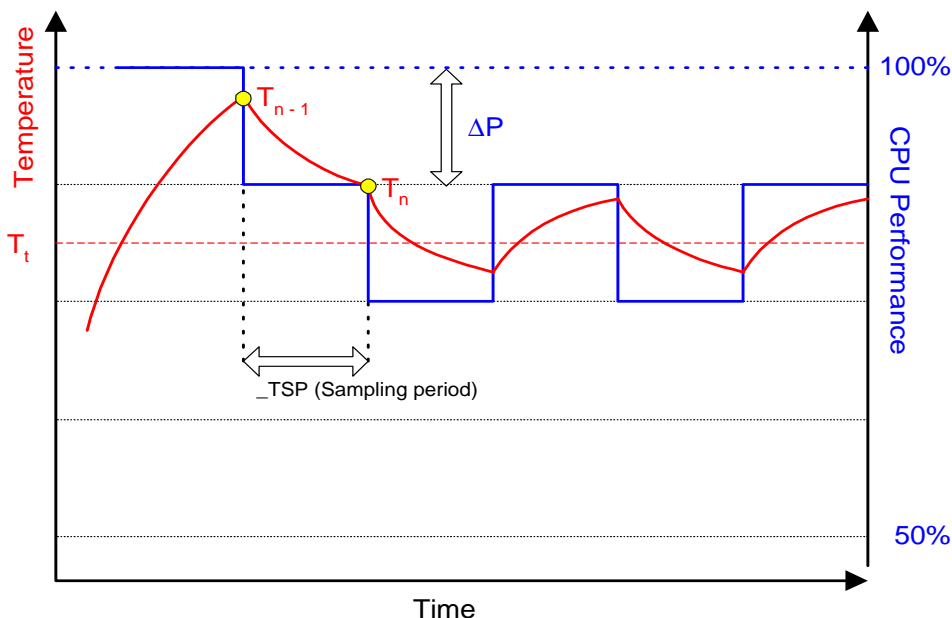
Dove:

T_n = temperatura corrente

T_t = temperatura target (che corrisponderà al $_PSV$, il valore di soglia per il passive cooling)

I due coefficienti $_TC1$ e $_TC2$ e l'intervallo di sampling della temperatura (tramite polling) $_TSP$ sono costanti dipendenti dall'hardware.

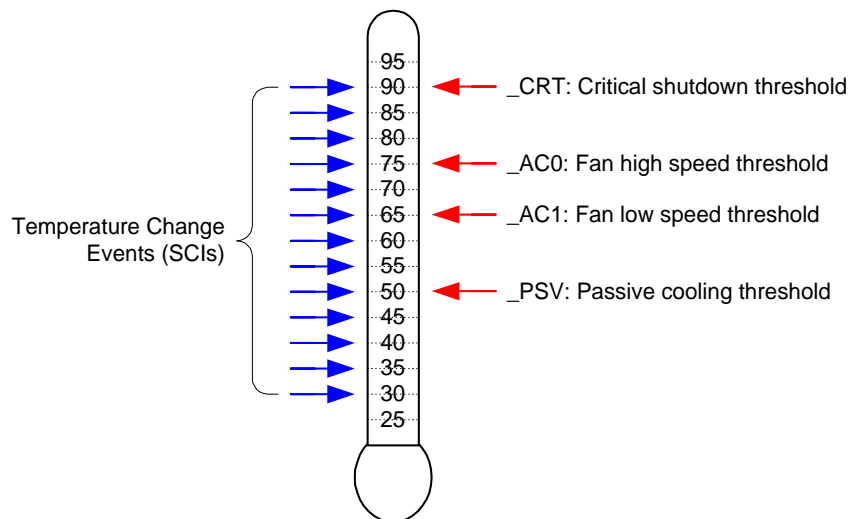
In una piattaforma compatibile con l'ACPI e nella quale si può variare la frequenza della CPU, l'applicazione di questa equazione porta ad avere un transitorio di questo tipo (la temperatura è la linea continua e la performance quella discreta):



Come nel caso del livello critico delle batterie, anche nel caso di emergenza-temperatura dovrà essere prevista una procedura di shutdown "soft".

L'OS potrà avere informazioni sullo stato termico del sistema direttamente tramite polling dei sensori (come nell'esempio visto prima) oppure tramite interrupt generati dall'hardware stesso.

Nel caso l'hardware supporti un sistema di notifica degli eventi termici basato su interrupt, quando l'OS riceverà una notifica di cambiamento di temperatura, confronterà la nuova temperatura con delle temperature di soglia e prenderà gli opportuni provvedimenti. Un esempio di una politica di questo tipo è illustrato nella figura seguente:



2.6 - Specifiche hardware

La specifica ACPI prevede per l'hardware ACPI-compatibile due tipi di modelli: Fixed e Generic.

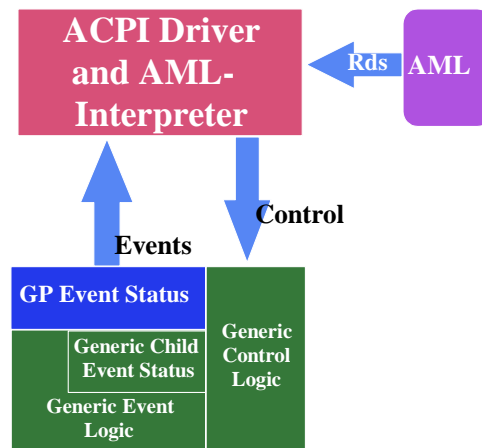
Le features di un sistema implementate tramite Fixed hardware dovranno essere manipolate tramite registri residenti all'interno di uno spazio di indirizzamento specificato dall'ACPI e dovranno avere un comportamento prefissato.

