Volume 10 Issue 02 Published, May 15, 2006 ISSN 1535-864X DOI: 10.1535/itj.1002

Intel® Technology Journal

Intel® Centrino® Duo Mobile Technology

Intel® Centrino® Duo mobile technology brings uncompromised performance to new heights, delivering faster performance and improved battery life while maintaining thermal cooling and small form factor. This issue of Intel Technology Journal (Volume 10, Issue 2) discusses the revolutionary technology implemented in the Intel® Core™ Duo processor and its supporting platform components.

Inside you'll find the following articles:

Introduction to Intel® Core™ Duo Processor Architecture

CMP Implementation in Systems Based on the Intel® Core™ Duo Processor

Power and Thermal Management in the Intel® Core™ Duo Processor

System Memory Power and Thermal Management in Platforms Built on Intel® Centrino® Duo Mobile Technology

Intel® 945GMS Express Chipset for Small Form Factor Platform Based on Intel® Centrino® Duo Mobile Technology

WLAN System, HW, and RFIC Architecture for the Intel® PRO/Wireless 3945ABG Network Connection

MIMO Architecture for Wireless Communication

More information, including current and past issues of Intel Technology Journal, can be found at:
http://developer.intel.com/technology/itj/index.htm

Volume 10 Issue 02 Published, May 15, 2006 ISSN 1535-864X DOI: 10.1535/itj.1002

Intel® Technology Journal

Intel® Centrino® Duo Mobile Technology

Articles

Preface iii

Foreword v

Technical Reviewers vii

Introduction to Intel® Core™ Duo Processor Architecture 89

CMP Implementation in Systems Based on the Intel® Core™ Duo Processor 99

Power and Thermal Management in the Intel® Core™ Duo Processor 109

System Memory Power and Thermal Management in Platforms Built on Intel® Centrino® Duo Mobile Technology 123

Intel® 945GMS Express Chipset for the Small Form Factor Platform Based on Intel® Centrino® Duo Mobile Technology 133

WLAN System, HW, and RFIC Architecture for the Intel® PRO/Wireless 3945ABG Network Connection 147

MIMO Architecture for Wireless Communication 157

THIS PAGE INTENTIONALLY LEFT BLANK

Preface

Meet the Intel® Core™ Duo Processor
By Lin Chao
Publisher, *Intel Technology Journal*

The Intel® Core™ Duo processor is the first microprocessor to use Intel's groundbreaking 65-nanometer semiconductor process, the first to have dual core on die and the first Intel® processor to be used by Apple Macintosh* computers. The Intel Core Duo processor balances dual-core computing capabilities with power savings that enable improved battery life in notebooks. Its major performance boost is achieved by integrating dual cores on the die using Chip Multi-Processing (CMP) architecture.

At Apple's annual shareholders meeting in April 2006, CEO Steve Jobs stated, "Intel has a great roadmap. This new [Intel Core Duo] chip is phenomenal–it blows away anything other suppliers have, including our former suppliers." Jobs described the company's upcoming products as "the best I've ever seen in my life." Intel's processor is an important part of Apple Computer's exciting upcoming products.

This issue of Intel Technology Journal (Volume 10, Issue 2) reviews the Intel Core Duo processor's architecture design, system, power, thermal and wireless features. It examines in detail the design considerations used in the Intel Core Duo processor to achieve a balance between improving performance and saving watts.

The seven technical papers include an overview paper describing the key objectives for Intel Core Duo processor architecture. The Core Duo processor is the first mobile processor to implement Chip Multi-Processing (CMP), also known as dual core on die. Its major performance boost is achieved by integrating the cores on the die. This paper asserts that continuing single-thread performance is costly in terms of power and may achieve diminishing returns in terms of efficiency. It explores the potential for greatly improved performance through parallelism between threads running on the two cores.

System, power and thermal topics are reviewed in the next four papers. The second paper examines the design considerations used in the Intel Core Duo processor. It provides recommendations on how to optimize software code developed for the Intel Core Duo processor so that future applications can take full advantage of the new design. The third paper discusses the power control unit and its impact on the power-saving mechanisms, and describes the distribution of P-states and other techniques used to improve the system power consumption including the impact of multi-threading on power consumption. The fourth paper looks at specific form-factor requirements for the mobile "thin and light" platform and limitations for the kinds of cooling solutions these platforms can have. With increased memory speeds and capacities, we are now reaching the point where the memory thermals are starting to exceed the cooling capabilities of mobile systems. This paper discusses memory throttling techniques implemented in platforms built on Intel® Centrino® Duo mobile technology. The fifth paper describes the power management features of the Intel® 945GMS Express Chipset Graphics and Memory Controller Hub (GMCH), the Intel® 82801GBM/GHM I/O Controller Hub (ICH), and the overall platform features.

The last two papers focus on wireless technologies. The sixth reviews the work and design approach in the development of the Intel® PRO/Wireless 3945ABG Network Connection card for the Intel Centrino Duo mobile technology platform. The seventh paper addresses one of the key ingredients for current and future versions of Intel Centrino mobile technology, which is the ability to provide users with fast and low-power communications. This is achieved by using multiple-input multiple-output (MIMO) architecture for the design of a superior wireless communication system with respect to reliability, throughput, and power consumption.

After reading these papers, we hope you agree Intel's Core Duo processor is phenomenal!

Foreword

Intel® Centrino® Duo Mobile Technology:
The Beginning of an Era of Mobile Multi-Core Computing

By Dadi Perlmutter
Senior Vice President
General Manager, Mobility Group

For many years mobile computing was perceived as a performance-compromised platform. The mobile PC was just a smaller, "portable" PC, and the perception was that it lacked the necessary performance and features. Intel® Centrino® mobile technology changed this position dramatically. Centrino brought uncompromised performance to a very small, thin and light form factor, while delivering the required mobility capabilities of long battery life and wireless connectivity. It redefined the notebook, creating a new category which is being wildly accepted by users. Before Centrino, notebook percentage of PCs was in its teens; post Centrino, this penetration is in the 30% range and growing.

The new Intel® Centrino® Duo mobile technology brings uncompromised performance to new heights. It delivers faster performance than many desktop computers shipping today while improving battery life and maintaining thermal cooling and small form factor. All this is possible due to the revolutionary technology implemented in the Intel® Core™ Duo processor and its supporting platform components. The Intel Core Duo processor brings dual core to the notebook with "leap-ahead" performance. The processor is developed in 65 nm process technology, and delivers almost double the performance of a similar die size in the previous process generation.

Of notable distinction in the Intel Core Duo processor is its smart-shared cache and dual-core power management. Unlike other architectures, this processor is a complete redesign of the cache and bus interface unit. It implements a dual-port shared cache, which enables both cores to have access to the full cache. This enables threads to dynamically use the required cache capacity. As an extreme example, if one of the cores is inactive, the other core will have access to the full cache. It also enables very efficient sharing of data between threads running in different cores.

Especially challenging for the Intel Core Duo processor was the support of dual core while supporting Intel SpeedStep® technology. Cores can be power-managed independently while coordinating their states for the support of deeper sleep. In deeper sleep, the cache is automatically flushed to memory and its voltage significantly lowered to sustaining transistor level. This is a huge benefit to battery life. These and many other innovations described in this issue's articles make the difference between other dual-core processors and Intel Core Duo processor's smart dual core optimized for notebooks.

The Intel Core Duo processor is the beginning of a family of Intel dual- and multi-core mobile processors. Merom, soon to be introduced in the latter half of 2006, is a significant revamp of Intel Core Duo processor's microarchitecture, which is more focused on core enhancements, delivering performance improvements while maintaining power. Merom's performance efficiency is so compelling that the processor will also be offered in desktops and DP servers. Those products are the genesis of a new era of energy-efficient computing. Future generations will continue enhancing the

core capabilities and increasing the number of cores to new levels of power-efficient performance which will be necessary to execute the requirements of future applications.

This issue of Intel Technology Journal (Volume 10, Issue 2) examines in detail the Intel Core Duo processor capabilities as well as the chipset and wireless capabilities of the Intel Centrino Duo processor-based platform. I'm very proud to have led the outstanding team that developed those wonderful capabilities, which will transform our lives and are critical for Intel's future success.

Technical Reviewers

David W. Browning, Mobility Group
George Cai, Digital Enterprise Group
Jack Doweck, Mobility Group
Yuval Finkelstein, Mobility Group
Koby Gottlieb, Software and Solutions Group
Steve Gunther, Digital Enterprise Group
Benson Inkley, Digital Enterprise Group
Shmuel Levy, Mobility Group
Prakash Narayanan, Mobility Group
Joseph Nuzman, Mobility Group
Shmuel Ravid, Mobility Group
Roni Rosner, Corporate Technology Group
William Sanderson, Digital Enterprise Group
Eli Savransky, Mobility Group
Nilesh Shah, Mobility Group
Shlomit Weiss, Digital Enterprise Group
Eliezer Weissmann, Software and Solutions Group
Paul Williams, Mobility Group

THIS PAGE INTENTIONALLY LEFT BLANK

Introduction to Intel® Core™ Duo Processor Architecture

Simcha Gochman, Mobility Group, Intel Corporation
Avi Mendelson, Mobility Group, Intel Corporation
Alon Naveh, Mobility Group, Intel Corporation
Efraim Rotem, Mobility Group, Intel Corporation

Index words: Intel Core Duo, low power, CMP, multi-threading

ABSTRACT

The Intel® Core™ Duo processor is a new member of the Intel® mobile processor product line. It is the first Intel® mobile microarchitecture that uses CMP (Core Multi-Processor; i.e., multi cores on die) technology. Targeted to the market of general-purpose mobile systems, the Intel Core Duo core was built to achieve high performance, while consuming low power and fitting into different thermal envelopes.

In order to achieve the required performance, a CMP-based microarchitecture was designed to achieve power-efficient architecture, each performance improvement was evaluated against the power cost, and only the power-efficient performance features were implemented.

On top of that, special hardware mechanisms were added to better control the static and the dynamic power consumption. As a result, the Intel Core Duo processor provides higher performance in the same form factors without needing to increase the cooling capability.

INTRODUCTION

The Intel Core Duo processor is a new member of the Intel mobile processor product line. It is the first Intel mobile microarchitecture that uses CMP (multi cores on die) technology. Building a general-purpose mobile core is a challenging task since, on the one hand, the system needs to maintain the highest level of performance, while on the other hand, the system must fit into different thermal envelopes, as illustrated by Figure 1, and improve power efficiency.

Intel Core Duo is based on Pentium M processor 755/745 core microarchitecture with few performance improvements at the level of each single core. The major performance boost is achieved from the integration of dual cores on the die (CMP architecture). This agrees with our assessment that continuing to improve single thread performance is rather costly in terms of power and may achieve diminishing returns in terms of efficiency, if major microarchitecture enhancements are not made. The big potential for improved performance is through exploring parallelism between threads. However, the CMP architecture presents many challenges for power and thermal control to still fit into the mobility constraints.

Figure 1: Products using different thermal envelopes*

In this paper we present the new Intel Core Duo microarchitecture and show how the need to target power-efficient general-purpose processors has affected many of our decisions. We provide a general overview of the different ingredients of the Intel Core Duo system, while the other papers in this issue of the *Intel Technology Journal* focus on more specific aspects of the system such as the CMP microarchitecture and the power and thermal control methods.

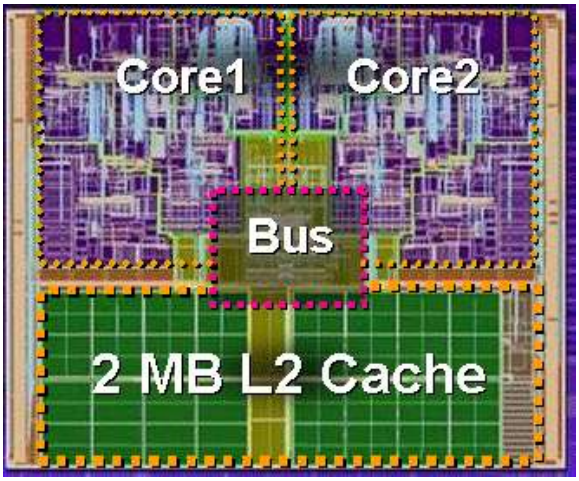


Figure 2: Intel Core Duo processor floor plan

As Figure 2 shows, Intel Core Duo technology is based on two enhanced Pentium M cores that were integrated and use a shared L2 cache. The way we integrated the dual core in the system had a major impact on our design and implementation process. In order to meet the performance and power targets we aimed to do the following:

- Keep the performance similar to or better than that of single thread performance processors in the previous generation of the Pentium M family (that use the same-size L2 cache).
- Significantly improve the performance for multithreaded and multi-processes software environments.
- Keep the average power consumption of the dual core the same as previous generations of mobile processors (that use a single core).
- Ensure that this processor fits in all the different thermal envelopes the processor is targeted to.

In this paper we provide a high-level description of the main Intel Core Duo features and discuss how each feature fits into the targets of the various projects.

THE IMPROVED PENTIUM M PROCESSOR-BASED CORES

The core of the Intel Core Duo processor-based technology is an enhanced Pentium M processor 755/745[1] core converted to 65nm process technology. The main focus of the core enhancements was to do the following:

- Support virtualization (Virtualization Technology[2]) [3].
- Support the new Streaming SIMD Extension (SSE3) [4].
- Address performance inefficiencies mainly in the handling of SSE/SSE2, FP (x87) and some long latency integer instructions.

Intel Core Duo Processor-based Technology Core Performance Improvements

Intel Core Duo processor-based technology introduces performance improvements in the following areas:

- Streaming SIMD Extensions (SSE/2/3)
- Floating Point (x87)
- Integer

The main difficulty with SSE implementation in Pentium M is caused by the fact that SSE/2/3 is a 128-bit wide microarchitecture while the Pentium M execution core is 64-bits wide (in order to meet power and energy constraints). Making the machine twice as wide may produce more heat and so will have a significant impact on the Thermal Design Point (TDP) of the system as well as some impact on battery life. Since the Pentium M was primarily designed for mobility we preferred to make it relatively narrow and cope with the SSE performance issues. The by-product of this tradeoff is that each SSE vector operation is "broken" into 64-bit wide micro-operation (uOp) pairs. Such instructions suffer from several performance bottlenecks in the Pentium M pipeline, mainly in the Front End (FE) of the pipeline. For example, the Instruction Decoder in the Pentium M processor can potentially handle three instructions per cycle but only the first decoder in a row is capable of handling complex instructions. The other two decoders are limited to single uOp instructions only. This works fine in most cases since the most frequent instructions are single uOp. However, this is not the case with SSE instructions: only scalar SSE operations are single uOps while the vector operations are typically 2-4 uOps. This results in several potential bottlenecks in the

[1] Intel processor numbers are not a measure of performance. Processor numbers differentiate features within each processor family, not across different processor families. See http://www.intel.com/products/processor_number for details.

[2] Intel® Virtualization Technology requires a computer system with a processor, chipset, BIOS, virtual machine monitor (VMM) and applications enabled for virtualization technology. Functionality, performance or other virtualization technology benefits will vary depending on hardware and software configurations. Virtualization technology-enabled BIOS and VMM applications are currently in development.

FE: the Instruction Decoder in the Pentium M can only handle one SSE vector operation per cycle, causing starvation in the rest of the machine. This bottleneck was addressed in the Intel Core Duo core: a new mechanism was introduced that allows lamination of pairs of similar uOps. This mechanism along with enhanced uOp fusion allows handling of the SSE/2/3 vector operation by a single laminated uOp. The instruction decoders were modified to handle three such instructions per cycle, increasing significantly the decode bandwidth of SSE vector operations. The laminated uOps streaming down the pipe are at a certain point un-laminated, reproducing again the 64-bit wide uOp pairs to feed the machine. These changes not only improve performance of vector operations but also save some energy since the FE, no more a bottleneck, can be clock gated whenever its uOp buffer is filled beyond a certain watermark.

Another bottleneck that was discovered was the handling of the floating point (FP) Control Word (CW). The FP CW is part of the x87 state and was usually viewed as "constant"; namely it is loaded once at the beginning and stays constant throughout the program. This is indeed the way the FP CW is used by most of the programs. However there are some FP applications that manipulate the "rounding control" which is located in this register: the default rounding mode is "rounding to nearest even" but before converting results to fixed point, some applications change the round control to "chop" (this is the rule with C programs for example). Such behavior was treated rather inefficiently by the Pentium M core: each manipulation of the FP CW was effectively stalling the pipeline until its completion. The Intel Core Duo core introduced a new renaming mechanism for the FP CW so that four different versions of this register can coexist on the fly without stalling the machine.

Intel Core Duo also improved the latency of some long latency integer operations such as Integer Divide (IDIV). Although these instructions are not very frequent, because of their extremely long latencies, their accumulative affect on integer benchmark scores have shown to be very significant. The basic Divide algorithm has remained unchanged; however, Intel Core Duo Divide logic exploits opportunities for "early exit." The Divide logic calculates in advance the number of iterations that are required to accomplish the operation. This is indeed data dependent; however, it is often significantly smaller relative to the maximal number of iterations. Once the required number of iterations is accomplished the divider wraps up the results. This does not impact the maximal Integer Divide latency; however, on average it is much faster.

Another enhancement that benefits different kinds of applications is the introduction of a new mechanism of H/W prefetcher. This mechanism identifies streaming loads at a very early stage in the machine and speculatively predicts the future incarnation of these loads. These speculative requests are looked up in the shared L2 cache and if miss, they're speculatively prefetched from the external memory. This mechanism is dynamically deactivated whenever there are many demand requests pending (a watermark mechanism). The benefit of this change is an average reduction in load latency.

The performance implication of these enhancements on single-threaded (ST) applications as well as on multithreaded (MT) applications are discussed in [1]

CMP–GENERAL STRUCTURE

Intel Core Duo processor-based technology implements shared cache-based CMP microarchitecture in order to maximize the performance of both ST and MT applications (assuming the same L2 cache size). Figure 3 describes the general structure of our implementation. The figure shows the following:

- Each core is assumed to have an independent APIC unit to be presented to the OS as a "separate logical processor."
- From an external point of view the system behaves like a Dual Processor (DP) system.
- From the software point of view, it is fully compatible with Intel Pentium 4 processors with Hyper-Threading (HT) Technology[3] [6], and DP-based systems. However, special optimizations could be applied to improve the performance of the share-based cache organization.
- Each core has an independent thermal control unit (discussed later in this paper and also covered in [2]).
- The system combines per-core power state together with package-level power state.

The paper *CMP Implementation in Intel Core Duo Systems* [1] extends the discussion on the CMP implementation and compares its performance with other configurations such as the use of split cache architecture. The results shown there indicate that the new proposed

[3] Hyper-Threading Technology requires a computer system with an Intel® Pentium® 4 processor supporting HT Technology and a HT Technology enabled chipset, BIOS and operating system. Performance will vary depending on the specific hardware and software you use. See www.intel.com/products/ht/Hyperthreading_more.htm for additional information.

microarchitecture maximizes the performance benefits of both ST and MT execution at a given cache size. The enhancements we implemented in each of the cores allow us to improve both the ST performance (in specific cases) as well as the MT execution. It also allows us to improve the power and the thermal control of the system, and to achieve similar average power consumption, as was the case in the single-core Pentium M processor.

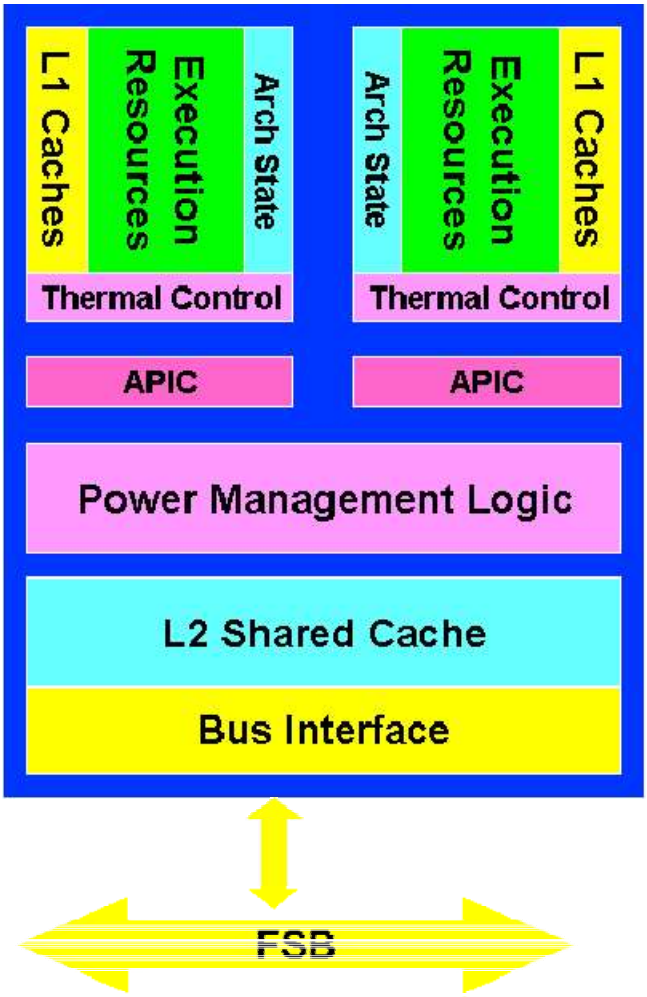


Figure 3: The general structure of the Intel Core Duo implementation

POWER CONTROL

Extending the battery life, while improving the performance, was one of the main goals in designing the Intel Core Duo processor. Battery life is affected by dynamic power, caused when the processor is active, and by static power, which is the power wasted when a unit or the entire processor is not active. Intel Core Duo microarchitecture saves both types of power.

Figure 4 describes the general process we followed in order to reduce the power during the development cycle of the Intel Core Duo processor. As can be seen, the average power consumption was reduced by handling the problem at all different levels of the design, starting with adjusting the process technology through all the design stages of production.

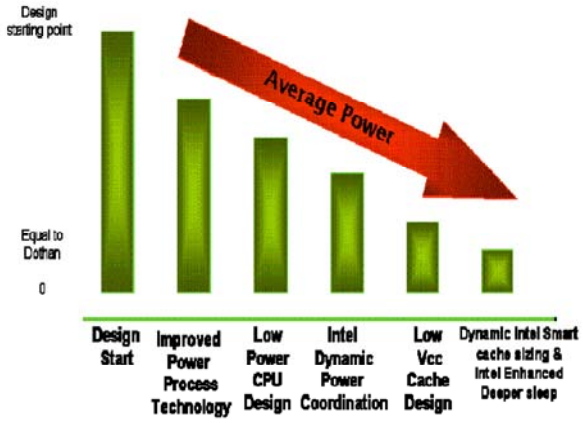
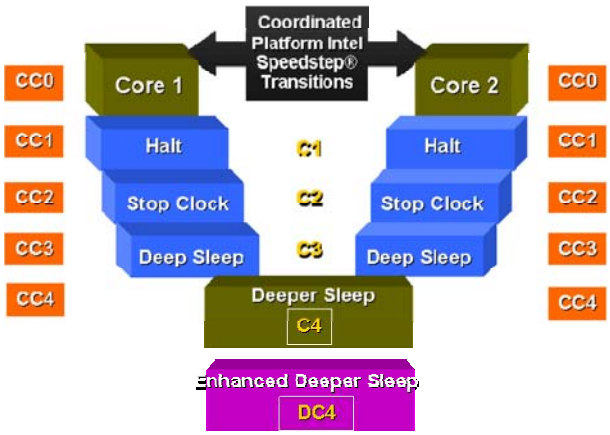


Figure 4: Low-power processor–design process

In order to save leakage power, the Intel Core Duo system uses mainly two techniques: enhanced sleep states control and Dynamic Intel® Smart cache sizing. In order to control the active power consumption, Intel Core Duo technology uses a technique based on Intel SpeedStep® technology .

The traditional way to control the power and the thermal of the system is via a software/hardware interface. One of the most common schemes to achieve this is called ACPI [5], where the system defines different levels of sleep modes, and each of the states represents a more efficient way to save power, at the expense of a longer time to bring the system back into operational mode. (For more details on this method, please see [2]). The challenge of adding a second core on die while improving the overall power-consumption demands an improvement to the power states of the system in order to avoid power being wasted whenever a core is not active. We face two main problems: (1) since only a single power plane is used, it forces us to run all cores with the same voltage and frequency, and (2) the chipset and the OS see both cores as a single entity that has the same state at the same time. Thus, the Intel Core Duo processor presents two separate views on the power state of the system; internally we manage the states of each core independently (we call it per-core power state) and externally we view the system as having a single, synchronized power state. Figure 5 provides an overview of this approach.