Al contrario al Generic hardware è permesso un più alto grado di flessibilità, ad esempio non vi sono limitazioni riguardo alla posizione dei registri da usare, registri a cui l'OS non accederà direttamente ma tramite chiamate a codice AML per lo specifico hardware che dovrà essere fornito dall'OEM.

Si è cercato di limitare il numero di features da implementare con Fixed hardware per minimizzare i cambiamenti richiesti per la migrazione dal legacy hardware. In pratica si è obbligati ad usare questo modello solo quando è fondamentale ridurre l'impatto che avrebbe sulle performance accedere ad un registro tramite diversi layers di driver.

Il vantaggio del modello Generic è quello di permettere agli OEMs di potersi differenziare dalla concorrenza tramite una differente implementazione dell'hardware (che potrà risiedere nello spazio di indirizzamento dell'I/O, della memoria, del PCI, di un controller embedded o del System Management Bus), di poter inserire nel loro hardware del valore aggiunto, che verrà utilizzato tramite i metodi di controllo proprietari scritti in AML. Un altro vantaggio è quindi l'indipendenza del comportamento dell'hardware dall'OS, a patto che quest'ultimo esegua correttamente il codice AML.

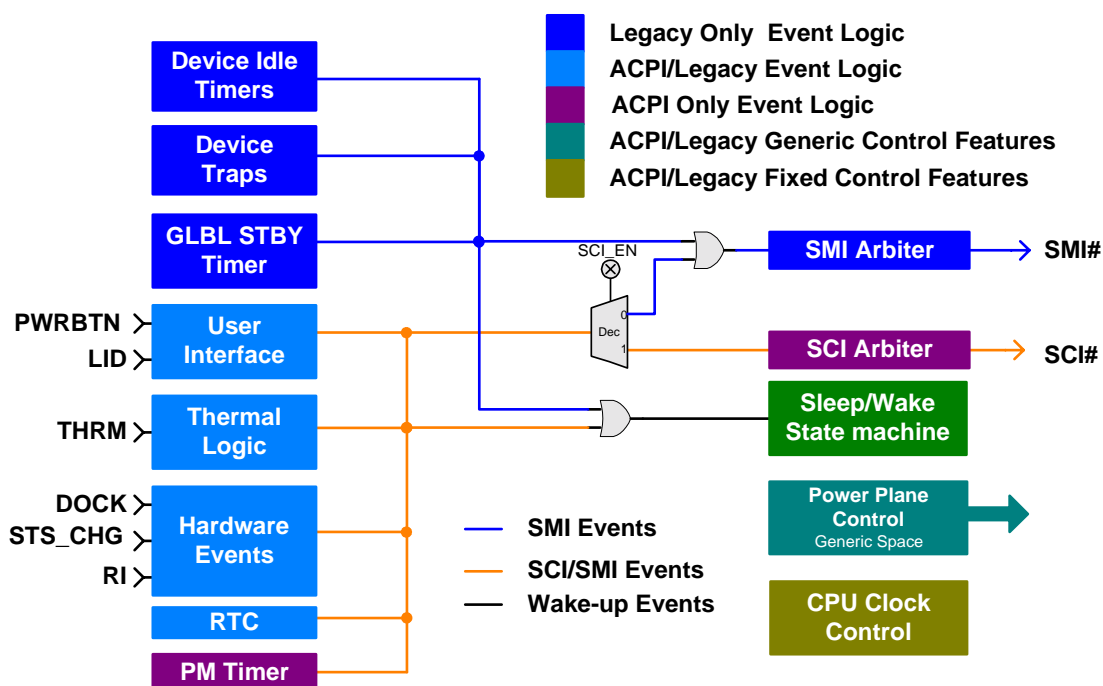
Il modello Generic può essere rappresentato dal seguente diagramma a blocchi:



Il paradigma eventi-controllo è comune anche al modello Fixed, con l'unica differenza che l'OS non risponderà chiamando un metodo di controllo AML ma direttamente tramite un driver ACPI.

L'architettura ACPI opera una distinzione logica fra eventi di tipo "hardware" e di tipo "interrupt". I primi, come ad esempio un evento di wake, causano incondizionatamente una qualche operazione, ad esempio una transizione di stato da G1 a G0. Gli eventi di tipo interrupt causano invece l'esecuzione di un interrupt handler. In entrambi i casi verrà generato un interrupt di tipo System Control Interrupt (SCI), che è visibile all'OS, al contrario di quanto accade per sistemi legacy che normalmente generano un System Management Interrupts (SMI), trasparente per l'OS, visto che in sistemi non-ACPI non deve gestire questo tipo di eventi.