- CPU/package sleep states**:**
- **C0 – Active** CPU is on
- **C1 – Auto Halt** Core clock is off
- **C2 – Stop clock** Core and bus clock are off
- **C3 – Deep sleep** Clock generator is OFF
- **C4 – Deeper sleep** Reduced VCC
- **DC4 –Deeper C4** Further reduced VCC

Figure 5: Power states of the Intel Core Duo processor

As we can see the Intel Core Duo processor defines five different sleep states of the system. The first three states allow local power-saving measures to be activated individually per core, while the last two states require a coordination of the entire package for the power-saving measures to be activated.

A core which is in C_0, power state is assumed to be in running mode. When the core has nothing to do, the OS issues a halt command that moves it to CC1, where execution is halted and clocks are stopped. When it detects even lower levels of activity (via the ACPI mechanisms [2]), the OS will further promote the idle state of each of the cores beyond CC1 to CC2, CC3, or CC4 states, based on the core activity history. In the CC2 and CC3 states, additional core-level power-saving measures can be activated, achieving a lower average power consumption. Starting from C4 state, core voltage reduction is applied to further increase average power savings. Since the cores are connected to the same power plane, this must be done in coordination between the two cores, and this is known as package-level C4 and package-level DC4.

While being in a sleep state, the system still consumes static power (leakage). In Intel Core Duo technology, we implement an advanced algorithm that tries to anticipate the effective cache memory footprint that the system needs when moving from a deep sleep state to an active mode. The new mechanism keeps only the minimum cache memory size needed active, and it uses special circuit techniques to keep the rest of the cache memory in a state that consumes only a minimal amount of leakage power.

In order to control the active power consumption, Intel Core Duo technology uses Intel SpeedStep technology. When a set of working points is defined, each one has a different frequency and voltage and so different power consumption. The system can define at what working point it works in order to strike a balance between the performance needs and the dynamic power consumption. This is usually done via the OS, using the ACPIs.

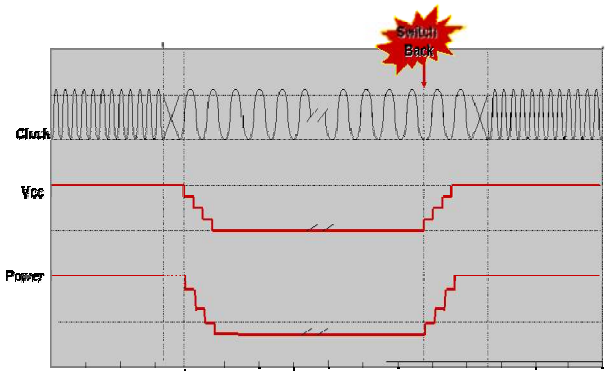


Figure 6: Changing working point in Intel Core Duo processor

The way the system moves from one working point to another is described in Figure 6. As illustrated, in order to move from a "high" working point to a lower one, the system can switch the frequency almost immediately, but it will take the system some time to lower the voltage. When moving from a low working point to a higher one, we need to increase the voltage first (slow operation) and only then can we increase the frequency.

By extending the hardware mechanisms to better support advanced power states and sleep states the Intel Core Duo processor achieves improved power performance efficiency. The power-efficiency improvement over processor generations is shown in Figure 7. As a result, the Intel Core Duo processor provides higher performance in the same form factor without needing to increase the cooling capability.

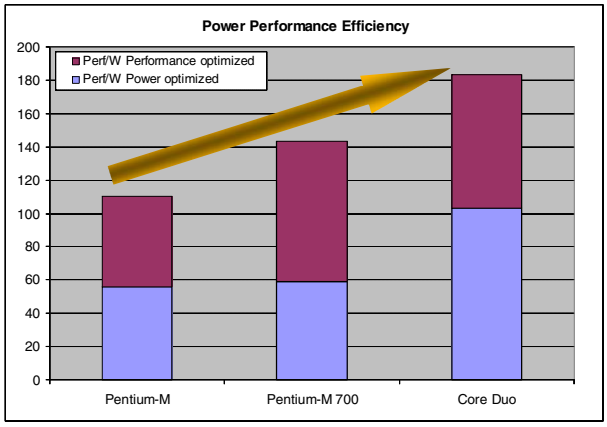


Figure 7: Power performance efficiency

THERMAL DESIGN POINT

Thermal management is another fundamental capability of all mobile platforms. Managing the platform thermals enables us to maximize CPU and platform performance within thermal constraints. Thermal management also improves ergonomics with a cooler system and lower fan acoustic noise.

In order to better control the thermal conditions of the system, the Intel Core Duo processor presents two new concepts: the use of digital sensors for high accuracy die temperature measurements and dual-core multiple-level thermal control.

In the previous Pentium M processor, a single analog thermal diode was used to measure die temperature. Thermal diode cannot be located at the hottest spot of the die and therefore some offset was applied to keep the CPU within specifications. For these systems it was sufficient, since the die had a single hot spot. In the Intel Core Duo processor, there are several hot spots that change position as a function of the combined workload of both cores. Figure 8 shows the differences between the use of the traditional analog sensor and the use of the new digital sensors.

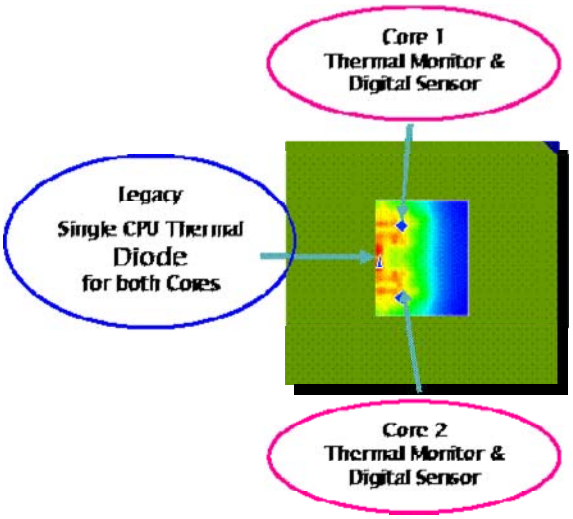


Figure 8: Analog vs. digital sensors in Intel Core Duo processors

As we can see the use of multiple sensing points provides high accuracy and close proximity to the hot spot at any time. An analog thermal diode is still available on the Intel Core Duo processor. The use of a digital thermometer allows tighter thermal control functions, allowing higher performance in the same form factor. The improved capability also allows us to achieve better ergonomic systems that do not get too hot, can operate more quietly, and are more reliable. Unlike diode-based thermal management algorithms that require some temperature guard band (or activating the self throttle mechanism as a safety-net), the digital thermometer is tested and calibrated against specifications. Full functionality and reliability of the processor are guaranteed, as long as the reported temperature is equal to or below the maximum specified temperature. Any inaccuracy or offset are programmed into the device and already accounted for.

The thermal measurement function provides interfaces to power-management software such as the industry-standard ACPI. Each core can be defined as an independent thermal zone, or a single thermal zone for the entire chip. The maximum temperature for each thermal zone is reported separately via dedicated registers that can be polled by the software.

In addition to the polling capability, the digital thermometer implements event-based reporting. Control software programs temperature thresholds that require actions. Such actions can be fan activation or passive control policy such as dynamic voltage and frequency scaling. Upon temperature crossing of the threshold, an APIC-defined interrupt is generated and it initiates the requested action.

Intel Core Duo technology implemented a dual-core power monitor capability. Power monitor functionality is provided in order to prevent thermal exceptions, and it can throttle the CPU once the temperature exceeds specifications. The overview of the power monitoring logic is described in Figure 9.

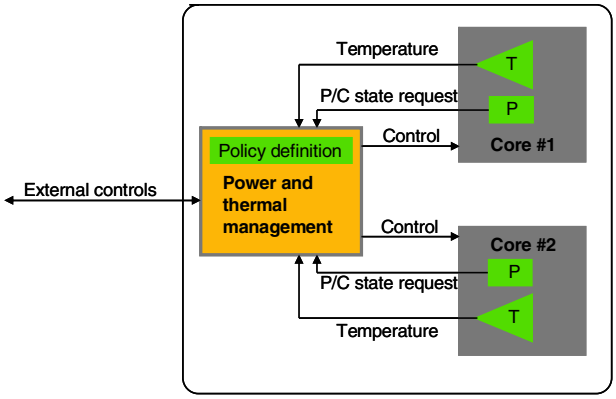


Figure 9: Thermal control overview

The power monitor continuously tracks the die temperature. If the temperature reaches the maximum allowed value, a throttle mechanism is initiated. A multi-level tracking algorithm is implemented. Throttling starts with the more efficient dynamic voltage scaling policy and if not sufficient, the power monitor algorithm continues lowering the frequency. If an extreme cooling malfunction occurs, an Out of Spec notification will be initiated, requesting controlled shutdown. Lastly, the CPU can initiate a thermal shutdown and turn off the system.

Power and thermal management activities in notebook computers are usually performed by the OS and platform control functions. These thermal management features are designed to best serve user preferences under notebook constraint conditions. Thermal monitor function is not expected to be activated under these normal operation conditions. The thermal monitor mechanism ensures that the CPU will never exceed the CPU-specified parameters and guarantees functionality and reliability at any time.

The use of high accuracy temperature reading together with thermal monitoring protection enables high performance in thermally limited form factors, while allowing improved ergonomics and high reliability.

PLATFORM POWER MANAGEMENT

Intel Core Duo processor technology closely interacts with other components on the platform. One such component is the Voltage Regulator (VR). VR power losses at low CPU utilization may get as high as the CPU power. The losses of the VR are due to the need to deliver high current at quick respond times. Intel Core Duo processors implemented a feedback mechanism to the VR. The CPU tracks its activity at any time. If utilization goes down, the CPU communicates a signal to the VR, allowing it to switch to a lower power consumption. A lower power state can be either a reduced number of phases or asynchronous operation. The communication is done using the voltage ID lines and PSI signal as described in Figure 10.

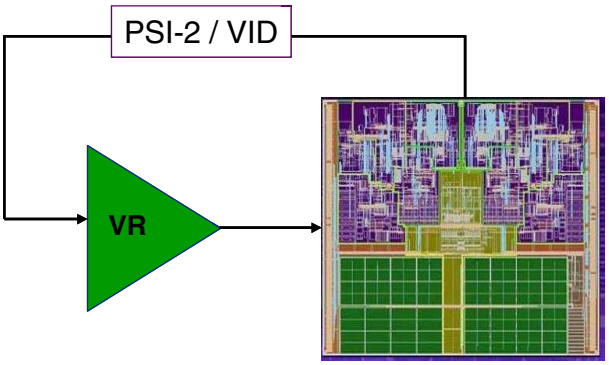


Figure 10: Voltage regulator interface

The CPU has internal knowledge of the activity demand and it communicates a request to go to higher power early enough for the VR to get ready for the increased demand.

Another power optimization is load line control. At low CPU activities, the voltage drop on the load line is smaller resulting in higher voltage and power to the CPU. At low workloads, the CPU reduces the voltage request, and early enough, before power consumption increases, a voltage increase request is sent to the VR.

Using utilization knowledge, available in the CPU, Intel Core Duo technology made it possible to reduce platform power, increase battery life, and improve form factor ergonomics.

INTEL® CORE™ SOLO PROCESSOR

In order to fit into very limited thermal constraints and power consumption, the Intel Core Duo processor has a derivative that contains a single core only. This can be achieved by either disabling one of the cores either at the OS level or as a BIOS option, or at the architecture level, where one core is disconnected from the power grid.

The first option is a user or OS decision. If you run a single-core OS on an Intel Core Duo system, it will keep the second core idle, at CC4 sleep state. Please note that due to the way the BIOS is set, each time an interrupt is received or a broadcast IPI is sent, this core may need to wake up and go immediately back to a sleep state, consuming small amounts of dynamic power.

The user can disable the second core via a BIOS option as well. In this case, the system does not recognize the other core and so it is kept in CC4 state all the time, consuming no dynamic power at all.

The disadvantage of the two methods described above is that the core still consumes static power. In order to avoid this and reduce the power consumption of the core even further, Intel introduces the single-core version of Intel Core Duo technology, called Intel® Core™ Solo processor, which disconnects the non-active core from the power grid, or saves the area and does not fabricate this part at all.

CONCLUSION

The Intel Core Duo processor is the first Intel processor that implements dual core on die. The processor addresses new challenges for providing the best performance under power and thermal constraints.

This paper described the main architectural features of the new processor focusing on the different performance, power, and thermal control features of the processor and of the system.

By applying punctual control between the performance, power and thermal features implemented in the Intel Core Duo system, we achieved a significant improvement in performance, at the same power consumption, and with improved thermal control mechanisms.

REFERENCES

[1] Avi Mendelson, et al., "CMP Implementation in Intel® Core™ Duo Processor," *Intel Technology Journal, Volume 10, Issue 2, 2006.*

[2] "Power and Thermal Management in the Intel® Core™ Duo Processor," *Intel Technology Journal, Volume 10, Issue 2, 2006.*

[3] "Intel® Virtualization Technology Specification for the IA-32 Intel® Architecture" in ftp://download.intel.com/technology/computing/vptech/C97063-002.pdf

[4] "IA-32 Intel® Architecture Software Developer's Manual Volume 1: Basic Architecture" in ftp://download.intel.com/design/Pentium4/manuals/25366519.pdf (chapter 13)

[5] ACPI Specification at http://www.acpi.info/spec.htm*

[6] G. Hinton, et al., "The microarchitecture of the Pentium 4 Processor," *Intel Technology Journal Q1, 2001.*

AUTHORS' BIOGRAPHIES

Simcha Gochman is a senior principal engineer with Intel's Mobile Platform Group in Haifa, Israel. Simcha has been with Intel for 21 years. Lately he has led the microarchitecture development of the Pentium M processors and of the Core Duo processor. Previous to this, he led the microarchitecture definition of the Pentium Processor with MMX technology and was involved in the design of the 80860 processor and the 80387 numeric coprocessor. Simcha received his M.Sc. degree from the Technion, Israel Institute of Technology in 1984. His e-mail is simcha.gochman at intel.com.

Avi Mendelson is a principal engineer in Intel's Mobile Platform Group in Haifa, Israel, and adjunct professor in the CS and EE departments, Technion, Israel Institute of Technology. He received his B.Sc. and M.S.c degrees from the Technion, Israel Institute of Technology and his Ph.D from the University of Massachusetts Amherst. Avi has been with Intel for 7 years. He started as senior researcher in Intel Labs, later he moved to the Microprocessor group where he serves as the CMP architect of Intel Core Duo processor. Avi's work and research interests are in computer architecture, low-power design, parallel systems, OS related issues and virtualization. His e-mail address is avi.mendelson at intel.com.

Alon Naveh is a senior architect with Intel's Mobile Platform Group in Haifa, Israel, focusing on processor and platform power management. He received his B.Sc. degree from the Technion, Israel Institute of Technology in 1983, and holds an MBA degree from San Jose State University. Alon co-lead the power management definition of the Intel Core Duo architecture and was involved in the definition of the Intel Pentium M power management, PCI Express* and the Odem chipset. Prior to Intel, Alon worked in Motorola Semiconductor and in National Semiconductor. His e-mail is alon.naveh at intel.com.

Efi Rotem is a senior architect with Intel's Mobile Platform Group in Haifa, Israel, focusing on processor and platform power and thermal management. Efi joined Intel in 1995 and was involved in the definition and development of Pentium 4 and Pentium M processors. Previously at Intel, he led the Intel Pentium with MMX technology testing. He received a B.Sc. degree from the Technion, Israel Institute of Technology in 1986. His e-mail is efraim.rotem at intel.com.

Copyright © Intel Corporation 2006. All rights reserved. Intel, Core, Pentium, Intel SpeedStep, and MMX are trademarks or registered trademarks of Intel Corporation

or its subsidiaries in the United States and other countries.

* Other names and brands may be claimed as the property of others.

This publication was downloaded from http://developer.intel.com/.

Legal notices at http://www.intel.com/sites/corporate/tradmarx.htm.

THIS PAGE INTENTIONALLY LEFT BLANK

CMP Implementation in Systems Based on the Intel® Core™ Duo Processor

Avi Mendelson, Mobility Group, Intel Corporation
Julius Mandelblat, Mobility Group, Intel Corporation
Simcha Gochman, Mobility Group, Intel Corporation
Anat Shemer, Software Solutions Group, Intel Corporation
Rajshree Chabukswar, Software Solutions Group, Intel Corporation
Erik Niemeyer, Software Solutions Group, Intel Corporation
Arun Kumar, Software Solutions Group, Intel Corporation

Index words: Intel Core Duo, low power, CMP, multi-threading, software optimizations

ABSTRACT

The Intel® Core™ Duo processor is the first mobile processor to implement Chip Multi-Processing (CMP), also known as dual core-on-die. This first implementation was carefully chosen to deliver maximum performance for a given power. The performance improvement was achieved by enhancing the single-core micro-architecture, which results in better single-threaded performance, and by implementing CMP, which improves the performance of multi-threaded applications and parallel application processing. The focus of this paper is to introduce the reader to the CMP aspects of the Intel Core Duo processor. Since the Intel Core Duo processor was designed to be a mobile processor, we examine in detail the design considerations that had to be taken into account to achieve a balance between performance improvements and power savings, and we provide recommendations on optimizing the code developed for the Intel Core Duo processor so that future applications can take full advantage of the new design.

INTRODUCTION

The Intel Core Duo processor is the first mobile core to implement Core Multi-Processor (CMP) technology on one die. The implementation was carefully chosen to maximize performance, so it can be used as a general-purpose processor, and to minimize power consumption, in order to extend the battery life and have it fit in a large variety of thermal envelopes. The performance improvement was achieved by enhancing the micro-architecture, based on Pentium® M processor-based technology, of the single core, and by combining dual cores on the same die. In order to achieve the power consumption goal, we examined each micro-architectural decision with respect to its power/performance benefit. A general overview of the processor and its unique features can be found in this special issue of the *Intel Technology Journal* [1]. This paper focuses on the multi-core design and performance aspects of the processor, but for each of the decisions we describe here, we discuss how the power and thermal aspects were taken into account as part of our decision.

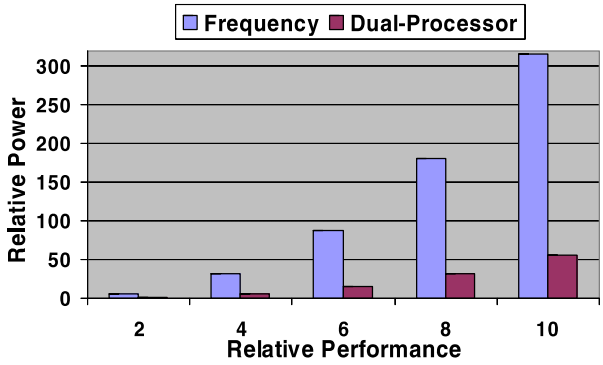


Figure 1: Theoretical power consumption for the same performance–single thread vs. dual thread

The first question one might ask is "why choose a CMP implementation for a mobile processor"? Figure 1 compares the power needed to complete the same amount of work, at the same execution time, assuming frequency scaling vs. using dual cores. In order to conduct the comparison, we assume a single-core processor that consumes 1 Watt at a given frequency and voltage, as a baseline. In order to double its performance one can either double both its frequency and voltage respectively, or he can double the number of cores (assuming perfect scaling of the software). As can be seen in the graph, it is clear

that under these simple conditions a better solution will be to use parallel execution than to improve the speed of the processor to achieve the same performance. It is a known fact that the power (P) a processor consumes depends on the voltage and the frequency of the processor. In order to explain the graph of Figure 1, consider a more realistic relationship between the power the processor consumes and its voltage and frequency. The basic relationship is given by Equation 1:

Equation 1: $P = \propto C V^2 F$

where P stands for power, C for capacitance, V for voltage, $\propto$ is the activity factor and F for frequency. For each frequency within the design space, there is a minimum V that can support it: we call the pair (Fi, Vi), a working point of the processor. As long as Vmin < Vi < Vmax, we can approximate that Fi is linearly dependent on Vi, and for every (Fj,Vj) such that Vj<Vmin, we set the Vj to be equal to Vmin. As a result, within the dynamic range of V, the power has a cube relation with the frequency, while below Vmin, the power has a linear dependency with the frequency. Figure 1 uses Equation 1 to estimate the power consumption of each configuration, but in order to represent more realistic scenario, we use an exponent of 2.5 rather than an exponent of 3 (cubical relation). Unfortunately, the exponential relation between the power and the frequency/voltage is only true as long as the working point is within the dynamic-scaling portion of the voltage and provided enough parallelism is available in the software being used.

Since Intel Core Duo technology is aimed at the general purpose mobile market, the design should be balanced between power consumption and performance. Thus, we used the following criteria to decide between different design alternatives:

(a) When the system runs single-threaded applications, its performance should be the same or better than previous-generation Pentium M processors (with the same cache size and at the same frequency).

(b) When the system runs multi-threaded applications, we wanted to maximize the performance of the execution and preserve power by introducing a new and efficient power and thermal control system.

On top of all the technical hurdles mentioned above, we also had to consider the complexity of different solutions, since our experience told us that complicated solutions consume much power. Thus, for any new feature, the performance improvement must be significant enough to compensate for its complexity.

The primary goal of this paper is to discuss the CMP implementation and resulting performance. We do not focus on the power-saving techniques in Intel Core Duo processors since reference [2] covers that aspect of the system. However, when we discuss our design alternatives and why we chose one solution over another, the reader will notice that power savings (both static and dynamic) were a major factor in our decisions.

The rest of this paper is organized as follows: in the next section we focus on the CMP implementation. This includes the tradeoffs we considered, why we chose the current implementation, and their power and performance impact. Next, we focus on performance measurements, and in last section we extend our discussion to cover software optimizations.

CMP IMPLEMENTATION AND DESIGN CONSIDERATIONS

The Intel Core Duo processor is a new member of the Pentium M processor family. Before discussing how CMP is implemented, let us describe the implementation of current processors in the Pentium M family.

Background – The Structure of the Pentium M Processor

All the Intel processors in the mobility family that preceded the Intel Core Duo processor were uni-processor, and therefore efficiently support only Single Threaded (ST) applications and had the same basic structure as presented in Figure 2.

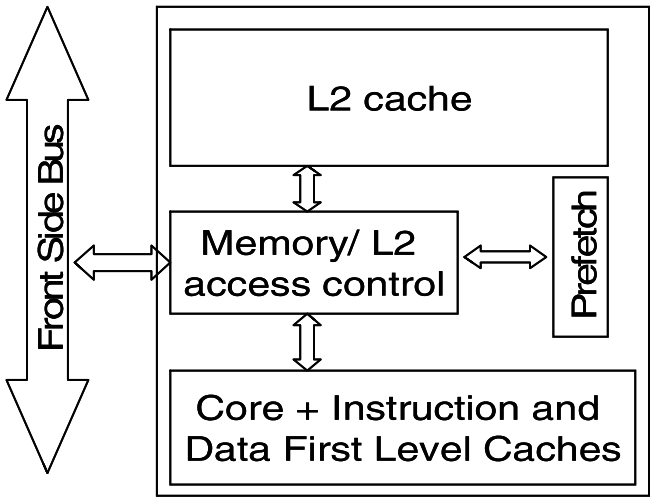


Figure 2: Structure of the memory cluster in the Intel Pentium M processor

Here, all the accesses to the L2 cache, as well as the accesses to the main memory and IO space, were under the supervision of a single control unit, shown in Figure 2 as *Memory/L2 access control* units (also called super-queue). Using this structure, cacheable requests from the

core first looked for the data in the L2 cache and only if not found there (L2 miss), were they forwarded to the main memory via the front side bus (FSB). Uncacheable accesses could be directly sent to the main memory. The *Memory/L2 access control* unit also served as a central point for maintaining coherency within the core and with the external world. Pentium M processors support the MESI [3] coherence protocol that marks each cache line as Modifid, Exclusive, Shared, or Invalid.

In a nutshell, the MESI protocol attaches for each cache line a state that can be M-modified, E-exclusive, S-shared, or I-invalid. A line that is fetched, receives E, or S state depending on whether it exists in other processors in the system. A cache line gets the M state when a processor writes to it; if the line is not in E or M-state prior to writing it, the cache sends a Read-For-Ownership (RFO) request that ensures that the line exists in the L1 cache and is in the I state in all other processors on the bus (if any).

The *Memory/L2 access control* unit manipulates the coherency of each level of the caches independently. It contains a snoop control unit that receives snoop requests from the bus and performs the required operations on each cache (and internal buffers) in parallel. It also handles RFO requests and ensures the operation continues only after it guarantees that no other version on the cache line exists in any other cache in the system.

The CMP Implementation

At the early stages of the project, we considered three alternatives for CMP implementation, as illustrated with two structural alternatives in Figure 3. The first option (Figure 3a) was to put two single-core Pentium M processors, side by side, split the L2 cache among them, and communicate between the cores via the FSB or another fast interconnect.

Both other options called for a shared L2 (Figure 3b) with a different implementation of the coherence protocol; one option called for the same basic MESI table as in a single core but "adjusting it" to the new structures, while the second option called for a simple version of a directory-based protocol to improve the performance of the proposed structure.