Se si vogliono supportare sia ACPI OS che legacy OS bisogna prevedere un modo per rimappare gli eventi di interrupt al cambio di modalità di funzionamento, come illustrato dall'esempio riportato nel seguente diagramma in cui si possono notare, in basso a destra, tre blocchi di logica di controllo necessaria per implementare parte del modello comportamentale definito dall'ACPI:

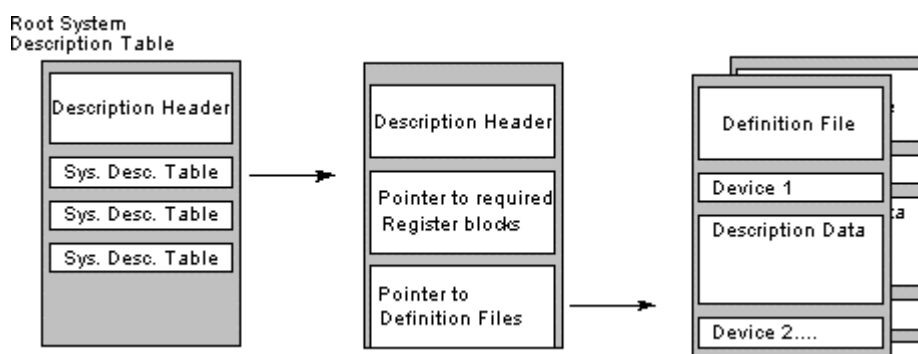


Infine la seguente tabella dà un'idea di quali features possono essere implementate ed in che modo:

Feature Name	Description	Programming Model
Power Management Timer	24-bit/32-bit free running timer.	Fixed Hardware Feature Control Logic
Power Button	User pushes button to switch the system between the working and sleeping states.	Fixed Hardware Event and Control Logic or Generic Hardware Event and Logic
Sleep Button	User pushes button to switch the system between the working and sleeping state.	Fixed Hardware Event and Control Logic or Generic Hardware Event and Logic
Power Button Override	User sequence (press the power button for 4 seconds) to turn off a hung system.	
Real Time Clock Alarm	Programmed time to wake the system.	Optional Fixed Hardware Event
Sleep/Wake Control Logic	Logic used to transition the system between the sleeping and working states.	Fixed Hardware Control and Event Logic
Embedded Controller Interface	ACPI Embedded Controller protocol and interface, as described in section 13, "ACPI Embedded Controller Interface Specification."	Generic Hardware Event Logic, must reside in the general-purpose register block
Legacy/ACPI Select	Status bit that indicates the system is using the legacy or ACPI power management model (SCI_EN).	Fixed Hardware Control Logic
Lid switch	Button used to indicate whether the system's lid is open or closed	Generic Hardware Event Feature
C1 Power State	Processor instruction to place the processor into a low-power state.	Processor ISA
C2 Power Control	Logic to place the processor into a C2 power state.	Fixed Hardware Control Logic
C3 Power Control	Logic to place the processor into a C3 power state.	Fixed Hardware Control Logic
Thermal Control	Logic to generate thermal events at specified trip points.	Generic Hardware Event and Control Logic (See description of thermal logic in section 3.9, "Battery Management.")
Device Power Management	Control logic for switching between different device power states.	Generic Hardware control logic
AC Adapter	Logic to detect the insertion and removal of the AC adapter.	Generic Hardware event logic
Docking/device insertion and removal	Logic to detect device insertion and removal events.	Generic Hardware event logic

2.7 - Specifiche software

Per dare all'OS informazioni sul sistema, come già accennato in precedenza, vengono usate le System Description Tables, memorizzate in parte nello spazio di indirizzamento della memoria del sistema, in parte nell'area dati del BIOS. La struttura dati usata per implementarle è piuttosto complessa. In pratica è una struttura ad albero nella quale si può accedere alle informazioni ed ai metodi di controllo delle varie features del sistema passando attraverso diversi livelli di puntatori:



Caricando i dati contenuti in queste tabelle l'OS costruisce una struttura gerarchica chiamata ACPI Namespace, che mantiene i riferimenti a tutti gli oggetti facenti parte dell'architettura ACPI presenti nel sistema. Essendo una struttura dinamica l'OS può aggiungervi e cancellarvi oggetti, caratteristica utile per supportare l'hot/warm/cold plug e removal, ossia l'inserimento e la rimozione di dispositivi mentre il sistema è in stato di working/sleeping/off. In un piccolo sistema l'ACPI Namespace potrebbe presentarsi in questo modo:

3. Tecnologie di Power Management basate sull'ACPI

Lo standard ACPI è stato usato da molte aziende come base sulla quale costruire avanzate tecnologie per risolvere specifici problemi di PM e di gestione del sistema. Nel seguito saranno discussi gli standard Instantly Available PC e SpeedStep di Intel e OnNow di Microsoft. Altri standard basati sull'ACPI sono stati sviluppati da costruttori di motherboards.

3.1 Instantly Available PC

La tecnologia Instantly Available PC (IAPC) (3) è stata proposta da Intel nel 1997 per risolvere fondamentalmente due problemi legati all'uso di PC di classe Desktop in ambiente Home e Office:

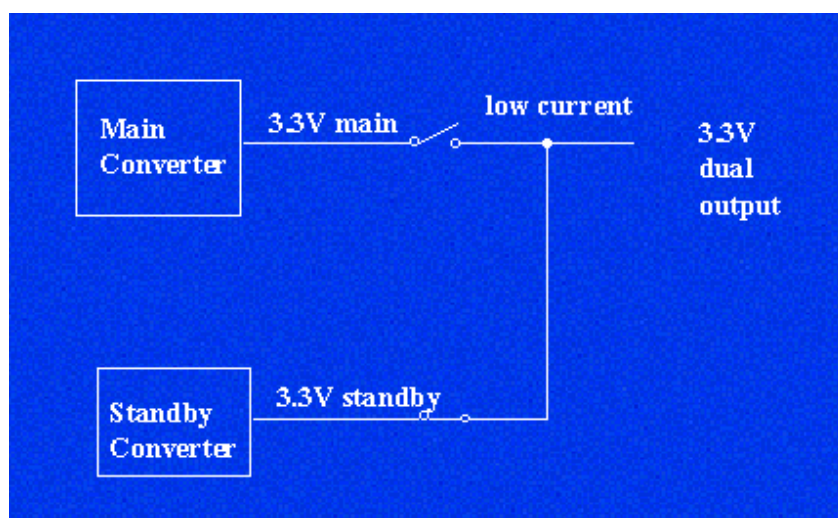
- il bisogno di rendere il PC, che è ormai diventato parte integrale della vita di tutti i giorni, più simile ad altri dispositivi di consumer electronic, come TV e telefono, in termini di disponibilità istantanea all'utilizzo. L'utente si aspetta oggi di potere accedere velocemente alle ultime notizie ed in ambito Office sono sempre più

importanti features come “wake on ring” e “wake on LAN”. E' quindi fondamentale poter ridurre il tempo di accensione del PC a valori dell'ordine di pochi secondi

- il bisogno di rendere il PC più efficiente nell'uso dell'energia senza che questo si ripercuota sulle performance, senza quindi troppi tradeoffs sgraditi agli utenti. Questa necessità si fa sempre più pressante dopo che molti governi in tutto il mondo, hanno apportato cambiamenti restrittivi alle norme riguardanti il consumo di energia. Alcuni regolamenti europei, come il Blue Angel tedesco, prevedono un consumo massimo di 5 Watt per un PC in stato di sleep, ed è proprio questo valore l'obiettivo dell'IAPC

La soluzione a questi problemi è data principalmente dallo sfruttamento dello stato S3 (Suspend to Ram) della specifica ACPI, che dovrà essere quindi implementato obbligatoriamente nel sistema. Lo si è preferito agli stati S4 (Suspend to Disk) e S5 (Soft Off) poiché questi ultimi impongono latenze di resume troppo elevate rispetto agli obiettivi visti prima (S5 verrà comunque utilizzato per ottenere il minimo consumo), mentre il tempo di resume da una DRAM è di circa 5 secondi.

Per permettere nello stato S3 un consumo minore a 5 Watt si è lavorato sulla rete di distribuzione dell'energia ai vari dispositivi del sistema, prevedendo che alcuni di questi si debbano alimentare da convertitori differenti a seconda dello stato del sistema. Questo sistema viene chiamato Dual Mode Output ed il suo funzionamento concettuale è illustrato nella seguente figura:



Il costo aggiuntivo per implementare questo sistema, per i produttori di motherboard, può essere limitato a 1-2 dollari. Il risultato finale è un partizionamento dei dispositivi del sistema a seconda della sorgente di potenza a cui sono collegati, come descritto dalla seguente tabella in cui si nota che verranno collegati a sorgenti di tipo dual solo quei dispositivi che devono essere alimentati anche nello stato di sleep.

Voltage planes	Consumers
3.3V	3.3 V PCI, 3.3 V components, processor cache, Expansion Bus Bridge
5V	processor VRM, PCI and other 5 V components on the motherboard
3.3V dual	Host bridge with memory controller, SDRAM components, memory clock driver, 3.3Vaux PCI connector auxiliary power pin
5V dual	USB devices (USB wakeup capabilities will be included in the future devices), 5V based PCI motherboard devices intended to be wakeup devices.

3.3V always present standby	This voltage is always present when the system is plugged in to the AC socket. It is intended to be connected only to the ACPI controller's resume well, the power button and power indicator (LED).
-----------------------------	--

3.2 SpeedStep

Mentre la IAPC è una tecnologia pensata esclusivamente per sistemi di tipo desktop, SpeedStep è stata implementata solo nei processori destinati a sistemi mobili.

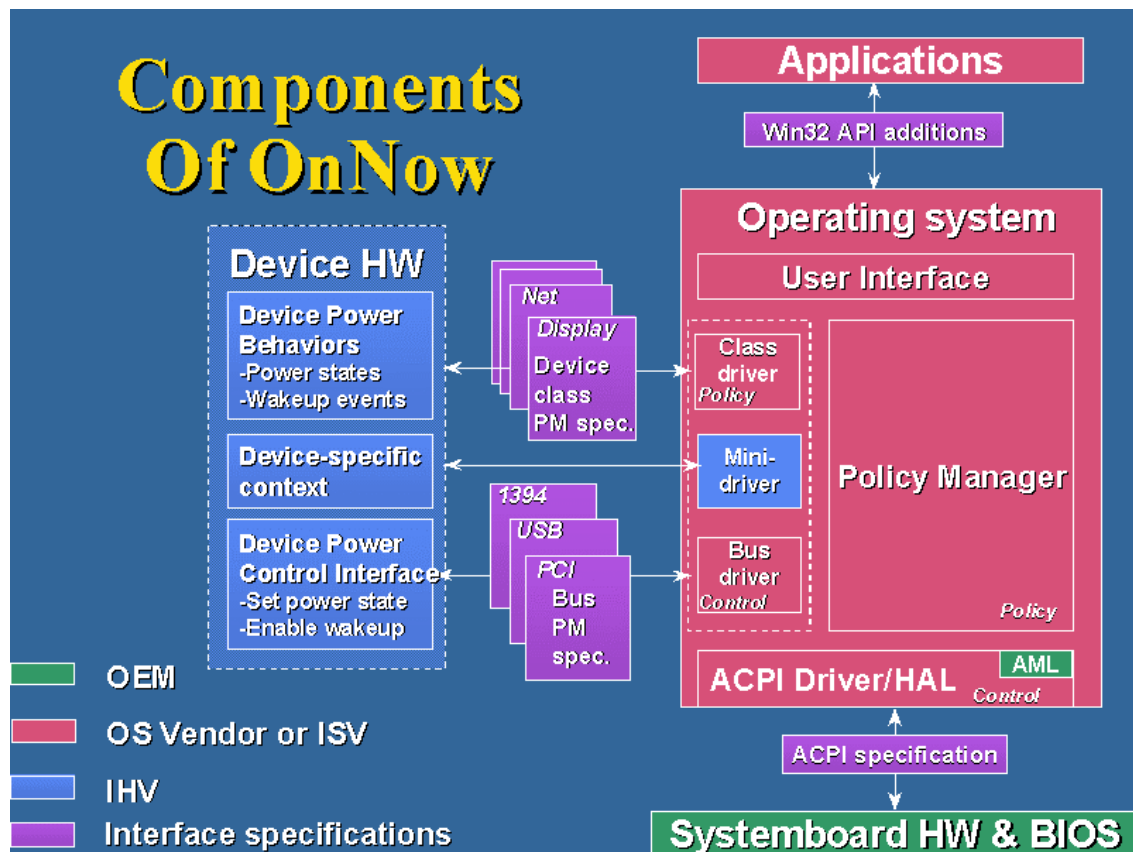
SpeedStep prevede due modalità di funzionamento, che corrispondono a due performance states dell'ACPI. Esiste una modalità normale nella quale il processore funziona alla massima frequenza, pensata per quando il sistema è connesso alla rete AC. Quando invece il sistema si alimenta dalle batterie il processore funziona in modalità chiamata appunto Battery Optimized Mode, nella quale si ottiene un risparmio di potenza abbassando la frequenza di funzionamento ed eventualmente anche la tensione. I cambiamenti di frequenza possono avvenire anche durante il normale funzionamento del processore poiché possono avvenire in meno di un millisecondo.

Specifiche tecniche più dettagliate non sono per ora state rese pubbliche.

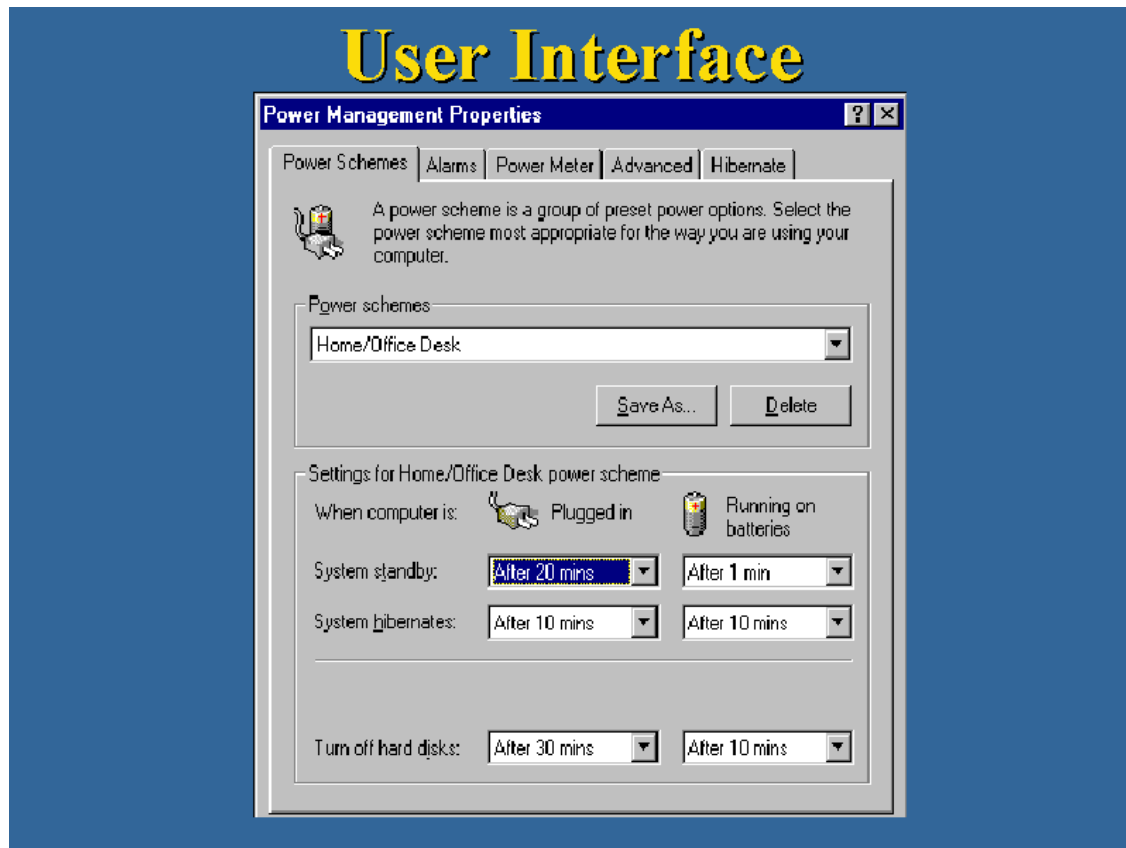
3.3 OnNow

Come accade per molti aspetti di un OS anche nel PM è opportuno separare chiaramente quelle che sono le politiche, ad esempio la decisione di portare il sistema in uno stato di sleep, da quelli che sono i meccanismi, ad esempio il meccanismo di controllo usato per effettuare la transizione di stato. L'ACPI ed altri standard di PM per specifici device o bus forniscono all'OS i meccanismi, mentre le politiche, nei sistemi Windows, vengono gestite da quella che Microsoft ha chiamato architettura OnNow (4) (5).