The “simple” shared L2 implementation called for us to take advantage of the fact that the latency of the access to the L2 cache is significantly longer than the L1 access latency. This difference in latency enables us to check/update the status of the cache line in first level caches in parallel with L2 access. Therefore, this option increases the active power consumption (with respect to a single core) for snoop activities, but keeps the static power (leakage) the same as the single core, since no additional tables are used.

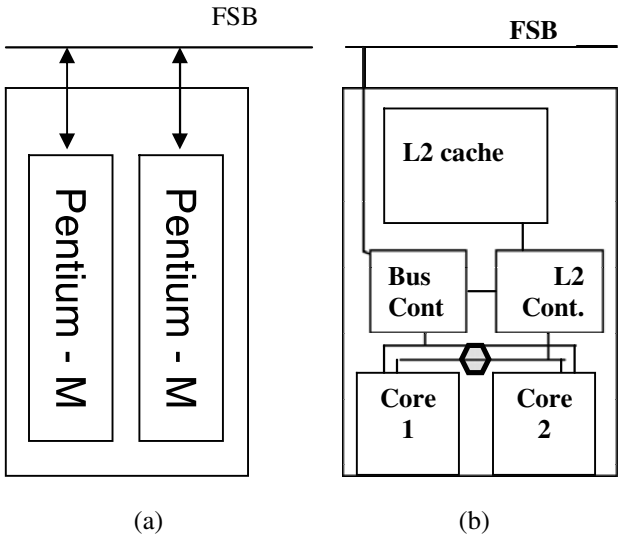


Figure 3: Implementation alternatives

The directory-based solution calls for extending the MESI information, as part of the L2 structure, and keeping information regarding the ownership on L2 cache lines. Here we assume that snoops are sent to the other core by the L2 controller, and only when needed. Thus, when a core accesses the line in the L2 cache, the cache controller knows if the line is shared with the other cache, and based on this information the cache control unit can optimize the number of snoops sent to the other L1. This technique reduces the active power due to reduced snoop activity, but increases the design complexity and the static power due to larger tag arrays.

Using the three criteria described in the introduction, we analyzed the performance and power and firstly eliminated the first option (3a). The reason for this was that it would reduce the performance of ST applications, since it provides only half of the cache size for each core. We also observed that the use of a split L2 cache could cause performance degradation when running multi-threaded (MT) applications with shared data, preventing effective data sharing between the threads, and requiring long latencies when moving data from one core to another. On top of that, it may reduce the performance of MT and parallel application processing since it could not dynamically partition the L2 cache

Deciding between the two implementations of the shared L2 cache was a tough task. The performance of the two options was very close and so we had to make our decision based on power efficiency. We decided to implement the simple solution and not the directory-based architecture due to its complexity. The directory-based solution was found to be less favorable since battery life mainly depends on static power consumption and less on dynamic power.

The general structure of the Intel Core Duo CMP implementation is given in Figure 3b. Comparing it with Figure 2 shows few structural changes: (1) The core and first-level caches structure is duplicated; (2) the traditional *memory and L2 control* unit (super-queue) is partitioned into two logical units: the *L2 controller* that handles all the requests to the L2 from the core and from the external bus (snoop requests) and a *bus control unit* that handles all the data and IO requests to and from the external bus; (3) in order to balance the requests to the L2 and memory, we added a new logical unit (represented by the hexagon) that aims to guarantee the fairness between the requests coming from different cores; and (4) we extended the prefetching unit to handle separately hardware prefetching by each core.

The new structure of the shared area allows us to enhance the performance while reducing power consumption. The new partitioned structure of the super-queue allows us to implement new power and performance optimizations, since the *L2-control unit* was designed to be relatively small, simple, and fast in order to reduce the latency to the L2 cache without increasing the power consumption. The *Bus Control Unit* was designed to be larger and more complicated, but since it was found to be more relaxed in timing, we could design it to have less leakage and even reduce its active power.

The power and performance results were measured on Intel Core Duo silicon and justified the CMP architecture we choose. We discuss this later in the paper.

THE PROTOCOL

From the external observer, the behavior of a CMP system should be looked at as the behavior of a dual package (DP) system. For that purpose, Intel Core Duo processor implements the same MESI protocol as in all other Pentium M processors.

In order to improve performance, we optimized the protocol for faster communication between the cores, particularly when the data exist in the L2 cache. A noticeable example of such a modification was done in order to allow the system to distinguish between a situation in which data are shared by the two CMP cores, but not with the rest of the world, and a situation in which the data are shared by one or more caches on the die as well as by an agent on the external bus (can be another processor). When a core issues an RFO, if the line is shared only by the other cache within the CMP die, we can resolve the RFO internally very fast, without going to the external bus at all. Only if the line is shared with another agent on the external bus do we need to issue the RFO externally.

For most Intel Core Duo systems, when only one package exists, this is a very important optimization. In the case of a multi-package system, the number of coherence messages over the external bus is smaller than in similar DP or MP systems, since much of the communication is being resolved internally. The number of required coherency messages is also much smaller than in the case of using a split cache (Figure 3a) which requires all the communication between the cores and split L2 caches to be done over the external bus.

PERFORMANCE MEASUREMENTS

This section describes the different measurements we did on an Intel Core Duo processor-based system. We start with basic measurements and then discuss the impact of programming models and optimizations on the overall power and performance of the system.

Basic Measurements

Two of the basic requirements we had from the system were (1) to keep, or improve the performance of ST applications, using the same frequency and L2 cache sizes, and (2) to take full advantage of parallel execution, when parallelism is available.

Figure 4 compares the performance of Pentium M processor and Intel Core Duo processors, using the same platform, running at the same frequency, and executing all the programs out of the SpecINT benchmark suite. As can be seen, on average, the performance of the Intel Core Duo and the Pentium M are the same.

Figure 5 compares the execution of all the programs out of the SpecFP performance benchmark. Here, some of the programs show a significant improvement over the Pentium M execution on the same platform. The main reason is the use of SSE3 new instructions by the compiler and a few other performance improvements described in [2].

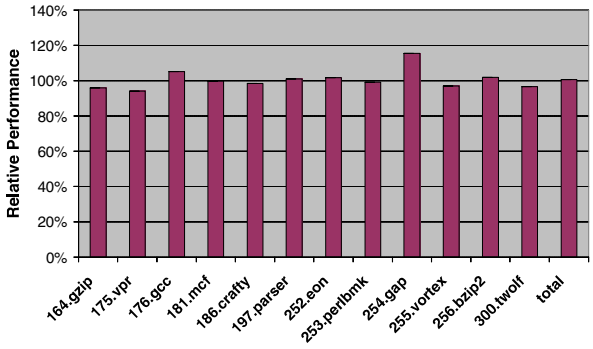


Figure 4: Single-threaded performance–SpecINT Core Duo vs. Pentium M (same cache, same platform)

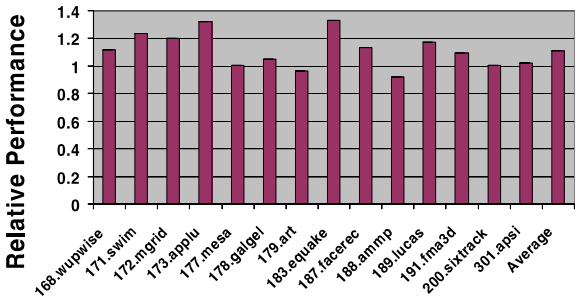


Figure 5: Single-threaded performance–SpecFP Core Duo vs. Pentium M (same cache, same platform)

After achieving the first goal of keeping the performance of an ST application at the same level (or better) as when run on a Pentium M processor, Figure 6 shows the speedup numbers that various MT applications can achieve.

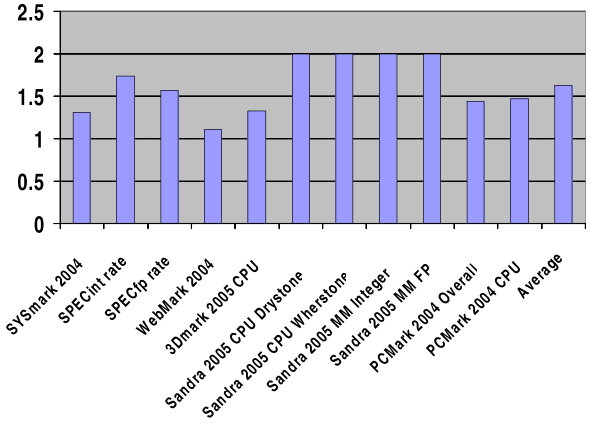


Figure 6: MT speedups

The speedup numbers presented here range from 1.2 to 2, which is the theoretical maximum that can be achieved by two cores. A closer look at the applications that reveal relatively low scalability shows that the main reason for that is lack of parallelism within the application. A few applications, such as SpecFP rate, suffer from high utilization of the bus. In these cases doing the same experiment but with a faster bus yields a better scalability.

Threading Models

When multi-threading an application, the choice of a threading model plays a key role in achieving maximum performance scaling. In this section, we discuss the effect of "data domain decomposition" and "functional domain decomposition" on the performance of an application.

Data domain decomposition usually results in a balanced threading model and is likely to produce a better scalable threading behavior when running the application on platforms with a higher number of processors. Functional domain decomposition is susceptible to imbalanced threads due to thread specific performance characteristics, and hence load-balancing issues need to be considered. A functional domain decomposed model is also likely to limit the scalability by any number of processors. One very important consideration with imbalanced threading behavior in applications is the operating system (OS) scheduling of threads on a CMP system (we illustrate this with an example in the sections below).

Applications With Balanced Threading Models

Applications studied here are CPU-intensive, consuming 95-100% of the CPU with the threads performing equal work and consuming equal processing resources. Here, we discuss the performance of these applications when they run in ST and MT modes. The performance data are measured in seconds.

The graph in Figure 7 indicates performance data for running ST and MT versions of the applications. Cryptography and Video Encoding applications have two MT implementations, and hence, results are indicated as MT1 and MT2. MT1 is implemented using a data domain decomposition methodology, and MT2 is implemented using functional domain decomposition.

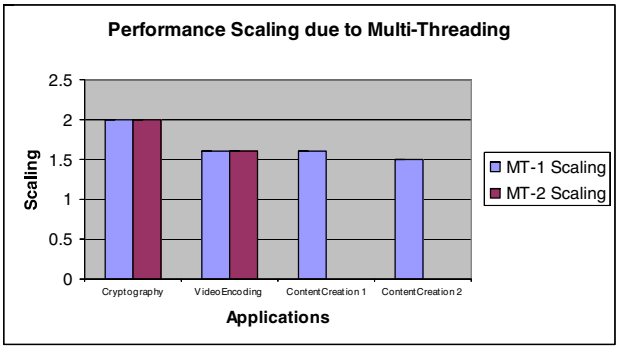


Figure 7: Balanced threading performance

As indicated in Figure 7, MT applications clearly demonstrate significant performance improvement over ST applications. Some of the applications have two different multi-threaded implementations. For example, MT-1, MT-2 versions of the Cryptography workload demonstrate a 2x performance improvement as compared to the ST version.

Applications with Imbalanced Threading Model

In this section, we examine the performance implications on an application with imbalanced threading models. For

this study, a sample game physics engine was created (using Microsoft DirectX*). The sample application has two parts: 1) Physics Computation (collision detection and resolution for graphics objects), 2) Rendering (updated positions are drawn onto screen). The application was deliberately designed such that balanced and imbalanced threading could be studied for a CMP processor:

a. *Balanced*: For this implementation, graphical objects (and background imagery) were divided into two parts and each thread took care of the collision detection and resolution of its own set of objects.
b. *Imbalanced:* In this implementation, one thread was tasked with performing collision detection and resolution for the colliding objects while the other thread calculated the updated positions. The result was the desired goal of the first thread being more CPU intensive than the second thread.

With the two implementations, performance data in different power schemes, MaxPerf and Adaptive, are as shown below. Adaptive mode here refers to the power-saving scheme where the OS optimizes overall power consumption, by dynamically changing CPU frequency on demand, using Intel SpeedStep® technology (the GV3 technology). The MaxPerf mode refers to the power scheme where the processor is always running at the highest clock speed.

Let us discuss the first two data sets in Figure 8 for now.

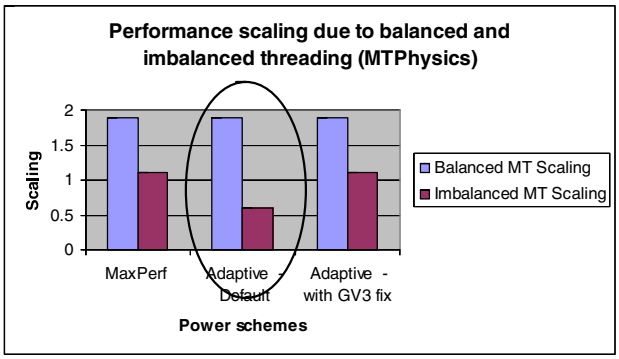


Figure 8: Imbalanced threading performance

The Imbalanced MT (Imbalanced-MT) implementation demonstrates a 2x performance degradation (0.6 scaling) when running in the Adaptive power scheme as compared to MaxPerf (indicated with the circle in Figure 8). In the Imbalanced-MT case, since one of the threads is doing a large amount of the work as compared to the other thread, the thread performing more work keeps migrating between the cores, making effective CPU utilization on the cores at ~50%. On systems running in "Adaptive" (portable/laptop) power mode, this thread migration causes the Windows* kernel power manager to incorrectly calculate the optimal target performance state for the processor. This reduces the operating frequency of both cores even when one of the cores is fully utilized in Adaptive mode and hence causes degradation in performance for the Imbalanced-MT case. Note that this issue may occur while running single-threaded workload as well. To address this issue, Microsoft provided a hot-fix (KB896256) to change the kernel power manager to track CPU utilization across the entire package, rather than the individual cores and hence calculate the optimum frequency for applications.

The third set in Figure 8 indicates data with the kernel hot-fix. In this case, the Imbalanced-MT implementation in Adaptive mode shows expected performance scaling as of MaxPerf mode. With this fix, both cores run at optimum frequency, not causing any degradation in Adaptive (PL) mode.

COMPARING SPLIT CACHE WITH SHARED CACHE

Recently, different architectures use a split last-level cache in order to achieve a fast time-to-market of a dual-core system. Clear downsides of this solution are as follows:

1. Cache coherent-related events that need to be served over the FSB, such as RFO or invalidation signals, greatly impact performance and power.
2. An ST application cannot take full advantage of the entire cache.

The hard partitioned cache may have one significant benefit over the unified cache; that is, it may prevent one application from significantly reducing the amount of cache memory available to an application running on the other core. Thus, in this section we compare two systems: one uses a split L2 cache and the other uses a unified model. In order to make the comparison fair, we present speedup numbers and not absolute numbers.

A sample physics engine game is created (using Microsoft DirectX) to perform this study. The application is MT using data domain decomposition. The threads are synchronized before rendering the updates on the screen. Since the dependency among the threads is very minimal, we expected to achieve ~2.0x performance improvement with the MT version as compared to the ST version.

The split L2 cache indicated approximately a 1.68x performance improvement due to MT. Running the same application on the Intel Core Duo processor-based system demonstrates ~1.90x scaling as per our expectations.

The root cause of the difference in the scaling is due to the shared L2 cache on the Intel Core Duo system. The sample application under study is designed in a way that both threads work on data from a shared data structure. Hence, on the system with the split L2 cache, to get access to the data modified by one processor, the second

processor needs to go to main memory, which results in many L2 cache misses. Since the Intel Core Duo system has a unified L2 cache, a penalty of cache miss and access to the main memory is avoided, as the data modified by one core can be made available to the other core immediately.

OPTIMIZATION OPPORTUNITIES FOR INTEL® CORE™ DUO PROCESSOR

Like any parallel system, the performance and the power of the Intel Core Duo processor may be sensitive to the memory access patterns. In this section we review three optimizations that are very important for getting the best out of the system.

Efficient Use of the Shared L2 Cache

Sharing data between two threads on the Intel Core Duo processor is fastest when done through the L2 cache. This section examines several scenarios for sharing.

One scenario is when one thread brings the data from memory, and the other thread later uses this data directly from the L2 cache. If the single-threaded workload needs to bring the same data several times from memory but the multi-threaded version is carefully designed to use the same data by the two threads simultaneously, the MT version gains performance by bringing the data less times from the memory to the cache hierarchy. Such a design can help applications with a larger than L2 cache data set and even achieve higher than 2x performance improvement.

Another scenario is when one thread generates the data and the other thread consumes it. A couple of variations of this scenario are possible and are further explained in the "Producer Consumer Models," Section 5.3 of the *Intel® Core™ Duo Processor Optimization Guide*[4]. Briefly, they are the "Delay" approach and "Symmetric" approach. Below is an example of the expected speedup when the producer-consumer model is run on an Intel Core Duo processor vs. a Dual Core Intel® Xeon® processor vs. an Intel Pentium 4 processor with Hyper-Threading Technology[1] (W = Write, R = Read, xxK = buffer size).

Not only do these data show the benefit of avoiding the bus/memory latency, they also demonstrate how varying multi-processor implementations behave in both code affinity (functional) decomposition and data affinity (data) decomposition threading models. If the produced/consumed data set size is bigger than the L1 data cache size, yet smaller than the L2 cache size, data decomposition and functional decomposition yield similar performance (assuming the functional decomposition implementation is well balanced), and the best performance that can be achieved for data sharing.

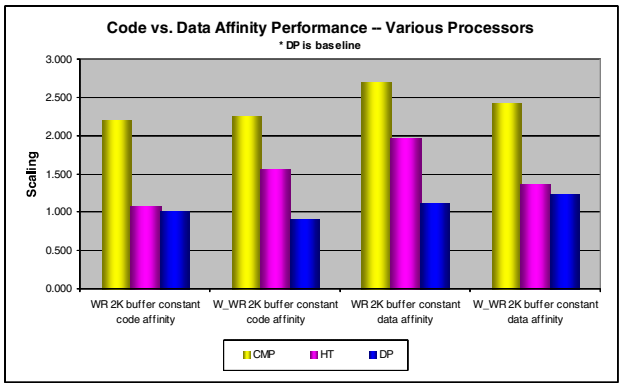


Figure 9: Code vs. data affinity performance on various processors

False Sharing Can Reduce Performance

False sharing happens when two or more threads access different address ranges on the same cache line simultaneously. This causes the cache line to be in the first level cache of the two cores.

False sharing causes a severe performance penalty if one or more of the threads writes to the shared cache line. This causes invalidation of the cache line at the first-level cache of the other core. As a result, the next time that the other core accesses the cache line in question it will have to transfer it from the core that wrote it earlier through the bus, thereby incurring a major latency penalty.

Below is an example of code that has false sharing when executed by several threads simultaneously.

```
int counter[THREAD_NUM];
int inc_counter ()
{
    counter[my_tid]++;
    return counter[my_tid];
}
```

Table 1 lists the penalties that an application can suffer if it uses false sharing intensively on an Intel Core Duo system. In order to avoid such an unnecessary overhead, the programmer needs to avoid false sharing, and in particular, needs to make sure it does not occur unintentionally in the following cases:

[1] Hyper-Threading Technology requires a computer system with an Intel® Pentium® 4 processor supporting HT Technology and a HT Technology enabled chipset, BIOS and operating system. Performance will vary depending on the specific hardware and software you use. See www.intel.com/products/ht/Hyperthreading_more.htm for additional information.

- Global data variables and static data variables that are placed in the same cache line but are written by different threads.
- Objects allocated dynamically by different threads can accidentally share cache lines.

Table 1: False sharing penalties

Case	Data location	Latency (cycles/nsec)
L1 to L1 Cache	L1 Cache	14 core cycles + 5.5 bus cycles
Through L2 Cache	L2 Cache	14 core cycles
Through Memory	Main memory	14 core cycles + 5.5 bus cycles + ~40-80 nsec depending on FSB and DDR freq.

Optimize Bus Access Between the Cores to Maximize the Bus Bandwidth

Be careful when parallelizing code sections that use data sets exceeding the second-level cache and/or bus bandwidth. If only one of the threads is using the second-level cache and/or bus, then it is expected to get the maximum possible speedup when the other thread running on the other core does not interrupt its progress. However, if the two threads use the second-level cache there may be performance degradation if one of the following conditions is true:

- Their combined data set is greater than the second-level cache size.
- Their combined bus usage is greater than bus capacity.
- They both have extensive access to the same set in the second-level cache, and at least one of the threads writes to this cache line.

To avoid these, we recommend that you investigate parallelism schemes in which only one of the threads accesses the second-level cache at a time, or that the level of using the second-level cache and the bus does not exceed their limits. This concept is explained further in Section 5.3.5 of the *Intel® Core™ Duo Processor Optimization Guide*.

CONCLUSION AND REMARKS

The full performance potential of the Intel® Centrino® Duo mobile technology architecture can be realized by efficiently multi-threading applications, using the methods detailed in this paper. The use of balanced threading techniques is likely to provide optimal performance improvements on CMP. Multi-tasking scenarios, one of the common usage scenarios, provide a richer user experience on the Intel Centrino Duo mobile technology system. The shared cache structure is likely to showcase better performance scaling for MT applications when the threads work on shared data sets. Avoid using false sharing which impacts performance scaling. The optimization guide mentioned earlier provides a detailed explanation of these and other performance optimization techniques.

REFERENCES

[1] Gochman et Al., "Introduction to Intel® Core™ Duo Processor Architecture," in *Intel Technology Journal, Volume 10, Issue 2, 2006.*

[2] Naveh et al., "Power and Thermal Control in the Intel® Core™ Duo Processor," in *Intel Technology Journal, Volume 10, Issue 2, 2006.*

[3] *IA-32 Intel® Architecture Software Developer's Manual Volume 3A*: *System Programming Guide, Part 1*, in ftp://download.intel.com/design/Pentium4/manuals/25366819.pdf

[4] IA-32 Intel® Architecture Optimization Reference Manual, in http://www.intel.com/design/Pentium4/manuals/248966.htm

AUTHORS' BIOGRAPHIES

Avi Mendelson is a principal engineer in Intel's Mobile Platform Group in Haifa, Israel, and adjunct professor in the CS and EE departments, Technion, Israel Institute of Technology. He received his B.Sc. and M.S.c degrees from the Technion, Israel Institute of Technology and his Ph.D from the University of Massachusetts Amherst. Avi has been with Intel for 7 years. He started as senior researcher in Intel Labs, later he moved to the Microprocessor group where he serves as the CMP architect of Intel Core Duo processor. Avi's work and research interests are in computer architecture, low-power design, parallel systems, OS related issues and virtualization. His e-mail address is avi.mendelson at intel.com.

Julius Mandelblat is a principal engineer in Intel's Mobile Platform Group in Haifa, Israel. He received his B.Sc. and M.Sc. degrees from Transport Engineers Institute in Moscow (USSR). Julius joined Intel 16 years ago. He worked on many of the processors that have been developed by Israel Design Center during this period. Julius worked as micro-architect and senior design leader for the CMP implementation of the Core Duo processor. His e-mail address is julius.mandelblat at intel.com.

Simcha Gochman is a senior principal engineer with Intel's Mobile Platform Group in Haifa, Israel. Simcha has been with Intel for 21 years. Lately he was leading the microarchitecture development of the Pentium M processors Banias and Dothan and of the Core Duo processor code name Yonah. Earlier in Intel he led the microarchitecture definition of the Pentium Processor with MMX™ technology and was involved with the design of the 80860 processor and the 80387 numeric coprocessor. Simcha received his M.Sc. degree from the Technion, Israel Institute of Technology in 1984. His e-mail address is simcha.gochmana at intel.com.

Anat Shemer is a senior software engineer working on mobile software enabling in the Software Solutions Group. Anat has been with Intel for 16 years. She worked on performance analysis of multi-threaded workloads supporting the evaluation of Intel Core Duo micro-architecture performance. Earlier at Intel she worked on binary tools and performance simulation tools that supported Intel future micro-architectures development. Anat received here M.Sc. degree from the Technion, Israel Institute of Technology in 1989. Her e-mail address is anat.shemer at intel.com.

Rajshree Chabukswar is a software engineer working on client enabling in the Software Solutions Group that enables client platforms through software optimizations focusing on multi-threading and mobile enabling. Prior to working at Intel, she obtained a Masters degree in Computer Engineering from Syracuse University, NY. Her e-mail address is rajshree.a.chabukswar at intel.com.

Erik Niemeyer is a senior software engineer working for Intel's Software and Solutions Group. His current assignment is in New Mexico with the Mobile Enabling Client Team working on IBM-Lotus Notes* to help improve performance and reduce power consumption of the upcoming R8 release. Erik has been with Intel for 6 years. Before Intel, Erik was a systems integrator/programmer with the Federal Government for 12 years and worked on enterprise-level application performance tuning. Prior to that Erik earned his B. Sc. from the University of New Mexico. His e-mail address is erik.a.niemeyer at intel.com.