Nella visione Microsoft quindi l'architettura complessiva di un OSPM può essere descritta dal seguente schema:



OnNow comprende le funzioni di PM a livello di sistema e dei vari device, coerentemente con le specifiche ACPI, oltre ad un'interfaccia utente che permette l'inserimento delle preferenze degli utenti:



L'interfaccia verso le applicazioni è formata da APIs e messaggi con i quali l'OS informa le applicazioni del power state corrente e di APIs che permettono invece alle applicazioni stesse di informare l'OS riguardo alle loro necessità di risorse e requisiti di latenza. Le seguenti funzioni sono implementate in tutte le ultime versioni di Windows:

Function	Description
CallNtPowerInformation	Sets or retrieves power information.
CanUserWritePwrScheme	Determines whether the current user has sufficient privilege to write a power scheme
DeletePwrScheme	Deletes the specified power scheme.
GetActivePwrScheme	Retrieves the index of the active power scheme.
GetCurrentPowerPolicies	Retrieves the current system power policy settings.
GetDevicePowerState	Retrieves the current power state of the specified device.
GetPwrCapabilities	Retrieves information about the system power capabilities.
GetPwrDiskSpindownRange	Retrieves the disk spindown range.
GetSystemPowerStatus	Retrieves the power status of the system.
IsPwrHibernateAllowed	Determines whether the computer supports hibernation.
IsPwrSuspendAllowed	Determines whether the computer supports the sleep states.
IsSystemResumeAutomatic	Indicates the current state of the computer.
ReadGlobalPwrPolicy	Retrieves the current global power policy settings.
ReadPwrScheme	Retrieves the power policy settings that are unique to the specified power scheme.
RequestWakeupLatency	Specifies roughly how quickly the computer should enter

	the working state.
SetActivePwrScheme	Sets the active power scheme.
SetSuspendState	Sets the system suspend state.
SetSystemPowerState	Suspends the system by shutting power down.
SetThreadExecutionState	Enables applications to inform the system that it is in use.
WriteGlobalPwrPolicy	Writes global power policy settings.
WriteProcessorPwrScheme	Writes processor power policy settings for the specified power scheme.
WritePwrScheme	Writes policy settings that are unique to the power scheme.

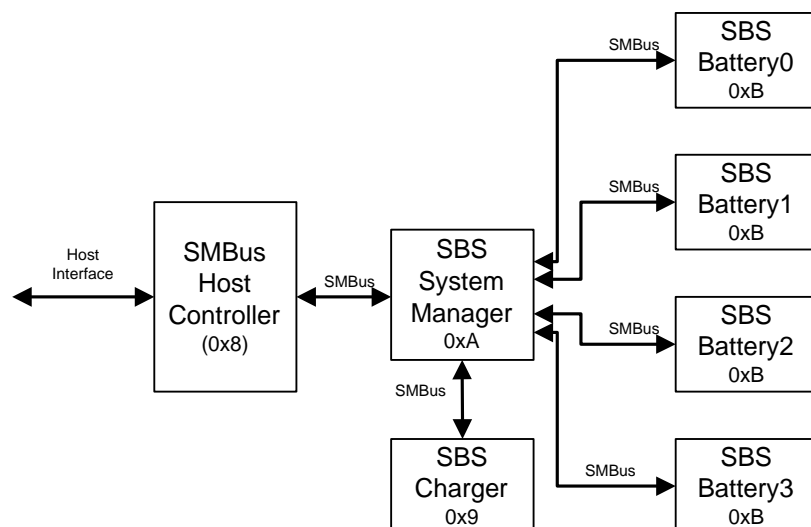
4. Smart Battery System

Lo Smart Battery System (SBS) (6) è uno standard aperto proposto nel 1996 da Intel e Duracell che definisce un protocollo per la gestione delle batterie dei sistemi portatili. Uno dei maggiori problemi per gli utenti era infatti la scarsa accuratezza delle indicazioni sullo stato di carica della batteria e le notevoli diversità che si riscontravano in questo senso tra batterie di diversi produttori. L'SBS non pone limiti sulla tecnologia di costruzione delle batterie, nè sulla loro forma o dimensione, ma cerca di unificare il comportamento della circuiteria integrata nel package della batteria che fornisce informazioni sullo stato di carica ed ottimizza le funzioni di ricarica.

Il chip della batteria fornisce informazioni ad un controller della motherboard. L'SBS definisce il bus per queste comunicazioni, il set dei dati scambiati, i comandi per il caricabatterie e per il "battery selector", nel caso nel sistema siano presenti più batterie.

L'OS ottiene informazioni sullo stato della batteria attraverso sistemi di PM proprietari oppure standardizzati come l'ACPI.