Arun Kumar is an engineering manager in Intel's Software and Solutions Group in Dupont, Washington. He received his B.Tech degree from the Indian Institute of Technology, Kanpur, and his Masters and PhD degrees from the University of Minnesota, Minneapolis, where his research was in the area of image processing and computer vision. Currently, his team works on client platforms based on Pentium-M, Pentium-D and Core Duo processor architectures with special focus on CMP, application software performance and platform power consumption. His e-mail address is arun.kumar at intel.com.

Copyright © Intel Corporation 2006. All rights reserved. Intel, Core, Pentium, Intel SpeedStep, Xeon, Centrino, and MMX are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

* Other names and brands may be claimed as the property of others.

This publication was downloaded from http://developer.intel.com/.

Legal notices at http://www.intel.com/sites/corporate/tradmarx.htm.

THIS PAGE INTENTIONALLY LEFT BLANK

Power and Thermal Management in the Intel® Core™ Duo Processor

Alon Naveh, Mobility Group, Intel Corporation
Efraim Rotem, Mobility Group, Intel Corporation
Avi Mendelson, Mobility Group, Intel Corporation
Simcha Gochman, Mobility Group, Intel Corporation
Rajshree Chabukswar, Software Solutions Group, Intel Corporation
Karthik Krishnan, Software Solutions Group, Intel Corporation
Arun Kumar, Software Solutions Group, Intel Corporation

Index words: Intel Core Duo, ACPI, power, thermal

ABSTRACT

The Intel® Core™ Duo processor is the first mobile processor that uses Chip Multi-Processing (CMP) on-die technology to maximize performance and minimize power consumption. This paper provides an overview of the power control unit and its impact on the power-saving mechanisms. We describe the distribution of P-states and look at other techniques used to improve the system power consumption including the impact of multi-threading on power consumption. Then, we provide recommendations on optimizing for CMP and building power-friendly applications.

INTRODUCTION

Power and thermal management are becoming more challenging than ever before in all segments of computer-based systems. While in the server domain, the cost of electricity drives the need for low-power systems, in the mobility market, we are more concerned about battery life and thermal limitations. The increasing demand for computational power in conjunction with the relatively slow improvement in cooling technology caused power and thermal control methods to become primary parts of the architecture and the design process of the Intel Core Duo processor-based system.

Intel Core Duo is the first general-purpose, multi-core on die Chip Multi-Processing (CMP) processor Intel has developed for the mobile market. The core was designed to achieve two main goals: (1) maximize the performance under the thermal limitation the platform allows; and (2) improve the battery life of the system relative to previous generations of processors.

Comparing the Intel Core Duo processors with previous-generation Pentium® M processors [1] reveals that it uses newer process technology, runs faster, and uses the same-size caches, but the overall die size is increased in order to accommodate two cores. The use of new process technology, together with special design and layout optimizations, allows each core of the Intel Core Duo technology to control its dynamic power consumption. In this paper, therefore, we focus on the CMP-related aspects of this technology. Controlling the static power (leakage) was found to be a real challenge due to the fact that static power depends on the total area of the processor and on process technology. Since the overall area was increased and the new process was found to produce more leakage power, static and dynamic power management became one of the main issues when designing the system.

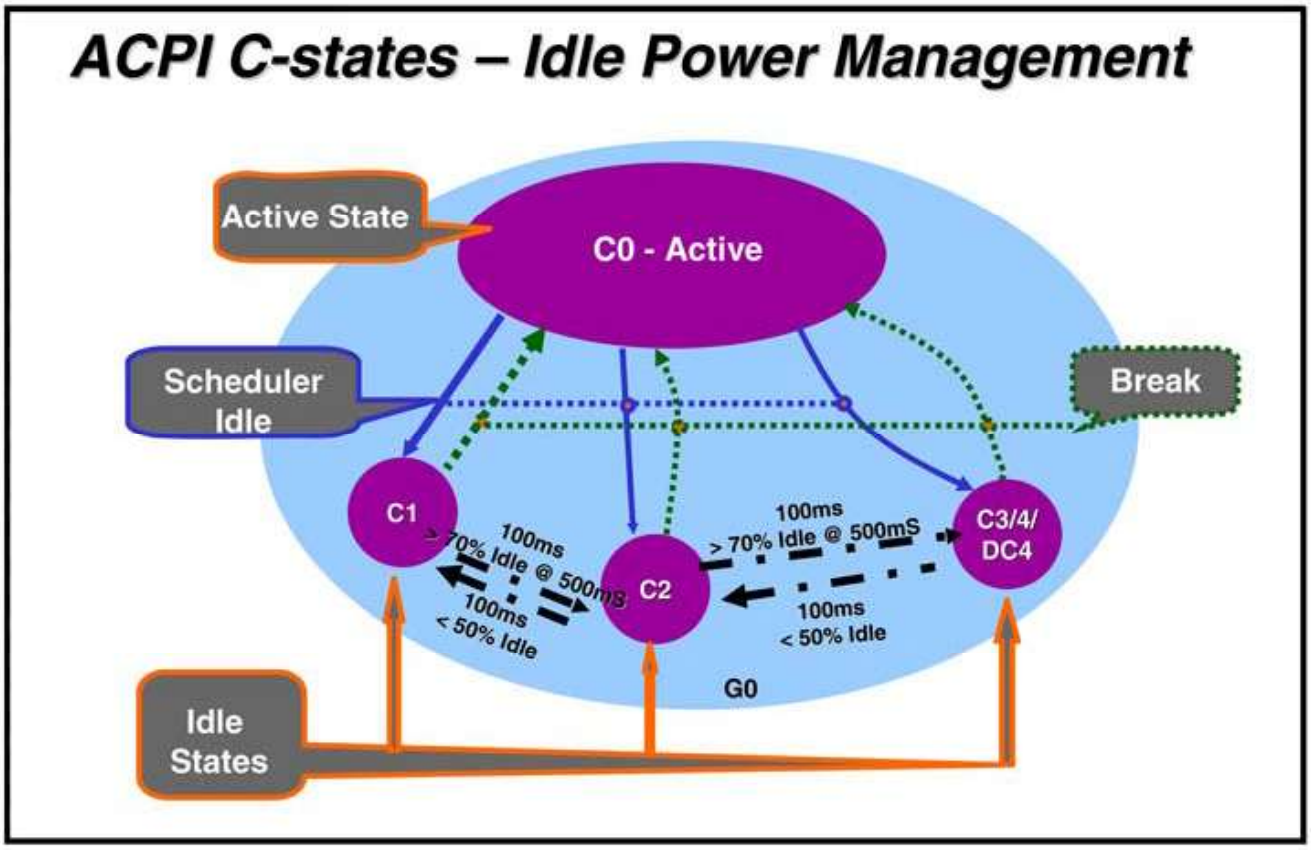


Figure 1: ACPI-based idle state management

Optimizing the system for maximum performance at the minimum power consumption is usually done as a combination of software (operating system) and hardware elements. Most modern operating systems (OS) use the ACPI [2] standard for optimizing the system in these areas. The ACPI processor sleep state control assumes that the core can be in different power-saving states (also termed sleep states) marked as C_0 to C_n. When the core is active, it always runs at C_0, but when the core is idle, the OS tries to maintain a balance between the amount of power it can save and the overhead of entering and exiting to/from that sleep state. Thus C_1 represents the power state that has the least power saving but can be switched on and off almost immediately, while extended deep-sleep (DC_4) represents a power state where the static power consumption is negligible, but the time to enter into this state and respond to activity (back to C_0) is quite long. Figure 1 shows the general principle of operation of how a traditional ACPI software layer controls the sleep states of the processor. When the OS scheduler detects there are no more tasks to run, it transitions the processor into the "selected" idle state, by executing a BIOS defined instruction sequence. The processor will remain in that idle state until a break event occurs and then return to the C_0 state. Break events would typically be interrupts and similar indicators that new tasks need to be executed. The OS evaluates the CPU activity factor over a registry-based time window and, periodically, modifies the selected target C-state for the next evaluation window.

The progressive increase in static power savings per state is achieved by employing more aggressive means as the C-state deepens. In C_1 idle state, only processor-centric measures are employed: instruction execution is halted and the core clocks are gated. In C_2 states and above, platform-level measures are also added to further increase the power savings. While in the C_2 state, the processor is obligated not to access the bus without chipset consent. Thus, the front side bus can be placed in a lower power state, and the chipset can also initiate power-saving measures. In C_3, the processor also disables its internal Phase Locked Loops. In C_4, the processor also lowers its internal voltage level to the point where only content retention is possible, but no operations can be done. Deep-C4, a new Intel Core Duo power state, is described later in this paper.

In order to control the dynamic power consumption, the ACPI standard tries to adjust the active power consumption to the "computational" needs of the software. The system controls the "active" state of the system (also termed P-states) by a combination of hardware and software interfaces: the hardware defines a set of "operational points" where each one defines a different frequency/voltage and therefore different levels of power consumption. The OS aims to keep the system at the lowest operational point yet still meet the performance requirements. In other words, if it anticipates that the software can efficiently use the maximum performance that the processor can provide, it puts it in the P_0 state. When the OS predicts (based on history) that the lower (and slower) operational point can still meet the performance requirements, it switches to that operational point (marked as Pn) in order to save power.

Power consumption produces heat that needs to be removed from the system. Naturally, each system has a total cooling capacity per its specific design, and each major component has a cooling limit or power consumption allowance. The system designers usually choose the maximum frequency the system can run at without overheating when running in a "normal" usage model; this is also called the system's Thermal Design Power (TDP). Thus, when an unexpected workload is used the thermal control logic aims to allow the system to provide maximum performance under the thermal constraints.

The remainder of this paper is organized as follows: we first describe the Intel Core Duo implementation of power and thermal control; next, we provide experimental numbers that examine the efficiency of the power and thermal mechanisms; and finally we extend the discussion to software optimizations that can help save power.

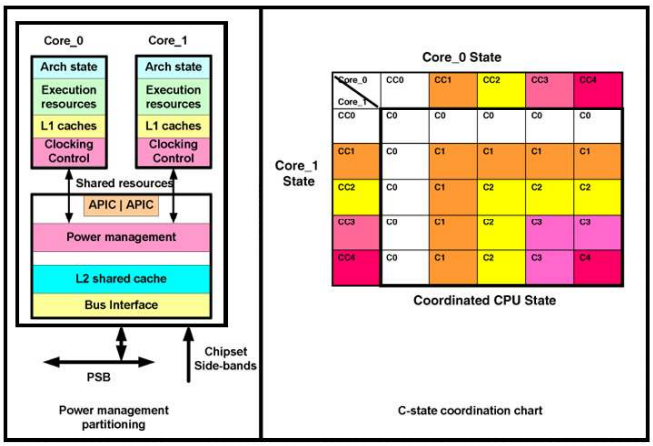


Figure 2: Internal power state in Intel Core Duo processor

POWER AND THERMAL MANAGEMENT

Power and thermal management of the Intel Core Duo processor raise a new challenge to the ACPI-based architecture since, from a software point of view, there is no difference between running under a CMP-based system and running under dual-core architecture, but from a hardware point of view, even though most of the logic is duplicated, major parts of the logic such as the power supply and the L2 cache are still shared. For example, in a Dual-Processor system, when the OS decides to reduce the frequency of a single core, the other core can still run at full speed. In the Intel Core Duo system, however, lowering the frequency to one core slows down the other core as well.

C-state Architecture

Since the OS views the Intel Core Duo processor as two independent execution units, and the platform views the whole processor as a single entity for all power-management related activities (C2 state and beyond), we chose to separate the core C-state control mechanism from that of the full CPU and platform C-state control.

This was achieved by making the power and thermal control unit part of the core logic and not part of the chipset as before. Migration of the power and thermal management flow into the processor allows us to use a hardware coordination mechanism in which each core can request any C-state it wishes, thus allowing for individual core savings to be maximized. The CPU C-state is determined and entered based on the lowest common denominator of both cores' requests, portraying a single CPU entity to the chipset power management hardware and flows. Thus, software can manage each core

independently, while the actual power management adheres to the platform and CPU shared resource restrictions.

As can be seen from Figure 2, the Intel Core Duo processor is partitioned into three domains. The cores, their respective Level-1 caches, and the local thermal management logic each operates as a power management domain. The shared resources–including the Level-2 cache, bus interface, and interrupt controllers (APICs)–are yet another power management domain. All domains share a single power plane and a single-core PLL, thus operating at the same frequency and voltage levels. However, each of the domains has an independent clock distribution (spine). The core spines can be gated independently, allowing the most basic per core power savings (C1-Halt state). The shared-resource spine is gated only when both cores are idle and no shared operations (bus transactions, cache accesses) are taking place. If needed, the shared-resource clock can be kept active even when both cores' clocks are halted, serving L2 snoops and APIC message analysis.

The coordination mechanism serves as a transparent layer between the individually controlled cores and the shared resource on die and on the platform. It determines the required CPU C-state, based on both cores' individual requests, controls the state of the shared resources, such as the shared clock distribution, and also implements the C-state entry protocol with the chipset, emulating a legacy single-CPU mobile processor. When detecting that both core states are deeper than C1, the coordination mechanism issues the proper indication to the chipset, triggering the platform C2-4 entry sequence.

When the platform detects a break event (such as an interrupt request), it negates the proper sideband signals (such as the STPCLK, SLP, DPSLP and DPRSTP). The coordination logic then analyzes the platform signaling, and initiates the proper C-state exit sequences for the shared resources, and if needed, the cores.

Thus, the required goal of coordinating across the two cores is achieved in an efficient and transparent manner. Software operates as if it is managing two independent cores, while the platform and the shared resources are controlled as one by the coordination logic, reflecting a single platform Level C-state in a backwards-compatible manner.

As part of the per core power savings, the independent cores' L1 cache is also flushed during core C3 and C4 states. Due to the ratio between L1 and L2 cache size, it is assumed that nearly all of the L1 is included in the L2. Therefore, the flushing should not incur a high overhead of a write-back into the system memory, and it will not incur a high warm-up penalty when restarting execution afterwards, since the data already reside nearby in the L2 cache. By flushing the caches, the cores can be kept asleep even when the L2 cache is accessed heavily by the other core or by a system device, thus improving power savings even further.

Dynamic Cache Sizing and Deep C4 State

Now that the processor has been able to enter C4, we face the challenge of lowering the C4-state leakage power even further. Since leakage power is directly proportional to the operating voltage, the most efficient means to save leakage static power would be to lower the C4 operating point. Unfortunately, lowering the voltage impacts data retention, and the first cells to be affected are the small transistor data arrays such as in the L2 cache. Therefore, the first step in achieving a lower voltage idle state is to implement a mechanism that can dynamically shut down the L2 cache in preparation for the Deep C4 state.

Cache Sizing

When defining the L2 cache dynamic sizing algorithm, the following considerations need to be addressed:

- L2 is a large array; therefore flushing it will incur some power and potentially C-state latencies, especially if done all at once or too frequently.
- Many applications will suffer performance degradation if running for a long period of time with little or no L2 cache. However, it has been proven that the short interrupt handling tasks, occurring at periodic timer ticks, do not have much use for the L2 cache and are not visibly impacted by running with the cache closed.

To accommodate the above restrictions, the dynamic sizing mechanism is implemented as an adaptive algorithm, with various built-in filtering properties and heuristics.

At the algorithms' base is an assumption that during long periods of very low utilization and idle residency (mapping to the C4 state), shutting down the cache will not result in a perceivable performance impact. This condition is detected by a state machine, described in Figure 3. In order to start shrinking the cache, the Finite State Machine (FSM) checks that the CPU frequency, controlled by the OS, is below a programmable threshold. It also checks that the CPU on the whole has not stayed too long in C0–which may indicate a streaming task being executed (e.g., DVD playback). This is done by a pre-programmed count-down timer, reloaded once the whole CPU enters C2-4 (package) states. Finally, the FSM checks the second core's idle state, requiring it to be in C4 as well, before allowing a shrink operation to begin.

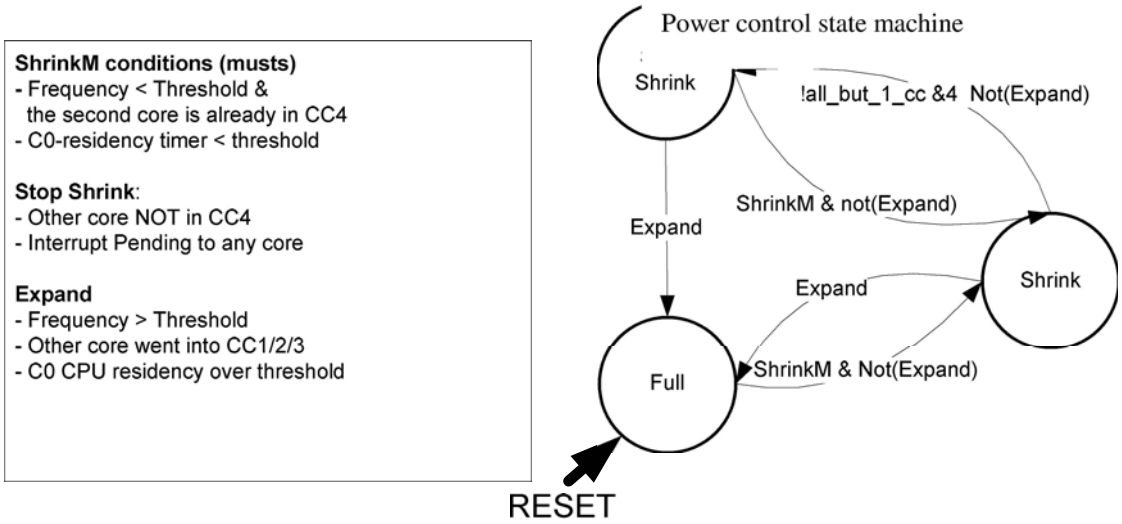


Figure 3: Shrink/Expand heuristics

The FSM freezes the shrink operation if a pending interrupt is detected, or if either core is in the active C0 state.

Cache expansion is requested once the activity indicators show that performance may be required. This is inferred in one of three ways: either the frequency is being increased over the programmable threshold, the CPU is staying in C0 for a period exceeding the pre-programmed timer, or one of the cores is entering a non C4 idle state (this is the OS's way of signaling that the core was not idle enough).

The actual cache flushing flow is performed by the microcode of the last core entering C4-state. In order to minimize the power impact and to filter too short C4 periods, the microcode flushes only part of the L2 (1/8 through 1/2 of the total cache size) during each consecutive C4 entry. The cache is flushed in chunks of lines (between 4 to 256 lines) checking for interrupts in between. Once a whole way is flushed, it is power-gated with sleep transistors, further reducing its leakage.

Microcode typically automatically expands the cache to a minimum of two ways upon every DeepC4 exit. Once the CPU enters C4 again, microcode will shrink the cache back down to 0. At the initial expansion it is assumed that the CPU has just exited from DeepC4. As such the L2 valid array may not be valid. Therefore, as part of the DeepC4 exit flow, the microcode also clears all of the L2 valid bits, ensuring the cache is indeed perceived as empty by the snoop logic. The same initialization flow can be applied also to other sensitive arrays should testing detect them as unstable.

DeepC4 (DC4) Entry

After the L2 cache has been shrunk to 0 and the CPU enters C4, the CPU voltage may be further reduced. Moreover, since no data are cached, the data cache does not need to be awakened for snoops. This feature is performed by the chipset, during the DC4 state. Once this is detected, the chipset starts diverting snoopable traffic directly to memory. During the DC4 state, the chipset scans the incoming traffic for interrupts and APIC messages, and once they are detected, queues them separately, while initiating a break sequence for the processor. Once the processor is fully awake, the interrupts are delivered to the processor and the memory traffic is diverted back to the CPU for snooping.

Handling P-states in a CMP Environment

ACPI P-state (Performance) control algorithm's goal is to optimize the runtime power consumption without significantly impacting performance. The algorithm dynamically adjusts the processor frequency such that it is just high enough to service the SW execution load. Operating point selection is done by the OS power management algorithms (OSPM) based on the CPU load observed over a window of time. Once the target point is set, the CPU is expected to modify its operating voltage and frequency to match the OSPM's request.

Figure 4 shows one example of the relationship between different working points (P-state points) in the Intel Core Duo processor and their relative power consumption. As can be seen, the benefit of going to a lower working point can be divided into an "exponential" part and a "linear" part. The exponential part represents a range of operating points where both frequency and voltage can be scaled to meet the new working point, while the linear part

represents a range of operating points where the system has already reached the minimum voltage allowed and so only the frequency can be scaled.

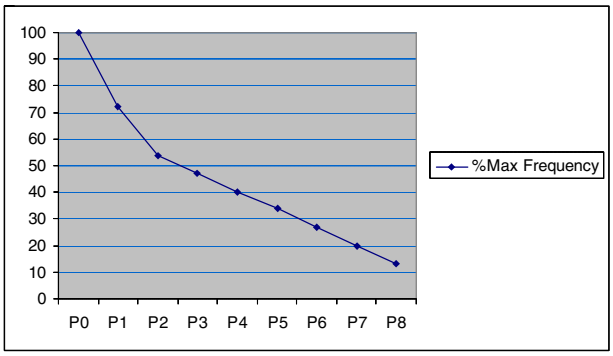


Figure 4: P-state CPU frequencies

When a multi-processor system was used, each core could be controlled independently by the OS and the individual processor state was set accordingly. The Intel Core Duo CMP implementation presents new challenges since some parts of the system, such as the PLLs and power planes, are shared and so both cores and the shared area need to operate at the same effective frequency/voltage point (i.e., the same P-state).

Initial SW versions that ran on the hyper-threading Intel NetBurst® microarchitecture-based CPUs utilized System Management Mode (SMM) code to coordinate the target P-state between the threads. This solution was targeted to optimize performance, which suited desktops and servers at which most hyper-threading CPUs were targeted. However, it incurred inherent overhead for every P-state request made by the OSPM, resulting in an average power impact prohibitive to the mobile focused Intel Core Duo architecture.

Consequently, a HW coordination approach was adopted for the P-state architecture as well. As with the C-state mechanism, each core's OS Power Management component can request a P-state separately via the standard IA32_PERF_CONTROL MSR. The HW coordination logic in the shared region tracks these P-state requests from both cores and determines the required CPU level target operating point based on the current execution state of the CPU:

- No thermal control state: In this case, the coordination logic will prefer performance over power, so as not to starve the SW threads. As a result, the higher operating point will be selected for the total CPU operating point, causing the whole CPU to execute the known advanced Intel SpeedStep® technology-based architecture transitions between operating points.
- Thermal_Controlled_State: In this case, the HW has detected an overheating condition and is trying to drive the operating point down for cooling. Here, the coordination HW will select the lower of the two cores' requests, allowing a quicker cooldown than if the maximum of both cores was used. Initial Intel Core Duo CPUs will use the same operation point for both cores; however, future implementations may choose to select a different operation point per core, thus taking advantage of this capability.

Due to the hardware coordination nature of the P-state architecture, OSPM may have a hard time tracking the exact frequency each core has run at, since the actual runtime frequency may be higher than the OSPM requested, and may vary without the core's OSPM knowledge. This may cause some confusion to the OSPM when trying to use the core's activity factor (non C0 state %) and the expected frequency to determine the actual code load. Intel Core Duo technology augments the previous frequency visibility mechanism by supplying two 64-bit counters, providing the OSPM information on the actual execution frequency of each core. ACNT (ActualCount) counts executed clocks (not Idle) at the currently coordinated CPU frequency. MCNT (MaxCount) counts the maximum number of non-idle clocks that the core could have run at, during the same period of time, had it run at the nominal frequency defined for the part. By dividing the ACNT by MCNT, the OSPM can determine the actual frequency each core has run at over a window of time, assisting the OSMP to assess the correct execution load on that core and then in turn determine the next operating point.

Thermal Control in a CMP Environment

The Intel Core Duo system was designed with power efficiency in mind and is aimed at different form factors, some of which are power and thermally constrained. Thermal management is a fundamental capability of all mobile platforms. Managing platform thermals enables us to achieve the maximum CPU and platform performance within thermal constraints. Thermal management features also improve ergonomics with cooler systems and lower fan acoustic noise.

In order to better control the thermal conditions of the system, Intel Core Duo technology presents two new concepts: the use of digital sensors for high accuracy die temperature measurements and dual-core multiple-level thermal control.

The general structure of the digital thermometer is described in Figure 5. It supports the legacy Intel® Centrino® mobile technology thermal sensor, PROCHOT and THERMTRIP, and the adding of a temperature reading capability to the fixed threshold sensor. A control

logic built around the thermal sensor performs a periodic scan and generates an output that represents the current temperature reading. The reading is loaded into an MSR, accessible to software. In order to improve temperature reading, multiple sense points monitor different hot spots on the die and report the maximum temperature of the die. An independent temperature reading from each core is available, with optional reporting of the maximum temperature of the entire die, e.g., the highest temperature of both cores.