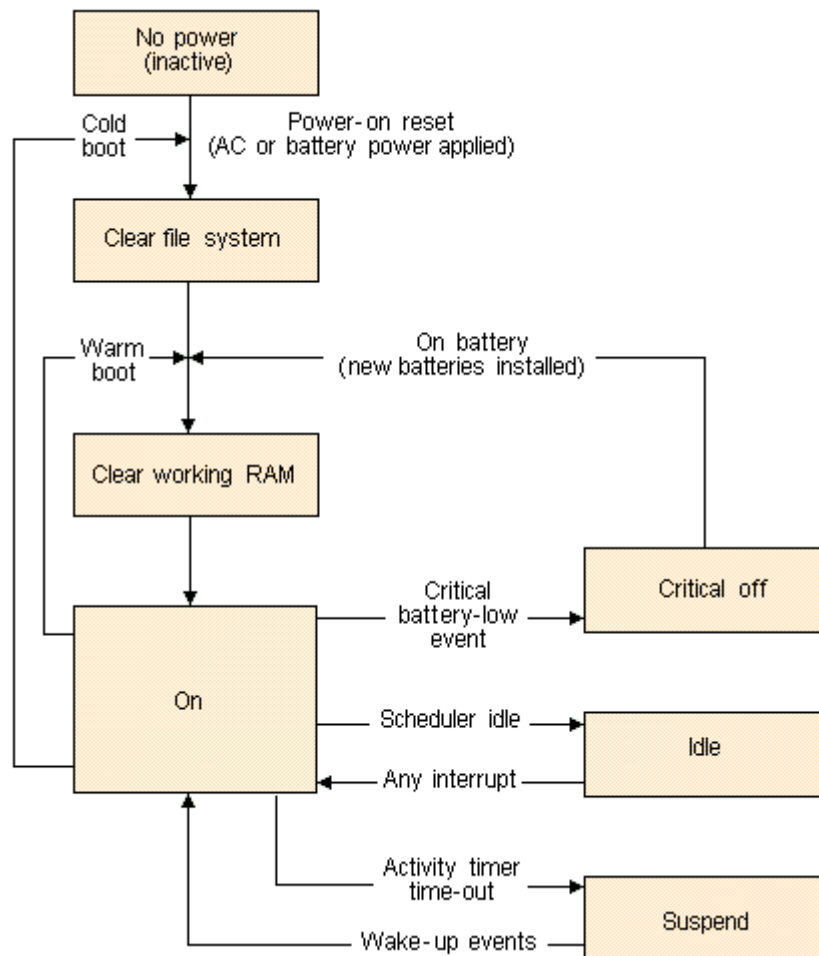
Questa è la struttura di un tipico sottosistema Smart Battery:



5 Power Management in Windows CE

Windows CE (7) è un sistema operativo progettato in modo modulare, così da potersi adattare a vari tipi di dispositivi target con caratteristiche di potenza di elaborazione e capacità di memoria diverse. Comprende molti componenti che, se richiesti, devono essere adattati dall'OEM per il dispositivo target per assicurare la compatibilità (per esempio per un file system) ed in generale per avere un comportamento ottimizzato.

Uno di questi componenti è il supporto per il PM: Windows CE non adotta alcuno standard di PM ma si limita a proporre un modello generale illustrato dal seguente diagramma:



Le fasi di Power-on reset e di Cold boot avvengono alla prima accensione del dispositivo, dopo un reset intenzionale oppure dopo che il dispositivo è stato privato di ogni tipo di alimentazione e prevedono una completa reinizializzazione del file system e della RAM.

Il Critical off è uno stato a cui si arriva quando le batterie raggiungono un livello molto basso e serve per preservare il file system. Gli stati di idle e di suspend sono stati rispettivamente a basso e bassissimo consumo di energia. Nello stato di idle viene eliminata l'alimentazione a tutti i dispositivi tranne il processore, permettendo transizioni On - Idle molto veloci, nell'ordine di pochi microsecondi. Nello stato di suspend invece vengono spenti tutti i dispositivi e vi si perviene solitamente dopo un timeout dell'attività dell'utente o dopo la pressione dello switch di accensione.

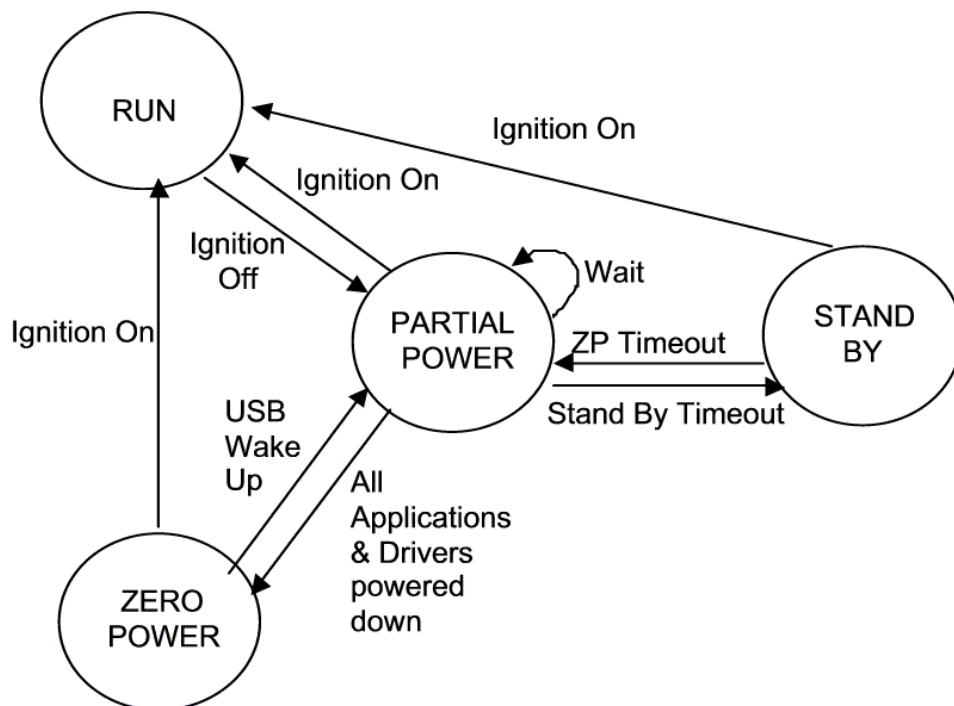
5.1 Power Management in Windows CE for Automotive

Windows CE for Automotive (WCEfA) è la versione di Windows CE progettata per gestire l'Auto PC, un dispositivo simile ad una normale autoradio, installato nel cruscotto di una automobile che fornisce, oltre alle consuete funzionalità audio, avanzate soluzioni per la comunicazione, l'entertainment e la sicurezza.

In WCEfA il sistema di PM è essenziale per assicurare che l'uso dell'Auto PC non interferisca con l'operatività dell'automobile ed ha la responsabilità di gestire i problemi relativi al fatto che l'Auto PC è connesso al sistema elettrico dell'automobile, oltre alla responsabilità di preservare i dati dell'utente.

WCEfA è stato quindi progettato per comportarsi al meglio in scenari chiave quali l'accensione dell'auto, durante la quale il livello di potenza può essere molto fluttuante, un'improvvisa perdita di potenza dovuta a problemi alle batterie, lunghi periodi di parcheggio in cui l'Auto PC non può prosciugare la batteria dell'auto e wakeup parziali ad auto spenta, ad esempio dovuti solo alla necessità di memorizzare il messaggio arrivato ad un pager.

Dall'analisi di questi requisiti è stata creata un'infrastruttura di PM di base estendibile dagli OEMs in base ad opportuni tradeoffs prestazioni/costo (in termini di requisiti di memoria e di potenza). Si tratta di un sistema basato su eventi, eventi hardware generati o dall'utente o dall'auto che si traducono in eventi logici per il sistema di PM, che dovrà gestirli eventualmente cambiando il Power State, come mostrato in figura:



Nelle tabelle seguenti sono descritti i Power States ed elencati i principali eventi:

Power State Name	Description
PM_STATE_RUN	Indicates the Run, or Working, state. The Run state indicates that WCEfA is fully powered on and running.
PM_STATE_STANDBY	Indicates the Standby, or Suspend, state. The standby state indicates the WCEfA CPU is powered off but RAM is preserved.
PM_STATE_PP	Indicates the Partial Power state. The partial power state indicates that WCEfA is operating with lower power consumption than the run state. For example, the display and audio can be turned off to conserve energy.
PM_STATE_ZP	Indicates the Zero Power State (ZP). The Zero Power State indicates WCEfA operating at virtually zero power. It is still able to respond to wake-up events such as Page (the event of receiving a pager message) or Ignition On.
PM_STATE_NOCHANGE	PM_STATE_NOCHANGE is returned by WaitForPMEvent to indicate that no state change happened within a specified timeout period, or that a device or application has acknowledged a power management event or message. The drivers or applications will never receive this special state.

Sample Event	Description
Ignition On	Goes directly to the run state from the current state.
Ignition Off	Goes to time-out to zero power state (see time-out to zero power state below).
Faceplate attach/detach	Signals the replacement or removal of the front panel faceplate for security reasons.
Main battery detected bad	Signals the failure of the main car battery.
Backup battery detected bad	Signals the failure of the backup car battery.
Peripheral signaling partial power required	Signals a service request sent from a peripheral. This can be either RI from a serial interface, a USB power up event, or an external I/O line.
Warm boot	Signals that the user initiated a warm boot of the automotive computing device.
Cold boot	Signals that the user initiated a cold boot of the automotive computing device.
Time-out to zero-power state	Causes the device to suspend to RAM. After an OEM-defined time period, it transitions to the Standby state, then to the Zero Power State.
OEM-defined event	Signals that an OEM-defined event has occurred. An example might be a temperature sensor that would allow you to detect an over-heating condition.

Bibliografia

- (1) Compaq Computer Corporation, Intel Corporation, Microsoft Corporation, Phoenix Technologies Ltd., Toshiba "Corporation Advanced Configuration and Power Interface Specification" Revision 2.0 July 27, 2000 disponibile su www.acpi.info
- (2) Progetto "ACPI4Linux" disponibile su <http://phobos.fs.tum.de/acpi/>
- (3) Intel Corporation "The Instantly Available Power Managed Desktop PC Design Guide" Revision 1.2 September 25, 1998 disponibile su <http://developer.intel.com/technology/iapc/tech.htm#DesignGuides>
- (4) Mike Flora, Microsoft Corporation "OnNow Architecture" ACPI Implementation Workshop, October 7, 1997 disponibile su www.acpi.info/present.htm
- (5) OnNow and Power Management disponibile su www.microsoft.com/hwdev/onnow/
- (6) Smart Battery System Implementer Forum disponibile su www.sbsforum.org
- (7) Microsoft Developer Network (MSDN) Library disponibile su <http://msdn.microsoft.com/library>