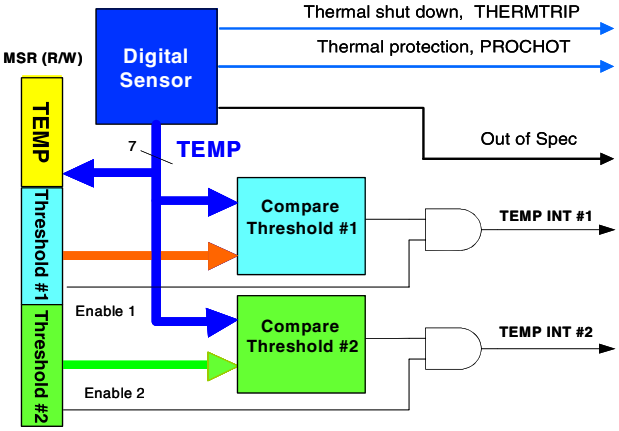


Figure 5: Digital thermometer block diagram

Interrupt generation capability is provided in addition to the temperature reading. Two programmable thresholds are loaded by S/W and a thermal event is generated upon threshold crossing. This thermal event generates an interrupt to a single core or to both cores simultaneously through the APIC settings.

The digital thermometer is intended to be used as an input to a software-based thermal control such as the ACPI. An interrupt threshold is defined to indicate the upper and lower temperature thresholds. An example of digital thermometer usage is illustrated in Figure 6.

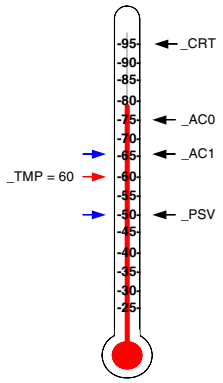


Figure 6: Digital thermometer and ACPI

In the above example, the die temperature is at 60°C and the thresholds are set to 50°C and 65°C. If the temperature rises above 65°C, a low-to-high interrupt is generated. The control software identifies the new temperature and initiates action, such as activating fans or initiating some passive cooling policy. The activation thresholds and policies are defined using BIOS and ACPI. New thresholds are loaded around the new temperature to further track new changes.

In previous Intel Pentium M processor-based systems, a single analog thermal diode was used to measure die temperature. A thermal diode cannot be located at the hottest spot of the die and therefore some offset was applied to keep the CPU within specifications. For these systems it was sufficient, since the die had a single hot spot. In the Intel Core Duo system, there is more than a single hot spot that moves as a function of the combined workload of both cores. Figure 7 shows the differences between the usage of the traditional analog sensor and the new digital sensors.

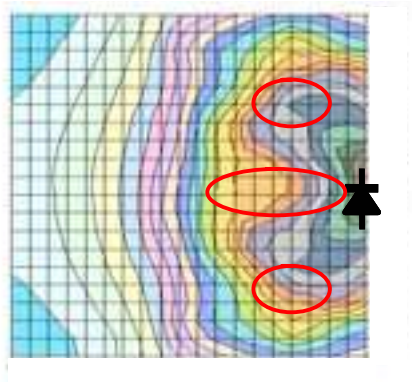


Figure 7: Analog vs. digital sensors in Intel Core Duo systems

As we can see, the use of multiple sensing points provides high accuracy and close proximity to the hot spot at any time. An analog thermal diode is still available on the Intel Core Duo processor. An example of the difference between the diode and the DTS is presented in Figure 8. The information presented in this graph was taken from simulations and not from a real system, but was correletaed with data from real systems and found to be close enough.

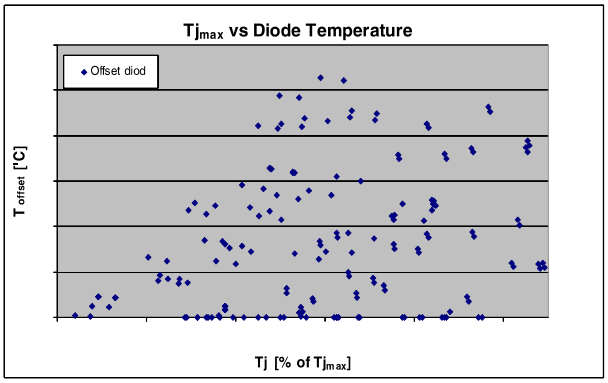


Figure 8: Diode to DTS temperature difference

It can be seen that significant temperature gradients exist on the die. The use of a digital thermometer provides improved temperature readings, enables higher CPU performance within thermal limitations, and improves reliability.

The Intel Core Duo system also implements hardware-based thermal control. Hardware-based thermal management is intended to handle abnormal thermal conditions and to protect the die from transient effects. Hardware-based thermal control ensures that the CPU will always operate within specified conditions. This improves reliability and allows higher performance with tighter control parameters.

Legacy thermal control features implement two externally visible signals:

- THERMTRIP: a fixed temperature sensor to detect catastrophic thermal conditions and to shut down the system if thermal runaway occurs.
- PROCHOT: a fixed temperature threshold that provides the DVS with a self-control mechanism that drops frequency and voltage to a new working point (a more detailed description of this mechanism can be found in [1]).

Intel Core Duo technology implements a new multiple-core thermal control algorithm. Both cores synchronize action requests and activity. A programmable policy can select DVS on both cores and linear control on each core either independently or as a locked operation. PROCHOT was also made available as an input, to allow thermal protection of platform components such as the voltage regulator (VR).

A finer grain overheat detection is performed by monitoring the thermal activity. If an extreme thermal condition occurs (fan malfunction, operation within a bag, etc.) the processor self thermal management may not be sufficient and the temperature may not drop below the threshold point. The internal control algorithm tracks the control behavior and further reduces power as needed. Continuous extreme conditions will eventually initiate an "Out Of Spec" thermal interrupt and register the warning in an internal status bit. The "Out Of Spec warning" signal is intended to initiate a managed system shutdown before a THERMTRIP shutdown occurs.

Platform Power Optimization

Intel Core Duo technology implemented power and thermal management features as described above to control the CPU power and thermals. Other components on the platform also contribute to the total platform power and work closely with the CPU. The CPU VR losses can get as high as 5W while running high workloads. A 3-phase VR was built to deliver high current operating at low efficiency while working at low utilization. Turning off phases or switching to asynchronous mode can save a significant amount of power. Efficiency, as a function of the number of phases, is described in Figure 9.

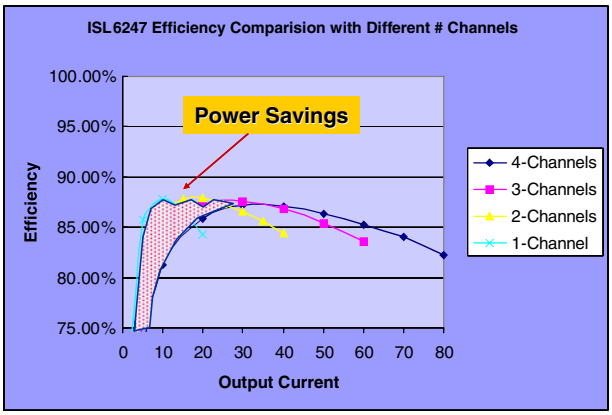


Figure 9: Efficiency as a function of number of phases

It can be noted that improved power efficiency mode cannot handle high currents and there is a need for feedback on the current requirements. The Intel Core Duo processor keeps track of the power consumption and gives feedback to the VR using the PSI-2 and VID signals, as described in Figure 10.

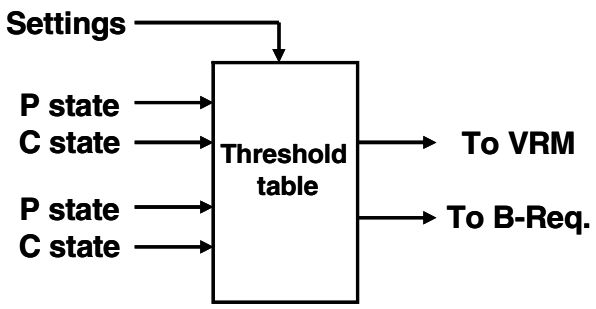


Figure 10: PSI-2 overview

The CPU tracks the activity requirements based on P- and C-states. When the OS initiates a request to go to a higher P-state, there is sufficient time to signal the VR to change to high current mode before the actual power consumption increases.

Another optimization at low workloads is done on the load line. The voltage delivery has serial resistance, and a voltage drop between the VR and the CPU is a function of the current consumption. At lower power consumption, the voltage drop is lower and the CPU voltage increases, driving higher power. The same mechanism on the CPU detects low power consumption and drives lower VID code to the VR. Voltage is increased ahead of time, before actual power consumption increases.

Communicating the CPU requirements to the rest of the platform enables additional power savings on the platform, increasing the battery life.

EXPERIMENT RESULTS OF POWER MANAGEMENT

In this section, we provide measurements that were taken from an Intel Core Duo processor-based system in order to examine the efficiency of its power and performance. We mainly focus on the relation between the use of MT applications and the utilization of the Intel Core Duo system.

Distribution of P-states

The OS along with the hardware optimizes the platform power by decreasing the processor speed when there is no demand and increasing the frequency as demand increases. Figure 11 shows a typical snapshot of the system power consumption during some period of time. As can be seen, the OS sets both processors to a single power state (P-state), but the total power consumption depends on the number of tasks run in parallel (if one task is run, we assume it is run in an ST environment) and the computational time of these tasks. We can clearly observe the phases of execution: the length of each phase reflects the total time it takes while the area below the curve reflects the total energy for this phase. We use this energy calculation methodology in the following sections to compare ST and MT applications.

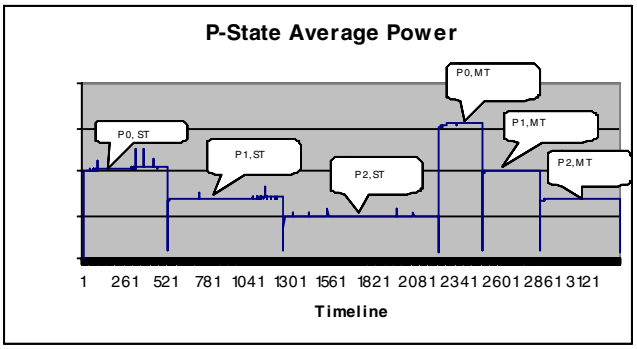


Figure 11: Energy calculation methodology

Power Consumption of Different Power States

In order to evaluate the benefit of using MT applications, we measure the power consumption of the system while in different working points when running an ST application vs. an MT application. Figure 12 shows the experimental data with both ST and MT code under different P states on an Intel Core Duo system. As can be seen, the power consumed by an MT code is less than twice the power consumed by an ST code in corresponding P states. This implies that the applications that are MT and demonstrate excellent scaling will not only reduce the total execution time, but also show greater savings in power consumed.

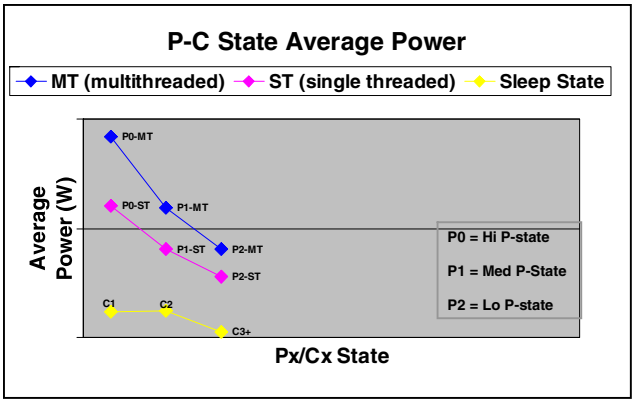


Figure 12: P/C-state average power

Impact of Clock Rate on Power Consumption

The OS uses a periodic clock interrupt to keep track of time, trigger timer objects, and schedule application threads. While the default interrupt rate is set by the OS, applications can increase the interrupt rate to any desired frequency (as low as 1ms).

While some multimedia applications with high-definition content may require a high interrupt rate for timely

scheduling of timer-based threads, some with enough processing headroom may not require it. Thus, in both cases, increasing the clock interrupt rate may have an important impact on the overall power (and therefore on the battery life).

Figure 13 shows the processor power impact due to a high interrupt rate on an Intel Core Duo system. Average power increases from 0.5 W to 1.05 W when the interrupt frequency is increased to 1ms.

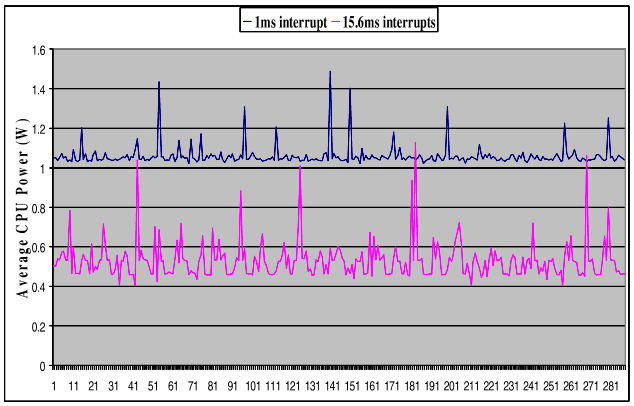


Figure 13: High interrupt rate impact

As can be seen, the "baseline" of the 1ms curve is higher than the normal interrupt rate. The reason for that is that every interrupt causes the CPU to go back from C3 to C0 (active state) and when the CPU utilization is high enough, it will also execute the clock service routine in high P-state.

Balance and Imbalance Threading Environment

This section includes a comparative study of the total energy consumed on the Intel Core Duo platform (CPU plus platform) for a given task with both MT and ST methodologies. We shall also analyze applications featuring various threading scenarios, such as "balanced threading" vs. "imbalanced threading."

Applications with a Balanced Threading Model

Real-world applications under study here are CPU bound workloads that were run in both ST and MT mode. The applications are CPU bound, but in ST mode, only one core is fully utilized, and in MT mode both the cores are fully utilized and consume equal processing resources. Adaptive mode here refers to the power-saving scheme where the OS optimizes overall power consumption by dynamically changing CPU frequency on demand using Intel SpeedStep technology (the GV3 technology).

The graphs (Figures 14 and 15) below indicate CPU and platform energy for adaptive mode. For each application run (ST and MT), power data gathering is normalized to the longest run-time. For example, a cryptography workload runs for ~50 seconds in ST mode and ~25 seconds in MT modes. The power data is measured for 50 seconds in both ST and MT mode. The intent here is to identify if total energy can be saved by finishing the task faster (as in the case of MT) and entering deep sleep states after task completion.

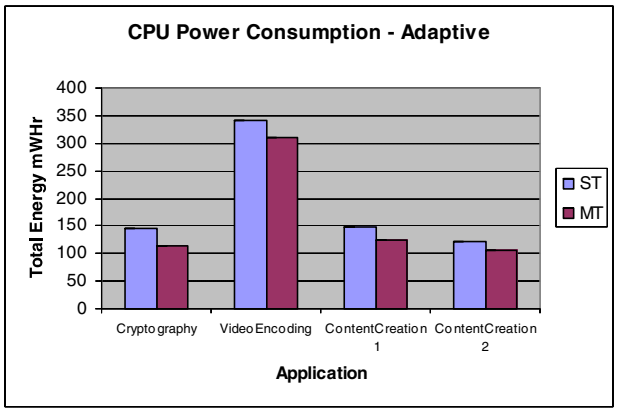


Figure 14: Balanced threading–CPU power

As indicated in Figure 14, the energy for a given task is reduced by completing the task sooner with efficient MT and letting the processor enter idle/sleep state. The MT applications above show linear performance scaling with the number of cores and demonstrate a ~9%-27% savings in CPU power.

The CPU power is a major component of the overall power, and the graph in Figure 15 indicates about an 8%-17% savings in the platform power due to MT.

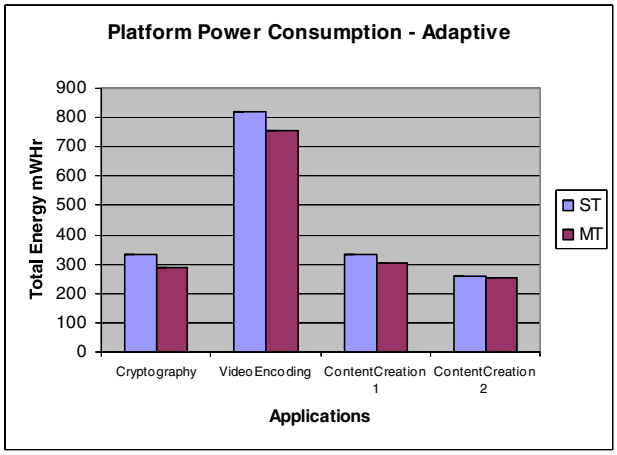


Figure 15: Balanced threading–platform power

Applications with an Imbalanced Threading Model

In this section, we examine power/performance implications on an application with imbalanced threading where the load on the threads is asymmetrical. The analysis here demonstrates how the imbalance in the

threads may cause certain OSs to incorrectly choose a sub-optimal P-state.

We used an in-house gaming prototype primarily composed of a physics computation (collision detection and resolution of graphics objects) and a rendering engine for the analysis. A balanced threading model was achieved by dividing the graphical objects into two cores with each thread taking care of collision detection and resolution of the objects it owned. For the imbalanced case, one thread was given the task of performing collision detection and resolution for the colliding objects while the other thread was given the task of calculating the updated positions. The imbalance was due to the fact that the first thread was more CPU intensive than the other.

The MaxPerf mode refers to the power scheme where the processor is always running at the highest clock speed. The following shows the total energy under different modes with a normalization similar to the one used before. The first data set in Figure 16 represents energy data with a MaxPerf power scheme. The second data set indicates energy consumption with an Adaptive scheme. We focus on these two data sets for now.

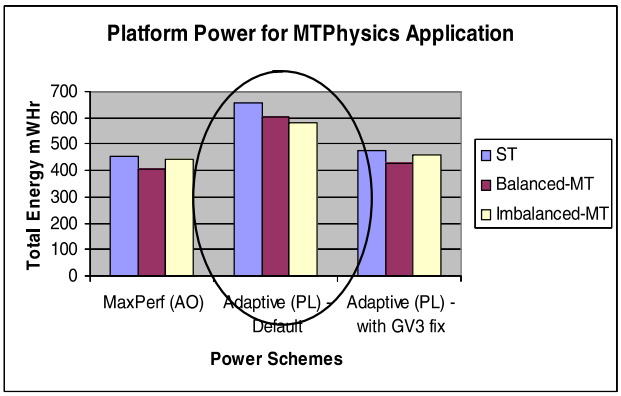


Figure 16: Imbalanced threading–platform power

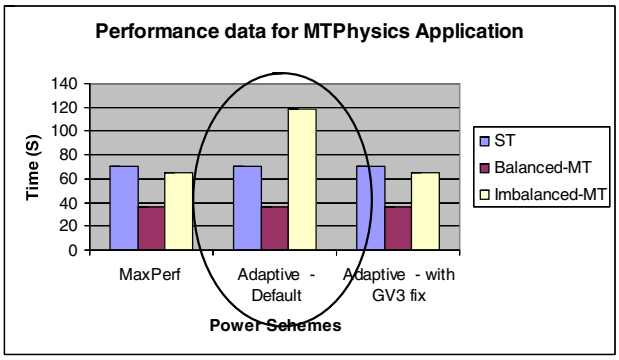


Figure 17: Imbalanced threading–performance data

As indicated in Figure 17 (circled data), the Imbalanced-MT implementation demonstrates 2x performance degradation when running in the Adaptive power scheme as compared to MaxPerf.

Since we use a normalization technique for energy measurements, the platform energy consumption increased for the Imbalanced-MT because the run time is now doubled with the Adaptive-Default scheme (indicated with the circle in Figure 17).

What caused a 2x performance degradation with the Adaptive power scheme?

By looking at the application profile while running the Imbalanced-MT version it is observed that since the first thread is doing more of the work than the second thread, the first thread keeps migrating between the cores, making the effective CPU utilization on the cores ~50%. This is a natural artifact of the manner in which the Windows* scheduler works. However, on a system running in "Adaptive" (portable/laptop) power mode, this thread migration causes the Windows kernel power manager to incorrectly calculate the optimal target performance state for the processor, as the individual cores may appear less busy than the whole package. Due to this, the Windows OS tends to reduce processor frequency in order to save power in Adaptive mode and hence performance may be degraded in Adaptive mode. This in turn causes increased power consumption.

To address this issue of incorrectly calculating the optimal frequency, Microsoft provided a hotfix (KB896256) to change the kernel power manager to track CPU utilization across the entire package, rather than the individual cores, and hence calculate the optimum frequency for applications.

The third data set in Figure 16 (Adaptive–with GV3 Fix) indicates data with the kernel hotfix. In this case, Imbalanced-MT implementation in Adaptive scheme shows power/performance data similar to that of MaxPerf mode. With this fix, processors run at optimum frequency produce expected results.

This study shows that the imbalanced threading model/under-utilized CPU may cause degradation in performance causing increased power consumption. Hence it is recommended to use a balanced threading model while running MT applications or utilize the appropriate operating system fixes.

Platform Power Savings Measurements

The following measurement shows the benefit of platform power savings while running low workloads. The workloads tested in this experiment are mobile mark and DVD playback. The power losses of the VR and power delivery to the CPU are measured with the PSI enabled and disabled. The results are described in Figure 18.

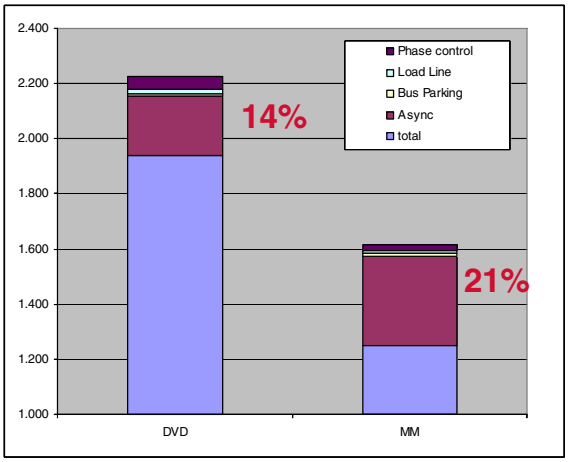


Figure 18: Power losses on VR with and without PSI-2

The above figure shows the overall power losses broken into its components. It can be seen in the above results that activating the PSI-2 reduces the VR losses by 14% while playing DVD and by 21% in Mobile Mark workload.

SOFTWARE RECOMMENDATIONS FOR POWER SAVING WITH MULTI-THREADING

In this section we provide a few highlights from the *Intel Core Duo Optimization Guide*[5]. We mainly focus on the power aspects of the optimizations:

- *Threading done right provides power savings*. As indicated by the data in the "Balanced Multi-threading Model" section, a properly multi-threaded application results in power savings. This includes MT implementations with minimum imbalance and synchronization points. While multi-threading an application, it is always recommended to use a threading model in which all the threads perform equal amounts of work independently. Minimizing synchronization points between threads leads to more time spent in the parallel section which translates to better power savings.
- *Thread imbalance may cause performance degradation and may not provide power benefits as compared to balanced threading.* As discussed in the "Imbalanced Threading Model" section, applications with an imbalanced threading model may show lesser performance improvement as compared to a balanced threading model; and hence consume more power than the balanced implementation. Thread imbalances may cause frequent thread migration across cores that may result in reporting of incorrect processor activity states. This may lead processors to go down to the lower frequency states (if the hotfix provided by Microsoft is not enabled) in Adaptive scheme even if one of the threads is utilizing full processor resources. This issue may occur while running ST applications on a dual-core system in "Adaptive" mode as well.
- *Utilize GV3 hotfix (KB896256) from Microsoft for Windows.* In the case of applications having huge thread imbalances, for example, Thread A utilizing 100% CPU and Thread B utilizing 10% CPU, Thread A will keep migrating between cores. This makes effective CPU utilization 50%. On a system running in "Adaptive" (portable/laptop) power mode, this thread migration causes the Windows kernel power manager to incorrectly calculate the optimal target performance state for the processor. To address this issue, Microsoft provided a kernel hotfix that calculates optimal frequency by looking at the entire package.

  Hence, if an MT application indicates increased power consumption due to reduced performance in Adaptive mode only, install the GV3 hotfix from Microsoft (as the issue might be the one described above). GV3 hotfix will track CPU utilization across the entire package rather than individual cores and will enable the OS to operate at optimum frequency.
- *Interrupt rate*. Applications need to be sensitive to the interrupt rate as the power penalty may increase further on future architectures. In-house testing with few multimedia applications that use a high interrupt rate have shown that the interrupt rate can actually be reduced to default frequency without impacting the user experience. It is recommended that applications tune the interrupt rate to the content being delivered and also to shutdown the interrupts after content delivery.
- *OS scheduling vs. hard affinitizing*. In general, for the Intel Core Duo processor-based systems, it is advisable to use OS scheduling instead of affinitizing threads or applications. OS scheduling may show better power savings due to better performance than affinitizing the threads. The OS scheduler will utilize any under utilized core; while hard affinitizing may potentially degrade performance, since applications may need to wait for the availability of a specific processor even though other processors in the system are idle.

CONCLUSION AND REMARKS

The Intel Core Duo processor-based technology introduces the enhanced power management-aware processor including CPU-based power management actions such as dynamic L2 resizing. Dynamically resizing/shutting down of the L2 cache is needed in

preparation for DeepC4 state in order to achieve lower voltage idle state for saving power. As described in this paper, efficient multi-threading provides optimal performance scaling as well as power savings. Multimedia applications reduce/shutdown interrupts after content delivery to save power.

REFERENCES

[1] Intel® Centrino® Mobile Technology – Enabling Battery Life at http://www.intel.com/design/mobile/platform/platform_collateral.htm

[2] ACPI Specification at http://www.acpi.info/spec.htm*

[3] "Introduction to Intel® Core™ Duo Processor Architecture," *Intel Technology Journal,* Volume 10, Issue 2, 2006.

[4] "CMP implementation in Intel® Core™ Duo Processor," *Intel Technology Journal,* Volume 10, Issue 2, 2006.

[5] *IA-32 Intel® Architecture Optimization Reference Manual,* in http://www.intel.com/design/Pentium4/manuals/248966.htm

AUTHORS' BIOGRAPHIES

Alon Naveh is a senior architect with Intel's Mobile Platform Group in Haifa, Israel, focusing on processor and platform power management. He received his B.Sc. degree from the Technion, Israel Institute of Technology in 1983, and holds an MBA from San Jose State University. Alon co-lead the power management definition of the Intel Core Duo architecture and was involved in the definition of the Intel Pentium M processor-based power management, PCI-E and the Odem chipset. Prior to Intel, Alon worked in Motorola Semiconductor and in National Semiconductor. His e-mail is alon.naveh at intel.com.

Efi Rotem is a senior architect with Intel's Mobile Platform Group in Haifa, Israel, focusing on processor and platform power and thermal management. Efi joined Intel in 1995 and was involved in the definition and development of Pentium 4 and Pentium M processors. Earlier in Intel he led the Intel Pentium with MMX™ technology testing. He received a B.Sc. degree from the Technion, Israel Institute of Technology in 1986. His e-mail is efraim.rotem at intel.com.

Avi Mendelson is a principal engineer in Intel's Mobile Platform Group in Haifa, Israel, and adjunct professor in the CS and EE departments, Technion, Israel Institute of Technology. He received his B.Sc. and M.S.c degrees from the Technion, Israel Institute of Technology and his Ph.D from the University of Massachusetts Amherst. Avi has been with Intel for 7 years. He started as senior researcher in Intel Labs, later he moved to the Microprocessor group where he serves as the CMP architect of Intel Core Duo processor. Avi's work and research interests are in computer architecture, low-power design, parallel systems, OS-related issues and virtualization. His e-mail address is avi.mendelson at intel.com.

Simcha Gochman is a senior principal engineer with Intel's Mobile Platform Group in Haifa, Israel. Simcha has been with Intel for 21 years. Lately he has lead the microarchitecture development of the Pentium M processors and of the Core Duo processor. Prior to that he led the microarchitecture definition of the Pentium Processor with MMX technology and was involved with the design of the 80860 processor and the 80387 numeric coprocessor. Simcha received his M.Sc. degree from the Technion, Israel Institute of Technology in 1984. His e-mail is simcha.gochman at intel.com.

Rajshree Chabukswar is a software engineer working on client enabling in the Software Solutions Group that enables client platforms through software optimizations. Prior to working at Intel, she obtained a Masters degree in Computer Engineering from Syracuse University, NY. Her e-mail is rajshree.a.chabukswar at intel.com.

Karthik Krishnan is a Software Engineer with the Software Solutions Group at Intel. He holds a Masters degree in Mathematics from the Indian Institute of Technology. His current focus is on power and performance optimization of software applications on dual-core platforms. His e-mail is karthikeyan.krishnan at intel.com.

Arun Kumar is an engineering manager in Intel's Software Solutions Group in Dupont, Washington. He received his B.Tech degree from the Indian Institute of Technology, Kanpur and his Masters and Ph.D degrees from the University of Minnesota, Minneapolis, where his research was in the area of image processing and computer vision. Currently his team works on client platforms based on Pentium M, Pentium D and Core Duo processor architectures with special focus on CMP, application software performance, and platform power consumption. His e-mail is arun.kumar at intel.com.

Copyright © Intel Corporation 2006. All rights reserved. Intel, Core, Pentium, Intel NetBurst, Intel SpeedStep, Centrino, and MMX are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

* Other names and brands may be claimed as the property of others.

This publication was downloaded from http://developer.intel.com/.

Legal notices at http://www.intel.com/sites/corporate/tradmarx.htm.

System Memory Power and Thermal Management in Platforms Built on Intel® Centrino® Duo Mobile Technology

Jayesh Iyer, Mobility Group, Intel Corporation
Corinne L. Hall, Mobility Group, Intel Corporation
Jerry Shi, Mobility Group, Intel Corporation
Yuchen Huang, Technology and Manufacturing Group, Intel Corporation

Index words: TS on DIMM, DT in SPD, throttle, thermal, memory performance

ABSTRACT

Specific form factor requirements of the mobile thin and light platform limit the kinds of cooling solutions these platforms can have. As a result, individual components inside mobile platforms (such as system memory devices) can heat up. With increased memory speeds and capacities, we are now reaching the point where the memory thermals are starting to exceed the cooling capabilities of mobile systems. To ensure that the memory devices operate within their thermal limits, their case temperatures need to be monitored and memory accesses throttled if any memory device overheats. In this paper, we discuss the need for memory throttling and address two memory throttling techniques, implemented in platforms built on Intel® Centrino® Duo mobile technology. These techniques greatly improve the memory power/thermal management in a thermally constrained platform by maximizing performance while keeping the system within its thermal limits.

First, we describe Delta Temperature (DT) in Serial Presence Detect (SPD), a novel memory throttling technique, which uses the Virtual Temperature Sensor (VTS) throttling mechanism (VTS was first implemented in the Intel® 815GMCH chipset) to optimize memory throttling control. VTS provides the means to predict the temperature of the memory devices in platforms that don't have a physical thermal sensor on the DRAM modules. When the predicted temperature exceeds the set memory thermal limit, throttling is enforced. We also discuss the implementation details of DT in SPD and the performance benefits it offers in terms of guardband reduction and bandwidth recovery, over the existing open loop throttling techniques.

Finally, we focus on Thermal Sensor (TS) on a Dual In-line Memory Module (DIMM), a closed loop solution for memory throttling control. This technique uses a physical thermal sensor on each of the memory modules that signals the chipset to throttle memory traffic when the memory module exceeds the thermal trip-point. Adding a thermal sensor to the memory module allows the system to continue running at full bandwidth until the critical temperature is reached. In addition to providing both a mechanism for preventing the DRAMs from exceeding the maximum temperature specification (85°C for memory) and a mechanism to control skin temperature, we discuss the performance benefits that TS on DIMM offers over existing VTS-based solutions, including guardband reduction and bandwidth recovery.

INTRODUCTION

Driven by demand and ever more efficient technological advancements, as the form factors of computing platforms get smaller and component densities get higher, system power consumption and thermal issues have become more challenging than ever before, especially in the case of laptops, slim desktops, and blade servers. Naturally, each system has a total cooling capacity per its specific design, and each major component has a cooling limit or power consumption allowance for balancing the system performance in that design. When the total system cooling capacity is smaller than the sum of each component's Thermal Design Power (TDP), as seen in many small-form factor systems, all the components simply cannot simultaneously operate at their TDP level since the system cannot cool them. In such a system, it is also unacceptable to allow one component to free-run without power consumption and thermal restrictions as that would leave the performance of other components to suffer. Balanced cooling limits for major components should thus be achieved at the design level. Having briefly laid the background on cooling limits, we now focus on system memory (specifically in laptops), which is a major power-consuming component of platforms; specifically we

discuss its cooling limits and enhancement in performance through better power/thermal management (memory bandwidth recovery using new throttling techniques).

To better understand the variation in memory cooling limits from system to system, a glimpse at three popular categories of laptops is helpful. The first of these categories is the Thin & Light (T&L) laptop, which is the mainstream, more conventional model. These laptops have a Z-height of around 1.1''-1.2'' and a memory cooling limit of 4-5 watts. The second category is the Mini-Note PCs, which have a smaller form factor than the T&L and a Z-height of around 0.9". They have a smaller cooling fan and a memory cooling limit of 2-2.5 watts. Finally there is the Sub-Note PCs, which have a form factor even smaller than the Mini-Note PCs. These laptops do not have a cooling fan and their cooling limit is around 1 watt. These data clearly show that mobile platforms have limited cooling capabilities, and as the form factors continue to get smaller, the cooling budget also shrinks considerably. With increased memory speeds and capacities, we are now reaching the point where the memory thermals are starting to exceed the cooling capabilities of mobile systems. When the cooling budget is exceeded, that means the system is no longer able to cool the memory subsection, and the DRAM case temperatures begin to exceed their maximum case temperature specification of 85°C. Our lab data, taken on multiple notebook systems while analyzing multiple Small Outline-Dual In-line Memory Modules (SO-DIMMs)[1], show that some 1 GB and greater capacity SO-DIMMs are exceeding their maximum specified case temperature when running a realistic workload at an ambient temperature of 35°C. Thus the memory bus needs to be throttled[2] to ensure that the DRAM devices operate within their thermal limits, reducing the risk of memory corruption and system instability.

Platforms built on Intel Centrino Duo mobile technology implement two memory throttling techniques to address this issue.

DELTA TEMPERATURE (DT) IN SERIAL PRESENCE DETECT (SPD)

Delta Temperature (DT) in Serial Presence Detect (SPD) [3] is a throttling technique that is particularly beneficial in platforms that do not have a physical thermal sensor on the memory modules. At this point in time, not all memory modules have a physical thermal sensor on the DRAM modules monitoring their case temperatures; therefore, their temperature needed to be tracked by some alternative means. A Virtual Temperature Sensor (VTS) throttling mechanism [1] implemented in the chipset (first implemented in the Mobile Intel® 915 Chipset family), provides this alternative solution. VTS predicts memory case temperature based on the measured memory access type at each clock cycle and the speed and type of memory in the system. DT in SPD provides DRAM thermal data for each memory access type (Reads, Writes, Self Refresh, etc.) and includes the DRAM maximum case temperature limit data in the SPD on the memory module. These data are used by the VTS throttle mechanism (which analyzes memory traffic) to estimate the DRAM case temperature and to throttle memory based on the thermal prediction. DT in SPD better predicts the power and thermal of the memory module and hence achieves greater guardband reduction over the open-loop throttling methods followed in the previous-generation mobile platforms, and thus enhances performance in terms of bandwidth recovery. Our experiments clearly demonstrate a performance benefit of almost 15-30% (in terms of sustained bandwidth) by using DT in SPD.

Though DT in SPD has its benefits in platforms that don't have a physical thermal sensor on the memory module, it is still an open-loop throttling technique since it does not actually know the operating temperature of the DIMM. Therefore it must always assume the laptop is operating under the maximum-allowed-room ambient temperature, which is typically considered to be 35°C for mobile environments. So if a notebook is actually running in an environment where the room ambient is only 20°C, then DT in SPD actually assumes it is running at 35°C and will initiate throttling ~15°C sooner than it needs to, thereby losing potential performance that could have been obtained had the system not initiated throttling so soon.

Thermal Sensor (TS) on Dual In-line Memory Module (DIMM)

TS on DIMM is the throttling technique that prevents memory from being over throttled and improves system performance. TS on DIMM integrates a physical thermal sensor onto the memory modules, providing real-time temperature information back to the chipset. The temperature feedback provides a closed-loop throttling mechanism that allows the system to adapt memory throttling to the actual environment the laptop is running in. So instead of just looking at memory traffic and estimating what the temperature of the DIMM *might* be based on estimated memory module power/thermals, now the chipset can wait to throttle until an actual thermal issue

[1] S0-DIMM: Small Outline Dual In-Line Memory Module (DRAM memory modules used in mobile platforms).

[2] Throttling: Solution to reduce memory traffic allowed on the memory bus.

[3] SPD stands for Serial Presence Detect. It's a Joint Electronic Device Engineering Council (JEDEC) spec.

arises. This allows systems running low-power DIMMs, or systems with excellent cooling mechanisms, or systems running at cool room ambient temperatures, to run high-bandwidth applications and not have to throttle memory until a critical temperature is reached. So unlike DT in SPD, TS on DIMM does *not* have to assume the worst-case room ambient temperature. This allows TS on DIMM to greatly reduce the guardband required and thus improve system performance by allowing the system to run unconstrained for longer periods of time. Our experiments clearly demonstrate, due to the guardband reduction achieved using TS on DIMM, the total bandwidth recovered is as much as 30% of the theoretical maximum bandwidth (running a particular application on a specific system and memory configuration).

In the following sections, we discuss the need for system memory throttling and describe the concept, system implementation, and benefits of the two above-mentioned throttling techniques: DT in SPD and TS on DIMM.

NEED FOR SYSTEM MEMORY THROTTLING

In this section we focus on the necessity for throttling the memory bus. The material forms the basis for our discussions in the rest of the paper.

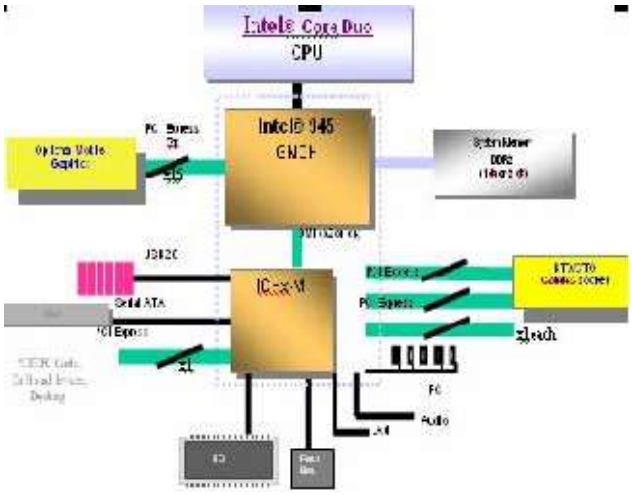


Figure 1: Schematic showing main components in current-generation laptops

Figure 1 shows a high-level view of the main components on a mobile platform built on Intel Centrino Duo mobile technology. The system memory gets accessed in almost all activities taking place in the platform. All data transfers to and from the system memory are managed by the Intel® chipset. Thus, if the chipset/system memory is idle, the platform itself is generally in an idle state. But if there is activity on the chipset/system memory, the platform is consuming more power, causing the chipset and memory components to heat up.

As mentioned earlier, mobile platforms have limited cooling capabilities. In the past, memory speeds and capacities generally allowed the system memory subsection to stay within the cooling limits of the platform, and the DRAM case temperatures were below the maximum operating specifications of 85°C. But with the increase in memory capacity and speed, we are now reaching the point where memory thermals are starting to exceed the cooling capabilities of mobile systems. When the cooling budget is exceeded, that means the system is no longer able to cool the memory subsection, and DRAM case temperatures begin to exceed their maximum case temperature specification.

In a recent laboratory study, several different thin-and-light laptop designs were tested with various SO-DIMMS of different memory speeds, capacities, densities, and vendors, using OpenGL Benchmark software. This software has high bandwidth utilization (about 52 percent of theoretical max) and causes DRAM devices to draw constant high power.

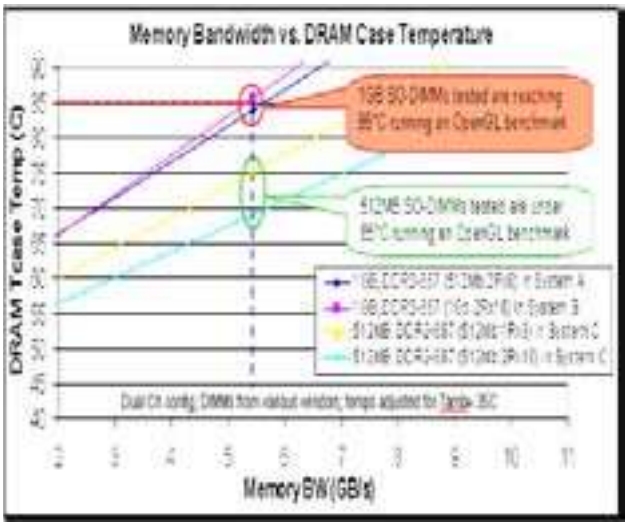


Figure 2: Lab data showing memory bandwidth and DRAM case temperatures

In Figure 2, temperature and bandwidth data are dependent on system, memory, and software configuration. The figure shows the results of a small sample of this thermal study. All three systems are thin and light designs: each system is from a different vendor. System A platform layout has one memory module on top and one on the bottom, System B platform layout has both memory modules on top, and System C platform has both memory modules on the bottom.

The results of the above study show that typical 512 MB SO-DIMMs in various thin-and-light notebook designs are

well below 85°C and do not appear to be in jeopardy of exceeding their thermal limits. In some cases, however, 1 GB-capacity SO-DIMMs are reaching, and sometimes exceeding, their maximum case temperature specification of 85°C. These results show that memory modules are already operating near their maximum specifications, and as memory power and thermals increase with system capacity and speed, memory modules will begin exceeding their maximum specification with multiple realistic workloads. Small form factor designs are of even greater concern due to their lower cooling budgets and thermally challenged environments. There is clearly a need for a robust thermal management solution.

If left unchecked, DRAM devices will start running above their maximum operating case temperatures, and memory-related reliability issues will begin to crop up. Thus the memory bus needs to be throttled to ensure that the DRAM devices operate within their thermal limits. Memory throttling provides a solution to cool the DRAM devices by reducing memory traffic allowed on the memory bus, thereby reducing the power consumed by the DRAM devices and thus reducing thermal output.

We now describe the two throttling techniques in detail.

DT in SPD

DT in SPD is a throttling technique that gives a huge performance benefit to platforms that do not have a physical thermal sensor on the memory module. Before going into the details of DT in SPD, let's briefly look into the throttling methodology followed in the chipsets of previous-generation platforms.

A DDRx[4]-based memory subsystem, as a major power-consuming component on a platform, is peculiar in that its power consumption is difficult to predict in reality. The reason for this is that memory vendors have their own unique processes and design technologies. This results in large variations among different DRAM vendors in power consumption given a fixed device density and speed for each particular DDRx technology. Figure 3 shows the differences in the Burst read current (IDD4R) of the best-, typical-, and worst-case memory supplier (DDR2 512 MB DRAM) across three different memory speeds. As the memory device power is a function of its IDD numbers, a huge difference in the IDD numbers between the best-case and worst-case memory suppliers translates into a huge difference in their power consumption.

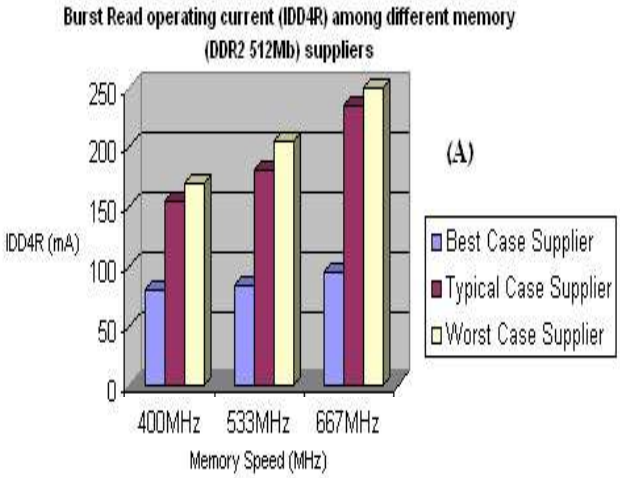


Figure 3: Power differences between best-, typical- and worst-case memory suppliers

In the previous-generation mobile platforms, the system could tell the memory type (device density, number of ranks, device width) and its vendor by retrieving the data from a small EEPROM (called SPD) on each memory module during boot up. But this doesn't indicate how much power the module consumes. Also these platforms did not have a thermal sensor on the DRAM modules to monitor their temperature, and hence their temperature needed to be tracked by some alternative means.

The VTS throttle mechanism (implemented for the first time in the Mobile Intel 915GMCH Express Chipset) provides this alternative solution. VTS predicts the memory case temperature based on the memory access type at each clock cycle. The power estimates for each of the memory access types are programmed in the chipsets throttle weights register [2] and a discretized lumped thermal capacitance model [6] is used to predict the time-varying temperature response to the power consumption. The chipsets in these platforms implemented a throttling methodology without any feedback on power or temperature from the memory module. Without this knowledge of power consumption from the memory module, the system (VTS throttle mechanism implemented in the chipset) was forced to assume the worst-case power consumption for design safety. Thus memory thermal limits (throttling threshold value) were set based on the worst-case supplier's power data. This approach thus resulted in over-guard-banding[5] and over-memory-throttling. The memory performance degrades as low-power memory modules are treated as the worst-case modules.

[4] X is 1, 2 or 3 based on the Double Data Rate Synchronous DRAM memory technology (DDR, DDR2, DDR3)

[5] Reduction in bandwidth to ensure power/thermal values stay within design safety limit.

Now the inevitable question is, how can a system, especially when it does not have a physical thermal sensor on the memory modules, accurately and cost-effectively tell memory power consumption to avoid over throttling memory, i.e., how can it reduce the guardband? It can by using DT in SPD.

Overview of DT in SPD

DT in SPD is a power/thermal prediction scheme providing DRAM power/thermal data in the SPD on the module. It stores key temperature rise data for each memory access type (Reads, Writes, Self-Refresh, etc.) and DRAM maximum case temperature limit data (Tcasemax) in the SPD on the memory module. The system (VTS throttling mechanism implemented in the chipset) uses this information to better estimate the temperature of the DRAM devices and to determine when throttling is necessary. When process shrinks or other power optimizations occur and DRAM power dissipation decreases, the system uses the information stored in SPD to reduce memory throttling and regain system performance.

Usage Model

Memory module vendors report the delta temperature rise parameters and Tcasemax in SPD. These parameters are read by the BIOS from the SPD at boot time. Subsequently, the system adjusts memory throttle limits based on these parameters.

Performance Benefit of DT in SPD

DT in SPD greatly reduces the guardband and helps recover the bandwidth as compared to the throttling methodologies implemented in chipsets in the previous platforms, which set memory thermal limits (throttling threshold value) based on the worst-case supplier's power data. A lab study was done on thin and light laptops using memory modules of different configurations from different vendors. As mentioned earlier, there is a huge difference in the power numbers between different vendors (for the same configuration). The best-case, typical-case, and worst-case DRAM vendors (for each configuration of the memory) in terms of power numbers were used for the study. A chipset stress utility based on Windows*, which generates very high traffic (almost all Page hits) on the memory bus, was used to exercise the system memory. Figures 4 and 5 each give the comparison of performance achieved running the utility with 30% Writes and 70% Reads on the thin and light design. It shows the sustained bandwidth across different memory throttle limits using the two throttling control methodologies: 1) with DT in SPD; 2) without DT in SPD (using worst-case supplier power data).

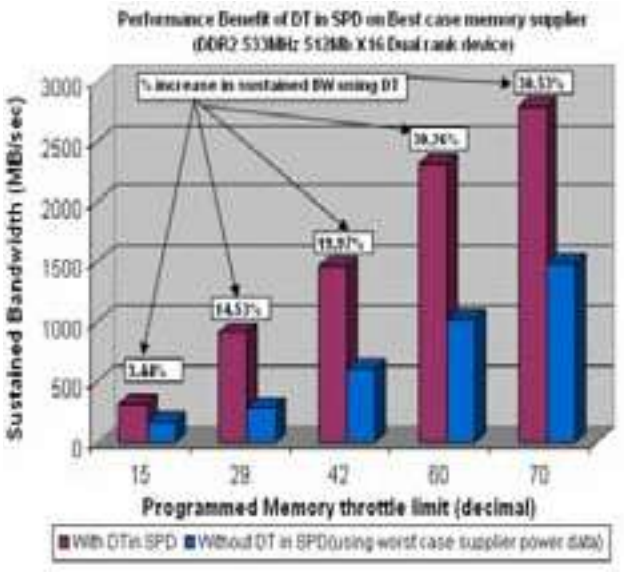


Figure 4: Sustained bandwidth across memory throttle limits (best-case supplier)

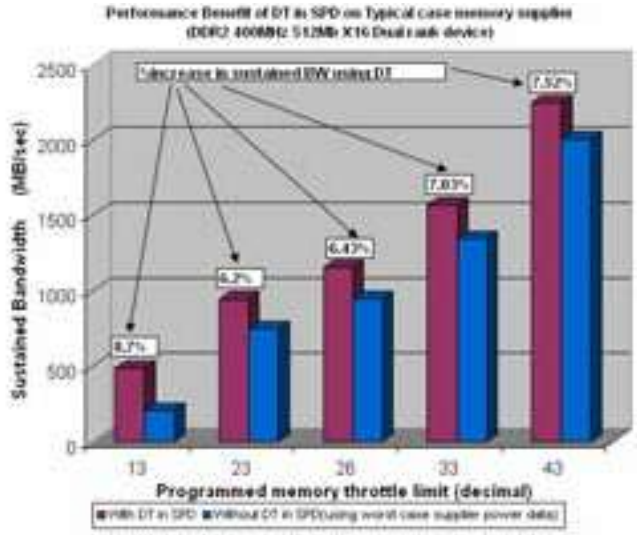


Figure 5: Sustained bandwidth across memory throttle limits (typical-case supplier)

The results in Figure 4 show that a typical-case memory supplier gives a performance benefit of 8% (in terms of sustained bandwidth) using DT in SPD. But for measurements done on a best-case memory supplier, the performance benefit goes up to 30% (in terms of sustained bandwidth) using DT in SPD as shown in Figure 5. Thus the results clearly demonstrate that using DT in SPD greatly reduces the guardband and helps recover the bandwidth. This enhances memory performance, especially for very low-power memory modules (best-case supplier).

Though DT in SPD has its benefits in systems that don't have a physical thermal sensor on the memory modules, it

is still an open-loop throttling mechanism as it does not actually know the operating temperature of the DIMM. Therefore it must always assume the notebook is operating under the maximum allowed room ambient temperature, which is typically considered to be 35°C for mobile environments. So if a notebook is actually running in an environment where the room ambient temperature is only 20°C, then DT in SPD actually assumes it is running at 35°C and will initiate throttling ~15°C sooner than it is needed, thereby losing potential performance that could have been obtained had the system not initiated throttling so soon.

TS on DIMM provides a much more robust thermal throttling mechanism while optimizing performance by offering reduced guardband over other existing methods. The following section describes in detail the concept, implementation details, and the performance benefit achieved using TS on DIMM.

TS ON DIMM

TS on DIMM is a closed-loop throttling technique that offers a more advanced approach to system thermal management through the use of a physical thermal sensor to further reduce guardband present in the existing methods. The reduction of guardband has a direct and positive impact on system performance.

Overview of TS on DIMM

Instead of using power *prediction* to estimate the temperature of the DRAM case, TS on DIMM uses a physical thermal sensor integrated on the DIMM module to monitor the temperature of DIMM. If the thermal sensor detects that the DIMM temperature is exceeding a programmable critical trip point, it triggers an event signal that tells the Graphics Memory Controller Hub (GMCH) to throttle the memory traffic, thereby reducing the DRAM case temperature. The addition of a physical thermal sensor results in a closed-loop throttling methodology that allows for real-time throttling based on the measured temperature. This closed-loop methodology leads to reduced guardband primarily due to the ability to sense ambient temperature. Most of the guardband present in existing throttling mechanisms is due to their open-loop policy, which means that since they do not receive feedback on the actual ambient or DIMM temperatures, they must always assume the worst-case scenario. TS on DIMM provides the needed temperature feedback and allows the system to hold off on throttling until the actual DIMM temperature reaches a programmed critical temperature trip point. The closed-loop policy offered by TS on DIMM is what allows it to reduce the guardband over existing memory throttling methods.

System Implementation of TS on DIMM

System implementation for TS on DIMM is very straight forward; it requires minor changes to the platform, the memory module, and BIOS. Figure 6 provides a block diagram showing the components involved and the logical connections required for TS on DIMM support. The thermal sensor [3] located around the central area of a memory module is connected to the chipset via three wires, two of them for SMBus[6], and the third one for the Event# signal. TS on DIMM implementation does not require any involvement with signals on the memory bus. Any time the temperature of the sensor exceeds the threshold limit programmed by BIOS, the thermal sensor asserts the Event# signal that triggers the chipset to initiate memory throttling until Event# de-asserts. The thermal sensor will de-assert the Event# pin after the temperature goes below the critical trip point.

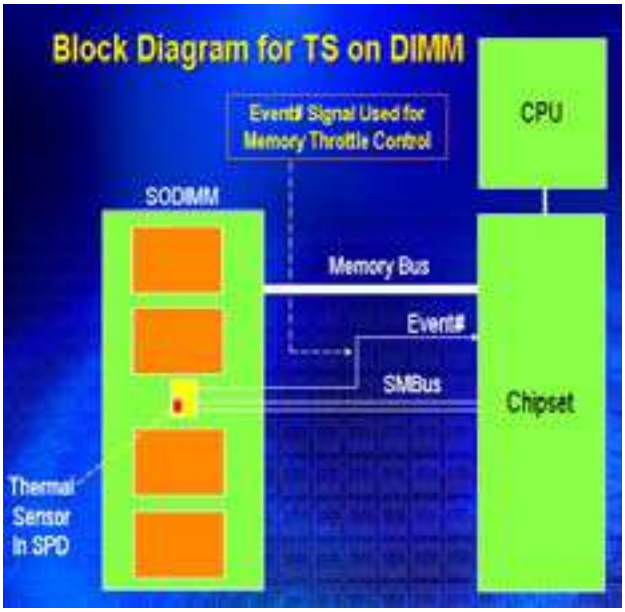


Figure 6: Block diagram for TS on DIMM system implementation

The SMBus master is inside the chipset. The sensor acts as a slave sitting on the SMBus with SPD on a module. Physically, the sensor and SPD [4] can be either integrated into one package as shown in Figure 3 or separated as two stand-alone parts on a module. However, the thermal sensor and the SPD are two separate logical functions using two different SMBus addresses. At power up, BIOS read data from the SPD and initiate the thermal sensor by programming its registers via SMBus, including setting the critical threshold value. When the initiation is

[6] System Management Bus defined by Intel for low-speed system management communication.

complete, the thermal sensor will begin monitoring the temperatures and assert the Event# signal when the temperature of the sensor exceeds the critical trip point. Once the thermal sensor is initiated the system can also poll the thermal sensor via the SMBus at any time and monitor the temperatures real-time.

Platform Requirements to Support TS on DIMM

Routing and hardware support at the platform level is simple; they just require routing the Event# signal from the GMCH [5] to each SO-DIMM connector and placing a 10k ohm pull-up resistor on this signal. For Intel chipsets, the Event# pin functionality and routing is supported on mobile platforms starting from 2006. DDR2 and DDR3 industry-standard SO-DIMM connectors both support TS on DIMM, which only required one additional connection for the Event# signal. SMBus signals were already routed through the connector for the SPD. From a platform perspective, that is all that is required to support TS on DIMM.

Thermal Sensors and Memory Modules

Thermal sensors for memory modules come in two varieties: remote and integrated. Remote thermal sensors are stand-alone 8-pin thermal sensors that can be placed on the memory module and used for monitoring temperatures. This remote thermal sensor option would be used in conjunction with standard SPD EEPROM devices used on modules today. The second option is to integrate the thermal sensor feature into the existing SPD device, also called TS in SPD, thus removing the need for an additional discrete part. Both versions operate in the same manner; they share the SMBus connections as well as the Event# signal connection. So from a platform routing perspective there is no difference between the two implementations: the module could have support for both, although only one implementation is used at a time.

Remote thermal sensors are available in the market today for use in memory module applications. The integrated TS in SPD devices are still in development with some samples starting to be available Q2'06.

Select DDR2 memory modules from limited suppliers will support TS on DIMM; some will support the remote thermal sensor implementation while others will support the integrated option.

RESULTS

Performance Benefits of TS on DIMM

The reduction in guardband that TS on DIMM offers over existing methods gives it a performance edge in both bandwidth as well as benchmark scores when running high-bandwidth applications. For example, when using an OpenGL benchmark on 1 GB DDR2-667 SO-DIMMs in dual channel mode, lab data showed that for every degree C of guardband removed, the system saw up to 220 MB/s bandwidth improvement per degree C, as shown in Figure 7. Since TS on DIMM typically saves ~15°C of guardband over existing methods when running a high-bandwidth application at room temperature, and at 220 MB/s per degree C, the total bandwidth recovered with TS on DIMM can be approximately 3 GB/s for this particular application, which is **30% of the theoretical maximum bandwidth** for this system and memory configuration. This bandwidth recovery assumes that the workload running on the system could utilize more bandwidth if allowed, and the bandwidth recovered is also dependent on the application used and the system configuration.

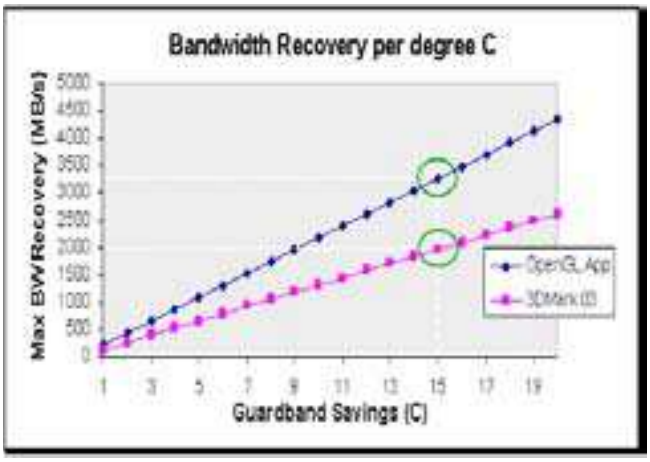


Figure 7: Bandwidth recovered per degree of guardband removed

Due to the increased bandwidth per degree C of guardband savings, benchmark scores also show improvements with TS on DIMM, as shown in Figure 8. (Benchmark score recovery is dependent on system, memory, and software configuration.) Using the same 1 GB, DDR2-667 SO-DIMMs in dual channel mode as described above, for every degree C of guardband removed, the system saw ~3.3 frames per second per degree C for a particular OpenGL application, and up to 13 3DMarks score recovery with 3DMark 03 per degree C. Again, since TS on DIMM typically saves ~15°C over existing methods, that results in an approximately 50 frames per second improvement as well as about a 200 3DMarks score improvement with these applications. Results may vary with different applications and different system configurations.

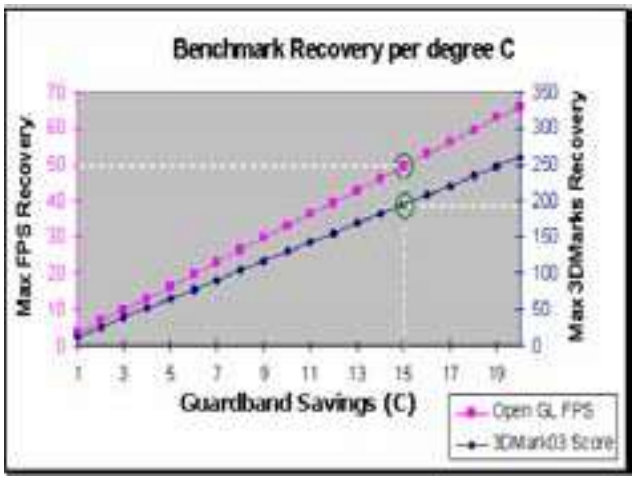


Figure 8: Benchmark scores recovered per degree of guardband removed

To further understand how the reduction in guardband impacts bandwidth availability, let's look at the thermal study from earlier in this paper and determine where TS on DIMM would throttle the worst-case 1 GB SO-DIMM module in comparison to DT in SPD. Please see Figure 9 below (temperature and bandwidth data is dependent on system, memory, and software configuration).

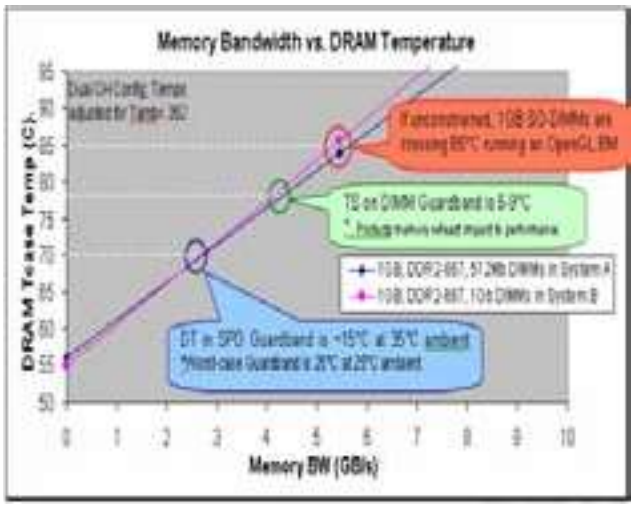


Figure 9: Guardband comparison between TS on DIMM and DT in SPD

The graph above indicates DT in SPD would throttle when the DRAM temperature reaches 70°C (bandwidth available is 2.5 GB/s) while TS on DIMM would throttle only at 79°C (enabling memory bandwidth of 4.5 GB/s). The results clearly show that the reduction in guardband directly impacts bandwidth availability as the temperature of the DIMM reaches and exceeds 60°C at a room ambient temperature of 25°C (or 70°C if the ambient temperature is 35°C).

Other Benefits of TS on DIMM

Aside from the performance benefits mentioned above, TS on DIMM offers other benefits over existing methods. First, since the thermal sensor is measuring true operating temperature, it is able to help protect the DRAM cases from exceeding their maximum specifications even if an outside source is heating the DIMMs. For example, if a fan breaks or if the notebook is sitting in direct sunlight, the thermal sensor is always monitoring the DRAM temperatures and if they exceed the critical trip point, the chipset will throttle the memory traffic and protect the DRAMs.

With the existing methods of temperature prediction based on memory traffic alone, the chipset has no idea if some abnormality is causing the DRAM temperatures to rise beyond their specifications. Since TS on DIMM is monitoring the temperature of the memory, it also provides a way for controlling notebook skin temperatures. Keeping memory within certain thermal limits will also help keep skin temperatures within certain thermal limits.

SUMMARY

Specific form factor requirements of notebooks limit them from having exhaustive cooling solutions. As a result, individual components inside the notebook (such as system memory devices) can heat up to the point of exceeding their operating specifications. To ensure that the memory devices operate within their thermal limits, their case temperature needs to be monitored and memory accesses throttled if any memory devices approach their thermal limits. Current lab data taken on multiple notebooks and from analyzing multiple SO-DIMMs show that 1 GB capacity SO-DIMMs and greater are nearing their maximum specified case temperature when running a realistic workload at an ambient temperature of 35°C. The data were taken from Thin and Light notebook designs: the thermal concern for small form factor designs are of even greater concern due do their limited cooling abilities.

Open-loop throttling mechanisms, including DT in SPD, throttle memory based on a thermal prediction scheme that analyzes memory traffic to estimate the temperature of the DRAM case temperatures. These mechanisms lead to over-guardbanding since they always have to assume worst-case environment conditions, such as the ambient temperature of the room. Also, because these mechanisms are estimating thermal parameters only based on memory traffic, they cannot protect all DRAMs from exceeding their maximum case specifications of 85°C.

TS on DIMM, a closed-loop throttling mechanism, offers several benefits over open-loop throttling methods. The

closed-loop methodology of TS on DIMM allows it to reduce the guardband substantially over existing methods, increasing system performance by allowing the system to run unconstrained longer before initiating throttling. TS on DIMM also provides a way to control notebook skin temperatures and provides a safeguard mechanism to help prevent the DRAMs from exceeding their maximum case specification of 85°C.

Currently TS on DIMM is an optional feature for DDR2 SO-DIMMs. Intel is working with memory industry to drive this feature into DDR3.

ACKNOWLEDGMENTS

Thanks to David Wyatt for writing the initial remote thermal sensor spec and doing the initial study that enabled all future work. We thank Rajeev Menath for his support during TS on DIMM measurements, and Ishmael Santos and Rafael Rodarte for helping during the test plan development as well as supporting the sensitivity study. Thanks to Christopher E. Cox and Rachael Young for helping drive this feature in JEDEC. Thanks to Bill Sanderson for reviewing the results and Deepa Mohan for reviewing the paper and providing feedback. Also, thanks to Ramesh Shetty for his immense help during the measurement setup in the lab. Special thanks to Nilesh Shah and Minh Ngyuen for driving this feature.

REFERENCES

[1] Eric Samson, Sridhar V. Machiroutu, Je-Young Chang, Ishmael Santos, Jim Hermerding, Ashay Dani, Ravi Prasher, Dawid Song, "Interface Material Selection and Thermal Management Technique in Second-Generation Platforms Built on Intel Centrino Mobile Technology," in *Intel Technology Journal*, Volume 09, Issue 01, February 2005.

[2] Jayesh Iyer, Yuchen Huang, Jerry Shi, "DT in SPD: A Better Approach to System Memory Throttling" at *Intel Design & Test Technology Conference*, August 15-19, 2005, Portland, OR.

[3] "Mobile Platform Memory Module Thermal Sensor Component Specification," *JC-42.4*, Ballot proposal (Item number: 1640.07).

[4] JESD21C JEDEC DDR2 SPD Revision 1.1.

[5] "RS-Mobile Intel® 955XM 945 GM/PM/GMS and 940 GML Express Chipset," *External Design Specification* (EDS), Vol. 1 & 2.

[6] R. S. Prasher, J. Shipley, S. Prstic, P. Koning, and J. L. Wang, "Thermal Resistance of Particle Laden Polymeric Thermal Interface Materials," in *Proceedings of International Mechanical Engineering Congress and Exposition*, Nov. 15-21, 2003, Washington, D.C.

AUTHORS' BIOGRAPHIES

Jayesh Iyer joined Intel's Mobile Platforms group in 2003. He is part of the Mobile Platform Architecture and Development team, designing Customer Reference boards for next-generation Intel processors and chipsets. He also works on platform memory power performance initiatives (such as DT in SPD and TS on DIMM) and memory profiling. He has also worked as a research intern at the Indian Space Research Organization. Jayesh received his B.E. degree in Instrumentation & Control Engineering from Gujarat University and his M.S. degree in Electrical Engineering and Computer Engineering from Drexel University, Philadelphia. His e-mail is jayesh.iyer at intel.com.

Corinne Hall joined Intel in 2001 and has worked in both the Desktop Products Organization as well as the Mobile Platform Group. Her current responsibilities include Thermal Sensor on DIMM enabling as well as other memory thermal and power-related activities for DDR2 and DDR3. Corinne received a B.S.E.E. degree in Computer Engineering from Oregon State University. Her e-mail is corinne.l.hall at intel.com.

Jerry Shi has been with Intel for seven years, working as a platform/memory architect. He designed the first Intel® Networking Processor-based cPCI system for demonstration at SuperComm. At Intel he also designed the first set-top-box which could do Picture-in-Picture MPEG playback. He also holds patents related to memory architecture. Prior to Intel, he worked for Philips Semiconductors, Hyundai Electronics, and Oak Technology. Jerry received a B.S.E.E. degree in Computer Science and Engineering from Tsinghua University, China and an M.S.E.E. degree in Electrical Engineering from UMASS at Amherst. His e-mail is jerry.shi at intel.com.

Yuchen Huang is a staff platform memory power Engineer in Intel's Technology Manufacturing Group. She joined Intel in1994, and has held various platform design positions there as Intel's Desktop Product Group platform memory power and performance initiatives owner and as a platform interconnect designer. Currently, Yuchen leads a cross-divisional engineering team within Intel that addresses memory power, thermal, and performance platform challenges. Yuchen received a B.S.E.E. degree in Electrical Engineering from Zhejiang University, China and a M.S.E.E. degree in Electrical Engineering from Portland State University. Her e-mail is yuchen.huang at intel.com.

Copyright © Intel Corporation 2006. All rights reserved. Intel and Centrino are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

* Other names and brands may be claimed as the property of others.

This publication was downloaded from http://developer.intel.com/.

Legal notices at http://www.intel.com/sites/corporate/tradmarx.htm.

Intel® 945GMS Express Chipset for Small Form Factor Platform Based on Intel® Centrino® Duo Mobile Technology

Suresh Subramanyam, Mobility Group, Intel Corporation
Taninder Sijher, Mobility Group, Intel Corporation
Sidharth Krishnama, Mobility Group, Intel Corporation
Parthasarathy Ramaswamy, Mobility Group, Intel Corporation
Deepa Mohan, Mobility Group, Intel Corporation
Vikas Shilimkar, Mobility Group, Intel Corporation
Satish Prathaban, Mobility Group, Intel Corporation
Eric Samson, Mobility Group, Intel Corporation
Michael Derr, Mobility Group, Intel Corporation
Samir Gundawar, Mobility Group, Intel Corporation

Index words: Intel 945GMS Chipset, package, reduction, manufacturing, motherboard, routing, power management, DDR2 memory down

ABSTRACT

Small form factor (SFF) platforms aim to position Intel firmly to address the fast growing mini- and sub-notebook market, expected to increase fourfold over the next three years. This trend is proven by the increasing number of design wins for Intel in this market segment. Intel has made significant advances in mobile technology with their SFF platforms based on the new Intel® Centrino® Duo mobile technology: these include the Intel® Core™ Duo processor Low Voltage/Ultra Low Voltage (LV/ULV), Intel® Core™ Solo processor ULV, and the Intel® 945GMS Express Chipset. To support the mobility vectors[1] of battery life and performance, several power/performance features were designed into the processor, chipset, and platform architectures. This paper describes the power management features of the Intel 945GMS Express Chipset Graphics and Memory Controller Hub (GMCH), the Intel® 82801GBM/GHM I/O Controller Hub (ICH), and the overall platform, all of which have reduced their power consumption while allowing for greater performance. Intel has designed the Intel 945GMS Express chipset to address the mini- and sub-notebook market segments. Given that the package size had to be smaller or equal to that of the Intel® 915GMS Express Chipset (27mm^2) [2], the package design had additional challenges to meet the electrical requirements of the higher frequency Front Side Bus (FSB)/Dual Data rate (DDR) memory and the additional 27 signals. In this paper we discuss the enhancements at the platform level, such as the processor steeper load line and the onboard memory decoupling, which help reduce the Bill of Materials (BOM) cost and save real estate. We also discuss in detail the platform signal and power integrity that enable the DDR2-533 MHz Small Outline Dual In line Memory Module (SODIMM) with onboard memory and decoupling methodology.

This paper serves as a guideline for system designers to understand the benefits of the features described and design better mini and sub notebooks. It also provides an insight into the challenges as package sizes shrink and interface speeds grow. With this SFF platform, Intel reaffirms its strong focus on the four vectors of mobility: breakthrough mobile performance, integrated wireless LAN capability, great battery life, and thinner/lighter design.

[1] Intel Centrino mobile technology is based on four vectors of mobility: breakthrough mobile performance, integrated wireless LAN capability, great battery life, and thinner/lighter design.

INTRODUCTION

The Mobile Intel 945GMS Express Chipset, a member of the Mobile Intel 945 Express Chipset family, is a specially designed Graphics and Memory Controller Hub (GMCH)

targeted for use with small form factor (SFF) platforms. The package and features of the Mobile Intel 945GMS Express Chipset are optimized for size, power, and performance to best suit SFF applications.

The feature set of the Mobile Intel 945GMS Express Chipset makes it a compelling product in mini- and sub-notebook segments for the following features.

- The Intel Core Duo and Core Solo processor Low Voltage (LV)/Ultra Low Voltage (ULV) support with a Front Side Bus (FSB) speed of 667 MHz and 533 MHz, respectively.
- DDR2 533 MHz single channel memory interface. The maximum memory supported is 2 GB.
- Integrated Graphics with a 166 MHz pixel clock. Integrated graphics supports CRT, TV, dual channel Low Voltage Differential Signaling (LVDS), and Serial Digital Video Output (SDVO) interfaces.

The Intel Core Duo LV and Core Solo low-voltage ULV processors are high-performance, low-power, mobile processors with several micro-architectural enhancements over existing Intel® mobile processors. The ULV Intel® Celeron® M processor is a low-power mobile processor with good performance targeted towards value products.

The Mobile Intel 945GM Express Chipset family optimized for SFF-based designs contains two core chipsets: the Mobile Intel 945GMS Express Chipset and the ICH7M (I/O Controller Hub). The Mobile Intel 945GMS Express Chipset provides the host interface controller, System Memory Interface (SDRAM), Direct Media Interface (DMI), and an integrated graphics engine with display output ports.

The ICH7M integrates a number of I/O device controllers and interfaces for legacy and high-speed devices to provide system design flexibility. The DMI, the chipset component interconnect, is designed into the chipset to provide an efficient high-bandwidth communication channel between the GMCH and the ICH7M. The ICH7M also supports PCI Express* x1 I/O ports, next-generation Serial ATA (AT Attachment), and Hi-Speed USB* 2.0 connectivity. An Advanced Configuration and Power Interface (ACPI)-compliant Mobile Intel 945GMS chipset platform can support the Full-On (S0), Suspend to RAM, Suspend to Disk (S4), and Soft-Off (S5) power management states, processor C states, and PCIe-based power management. Through the use of an appropriate LAN device, the chipset also supports wake-on LAN for remote administration and troubleshooting.

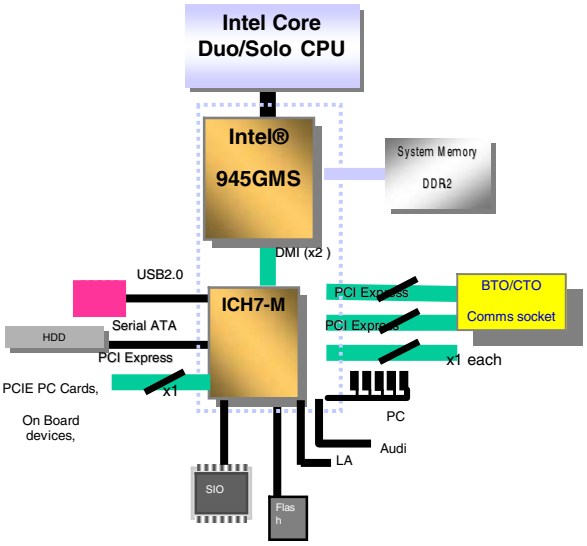


Figure 1: Intel Centrino Duo mobile technology-based SFF platform

In this paper, we discuss the benefits of the Mobile Intel 945GMS chipset over the Mobile Intel 915GMS chipset [2] in terms of the features supported, the challenges in the Intel 945GMS Ball Grid Array (BGA) breakout, motherboard routing, and manufacturability. In comparison to the Mobile Intel 915GMS, the Mobile Intel 945GMS had an additional 27 signals to be routed on the package. Also, the Intel 945GMCH die size is bigger than that of the Intel 915 GMCH by about 15 mils on one side and 22 mils on the other side. The development efforts to deliver the smallest package, meeting all the electrical, layout, and manufacturing requirements are key to Intel's competitive advantage, while pushing the design envelope for thinner, lighter notebooks.

The platform signal and power integrity analysis to enable the DDR2-533 MHz SODIMM with onboard memory and decoupling methodology are discussed in detail. As real estate and BOM costs are key factors in SFF platforms, we describe the platform power delivery with a steeper load line technique and the motherboard decoupling for the Intel Core Solo ULV processors. A steeper load line implementation results in a 50% reduction in the processor decoupling BOM cost compared to the previous-generation SFF platform. Despite the significant improvement in performance in terms of FSB-667 and DDR2-533, the advanced platform power-management features enable lower platform power, compared to previous platforms. The power-management innovation in the Mobile Intel 945GMS Chipset and ICH7M are also discussed in greater detail. Lastly, we discuss the rising challenges foreseen in power management, packaging, design, routing, and electricals.

OVERVIEW OF THE MOBILE INTEL® 945GMS PACKAGE

The Mobile Intel 945GMS Chipset package is 27 mm x 27 mm in size (998 pins). A regular Mobile Intel 945GM Chipset die is used for the Mobile Intel 945GMS package. The biggest challenge was reducing the size of the package since this chipset has an additional pin count (VCCAUX power rail) and more features (LVDS second channel) than the previous-generation Intel 915GMS Chipset (840 pins) [2]. A detailed study of various footprints was done to overcome these challenges. Accommodating additional pins when compared to the Intel 915GMS Chipset in the same package size was possible only by reducing the pin pitch and BGA pad size. Hence, the Mobile Intel 945GMS Chipset was designed with a 0.8 mm uniform pitch and a 14 mils pad size. This smaller pin pitch also introduced a lot of routing complexities. A thorough analysis of motherboard breakout and power-delivery shapes was done to overcome these complexities. This footprint uses the parquet package technique to meet the 27 sq. mm using an 0.8 mm ball pitch. This gives a parquet region of 1.2 mm, utilized as routing channels. Figure 2 shows the uniform 0.8 mm pitch, 27 mm x 27 mm 945GMS package.

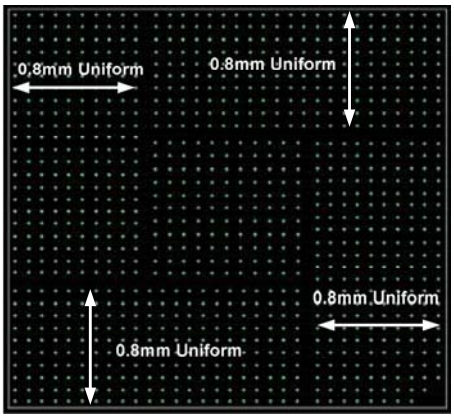


Figure 2: 0.8 mm uniform pitch package

The Challenges of an 0.8 mm Pitch in Motherboard Routing

As discussed earlier, the 0.8 mm (32 mils) pitch poses a significant challenge in motherboard routing. Due to the smaller pitch, it is possible to route only one trace between the two vias, assuming that a 10/20/28 mil via is used, as shown in Figure 3.

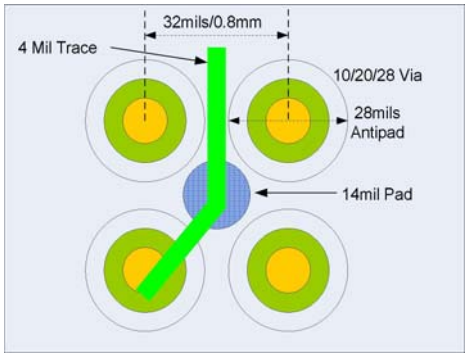


Figure 3: Breakout of one Trace

Therefore, it is difficult to route a 1-byte lane of DDR or FSB signals on one layer. However, this is achieved by routing channels in a special way in the package. Reduced pin pitch provides 70 additional pins in the 27 mm x 27 mm package, which are distributed as No Connect (NC) pins, and additional power pins provide the routing channels for the byte lane in the inner layer as shown in Figure 4. These NC pins or power pins do not require a PTH via, resulting in routing channels that enable to route five traces of 4 mils width through these channels. If those pins are power pins then those corridors are used as power entry corridors on the Top or Bottom layers to feed the power shapes on the motherboard from the Voltage regulator to the GMCH pins, as shown in Figure 4. The Top layer power corridors are used to feed the power to the pins in the parquet region, and Bottom layer copper shapes can be used to feed the power to VCC Core pins located in the center array region.

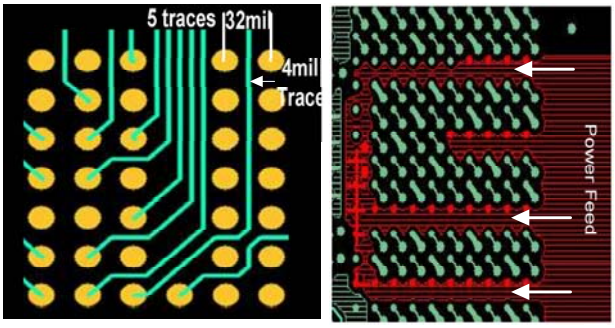


Figure 4: Routing channels in 0.8 mm pitch

Intel® 945GMS Differential Breakout Challenges and Mitigation

The 0.8 mm ball pitch for the Intel 945GMS requires differential signals to break out of the BGA pads on the motherboard as single ended signals for Serial Digital Video Out (SDVO) and Low Voltage Differential Signal (LVDS) interfaces. This single ended breakout implies that the differential signals go through a higher impedance than the target differential 100 ohms till the breakout ends.

Figure 5 shows a snapshot of the motherboard breakout. The impact has been analyzed in simulations and the timing margins and signal quality impact assessed. The design for robustness requires that the length of the single ended breakout for differential signals be restricted to 10 mm. The impact is particularly significant when the motherboard trace lengths are smaller. The MCH driving signals with lower motherboard trace lengths is the worst case condition. The jitter and noise margin numbers for all worst-case conditions show a degradation of 5-7% at the receiver, which is within acceptable limits.

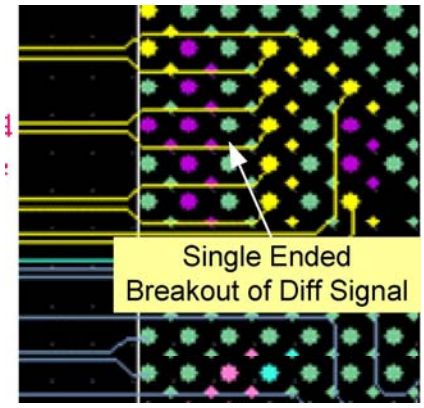


Figure 5: Single-ended routing of differential signals

Challenges in Manufacturing and Solder Joint Reliability (SJR)

Due to smaller pin pitch, the Via to BGA pad spacing is reduced to below 5 mils and hence it adds to the risk of solder bridging during the assembly process. Therefore, efforts have been made to reduce the BGA pad size to 14 mils, which increases the BGA pad to via spacing to 5.6 mils.

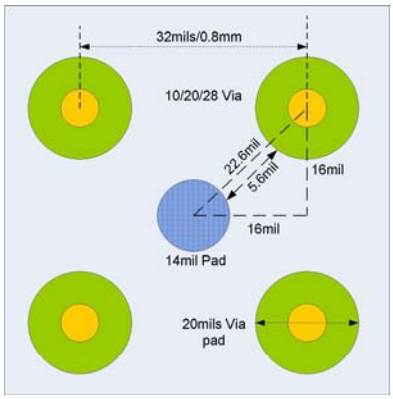


Figure 6: Via to BGA pad spacing

It is also suggested to either encroach the via pad with solder mask in the BGA region completely, or at least partially, to reduce any risk of solder bridging. As the pin pitch and BGA pad size for the Intel 945GMS is lower than the same parameters for Intel 945GM and Intel 915GMS [2], the solder joint reliability for shock and thermal stresses was analyzed by the Advanced Technology group. It was recommended that the five balls in each corner and the center array (die shadow) balls should be sacrificial balls. These sacrificial balls, if used for non-critical to function (NCTF) power pins, should be used as metal defined (MD) pads (50% solder mask opening). Hence, all the center array pins are assigned as NCTF power pins, which serve as additional power pins.

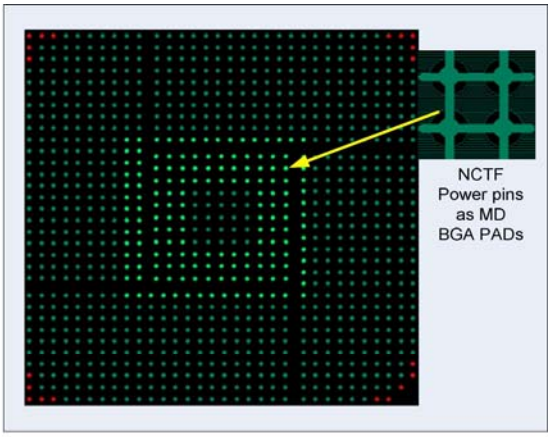


Figure 7: NCTF MD pads for solder joint reliability

INTEL® 945GMS SUBSTRATE LAYOUT AND TECHNOLOGY CHALLENGES

Keep Out Zones

There are certain areas on the surface of the package that need to be clear of any kind of components such as die, capacitors, or materials, such as epoxy. These areas are used for the handling of parts in package assembly and testing and are called Keep out Zones (KOZs). These zones are primarily driven by position of die and the package edge. On the Intel 945GMS Express Chipset with an Intel 945GM Express Chipset die and a reduced package size, KOZs become a considerable challenge as they start overlapping with each other and leave no space for critical components such as package capacitors. In order to resolve some of these concerns, the reduced KOZs that were used in the Intel 915GMS chipset, after a considerable number of experiments, were adopted.

Another challenge arising from KOZs was overlap of the underfill tongue with the package KOZ. Underfill is an epoxy material that is filled under the die for reliability, and it leaves an extended tongue from the side of the application which is east of the die on the 945GMS chipset. This underfill flow was reduced by cutting a trench on the solder mask layer of the package that reduced the capillary effect. This is similar to what was implemented on the 915GMS chipset, but the overlap is

more, due to the larger die size on the 945GMS. DDR and FSB stripline routing in that region was another key factor in enabling implementation of this trench. Figure 8 shows the KOZs and trench.

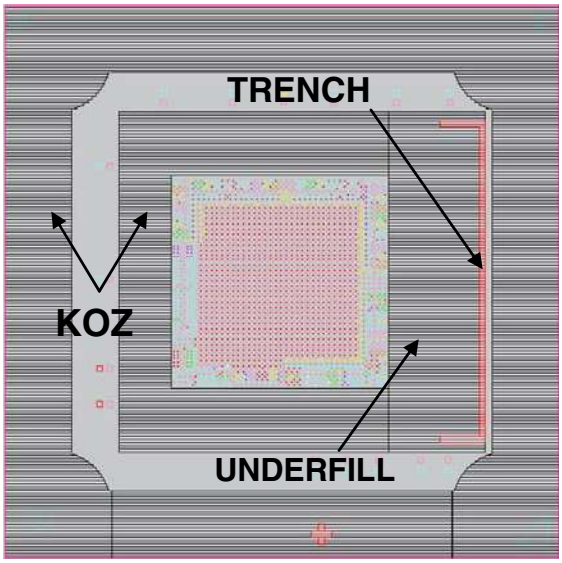


Figure 8: 945GMS KOZ and trench

Layout

Compared to the previous generation Intel 915GMS chipset [2], the Intel 945GMS chipset, in addition to enhanced features like faster DDR and FSB interfaces, also has additional features like dual channel LVDS which resulted in increased signal count on the same package size as the Intel 915GMS. The package capacitor count has increased to 15 compared to 9 on the previous generation Intel 915GMS. This poses additional routing challenges given the same routing area as 915GMS. Die area on the Intel 945GMS has increased by about 10% over the Intel 915GMS. With all these challenges and requirements the package size was maintained at 27 mm sq.

Intel 945GMS Package Layer Stack-up

The biggest challenge the package development team had was to deliver the smallest possible package size that still met all the electrical requirements for high performance and at the same time stayed within the envelope of manufacturing technology capabilities. The primary drivers enabling smaller size was an 8-layer package with stripline routing for single channel DDR and FSB interfaces which forms the bulk of the IO count. Compared to the Intel 945GM package, with dual channel DDR, which had a 44 um trace width and a 54 um spacing in microstrip resulting in 55 ohms impedance, equivalent stripline routing in the Intel 945GMS resulted in a 26 um trace width and a 30 um spacing, resulting in 50 ohms single ended impedance. With 26 um being the minimum trace width allowed in manufacturing, a 5 ohms mismatch between package and the motherboard was treated as low risk from a Signal Integrity (SI) perspective. Reduced spacing to 30 um is compensated by reduced trace-to-trace coupling in stripline. Figure 9 and Figure 10 show the microstrip and stripline structures on the package.

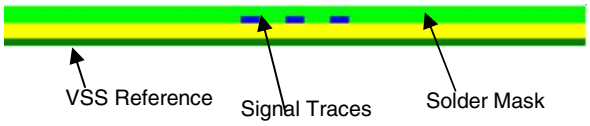


Figure 9 : Microstrip signal routing

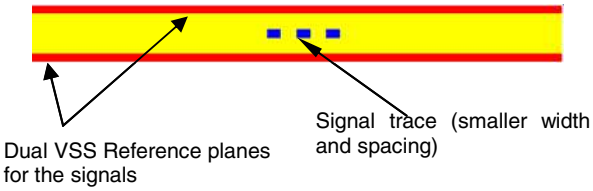


Figure 10: Stripline signal routing

Intel 945GMS Substrate Power Delivery Challenges

While taking advantage of the technology developments made on the 915GMS package (KOZ reduction, shift to 0201 FF package decoupling capacitors, 8L stripline), the Power Delivery (PD) team continued to push design boundaries in order to deliver next-generation performance features. Additional package decoupling capacitors (see Figure 11) and improved power and return path routing on the internal substrate layers helped control AC loop inductance to acceptable levels. The reduced impedance versus frequency response of the power delivery network allowed the PD team to meet stated design targets, thus helping to enable FSB 667 MT/s and DDR 533 MT/s on the Mobile Intel 945GMS Express Chipset.

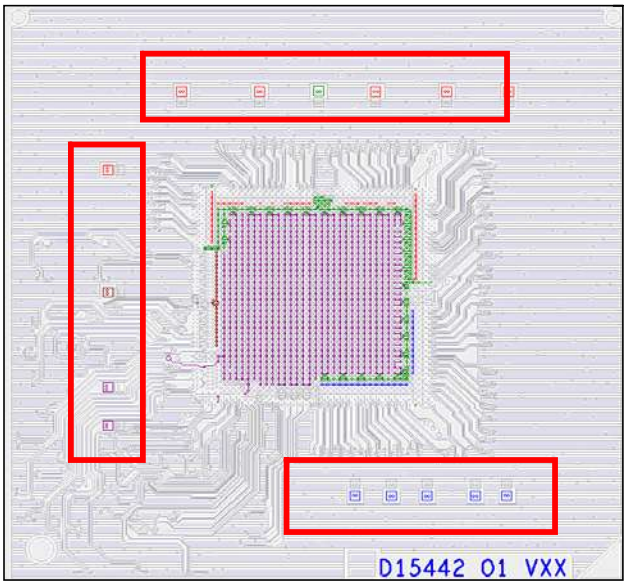


Figure 11: Substrate top layer with package decaps

Another significant technology enabler was the shift to a uniform 0.8 mm (32 mil) ball pitch. The PD team was able to take advantage of the resulting increased pin count to opportunistically assign VCC "power corridors" and VCC islands in the package (see Figure 12). These power corridors allow a wide uninterrupted copper flood on the surface and bottom layers of the motherboard and serve a dual purpose. They can accommodate higher currents, thus improving DC and AC response of the power delivery network.

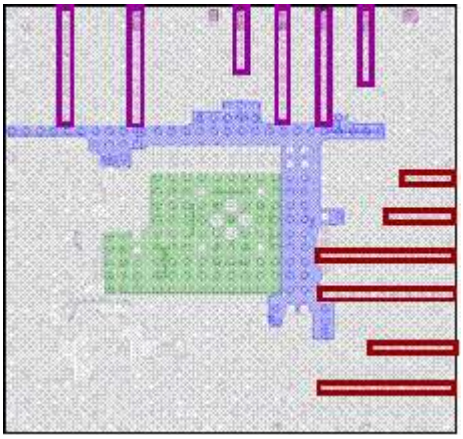


Figure 12: Substrate power corridors and VCC islands

Another benefit of the corridors is that they allow higher signal density breakout on the inner stripline layers of the motherboard due to that region being absent of vias. The absence of a via field on internal signal layers creates additional routing channels, and this allows for greater signal density within a smaller package size. Increased ballout flexibility with the 32 mil ball pitch also enabled the PD team to uniformly distribute "edge" decoupling capacitor pins along the periphery of the package. These capacitors utilize the much smaller substrate routing for current return paths, thereby increasing their effectiveness and improving the robustness of the overall power delivery solution.

PLATFORM POWER DELIVERY

SFF platforms are designed for the mini- and sub-notebook market segment wherein power (battery life), real estate, and BOM costs are the key factors. This section explains how these aspects are taken into account while designing a platform power-delivery solution. Power-delivery architecture for the platform and different power-delivery optimization techniques are discussed.

Platform Power-Delivery Architecture

All the products are battery based and use an adapter with a voltage that varies from 16-28 V. For such a large variation in the input voltage of all the voltage regulators, it is hard to predict the efficiency of the power conversion. In case of higher input voltages and lower output voltages, the efficiency is very low. To resolve the issue, the platform implements a narrow VDC technique in which input to the VR controllers is regulated to give 12.6 V. With this technique, the efficiency of the VRs can be closely controlled, and higher thermal efficiency is obtained as there are fewer thermal losses. This significantly improves battery life. Another important aspect of power-delivery architectures is explained below.

Processor Core Power Delivery

Among the multiple ways to optimize the power-delivery network, is to design the network for higher DC and AC impedance targets to achieve greater savings on real estate and cost.

All the power-delivery network optimization techniques are aimed at a lower droop in order to meet Vmin specifications at the die of the device. This droop basically depends on DC and AC impedance of power delivery networks over the operating range of frequency. Figure 13 shows the relationship of core voltage with maximum frequency.

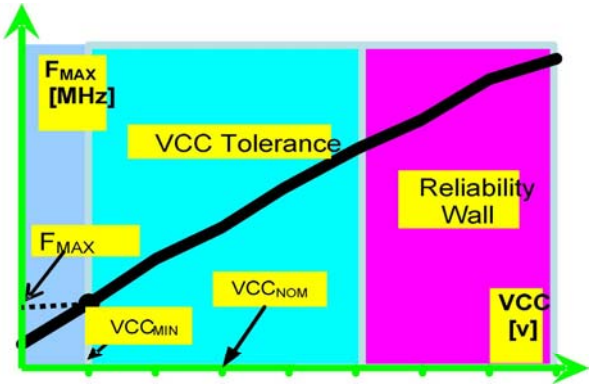


Figure 13: Fmax dependancy on Vcore-min

As is clear from the figure, to maintain the performance, i.e., high speed, a particular minimum voltage needs to be maintained on the core rail. This gives a maximum frequency at which processor core logic can operate without failure. The maximum voltage or overshoot on the core rail is limited by the metal oxide limit of the particular silicon technology as shown by the reliability wall.

The processor platform power-delivery solution comprises a single-phase voltage regulator with an adaptive voltage positioning technique in which the power-delivery network is designed for a steeper load line in the case of ULV.

The ULV processors maximum core current has been significantly reduced from 36A (for Standard Voltage processors)/20A (for LV processors) to 8A (for ULV processors) and the power-delivery network can be designed for a load line with a steeper slope. The concept is explained in Figure 14.

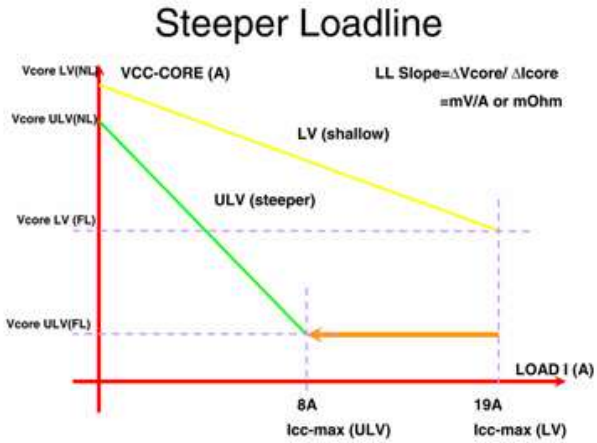


Figure 14: Steeper load line (LL) concept

Load line slope = ΔV/ΔI,

As ΔI ↓, LL can be increased keeping Vmin the same as shown in Figure 14.

As the core maximum current is reduced, the load line gets steeper maintaining the Vmin (refer to Figure 14). Due to the steeper load line, the capacitance required to meet the higher AC impedance is reduced. Therefore the bulk and Multi Layer Ceramic Chip (MLCC) capacitor recommendations for motherboard decoupling are reduced by more than 50%, saving real estate and BOM costs, which are very significant factors in the SFF platform designs.

Power Delivery for Onboard Memory

The Intel 945GMS Express Chipset supports only a single channel 4-rank DDR2 interface, with a maximum memory support up to 2 GB (1 GB per two ranks) at 400 MHz and 533 MHz, respectively. In this section, we discuss the decoupling methodology for the onboard memory configurations that use a total of eight components where four components are on the Top and four are on the Bottom.

Onboard Memory Decoupling Methodology

For the eight DDR2 SDRAMs directly assembled on the motherboard with the four devices on Top and four devices on the Bottom, extensive simulations were done to obtain the optimum decoupling solution. The motherboard is an 8-layer board with Layer 1 (L1), Layer 3 (L3), and Layer 8 (L8) providing the power delivery for the DDR2 core and the I/O. L1 and L8 are the primary layers providing the power to the device pins. The internal layers, L4 and L5, cannot be used for power planes as the memory interface signals in L3 need to have ground referencing on L2 and L4, and signals in L6 need to have ground referencing on L5 and L7 layers. A frequency domain analysis was done on the layout to achieve constant impedance up to the DDR2 operating frequency. A time domain analysis, including single pulse, even mode switching, and worst-case patterns, was performed assuming ideal transmission lines, to verify if the voltage requirements of the devices were met. The DDR2 voltage regulator supplies 1.8 V with a load line of -8.75 mE for a max of 8 A. The VR is modeled as a simple Resistor-Inductor-Capacitor (RLC) network using an extended adaptive voltage positioning concept. The RLC parameters are extracted from the layout to obtain the impedance profile near the pins of the device. The impedance profile is observed at the farthest device VCC pin from the VR output. The simulations were carried out to meet the impedance profile of around 6.75 mohm (3% of 1.8V/8A) up to the operating frequency of the device so that the voltage droop is within the specification of the device. Figure 15 shows the layout implementation for the 1.8 V plane.

We analyzed the generic layout with VR placements both on the East and North of the onboard memory devices

giving the customer maximum VR placement flexibility. The decoupling solution reduced the BOM cost compared to previous platforms providing optimum power-delivery solution.

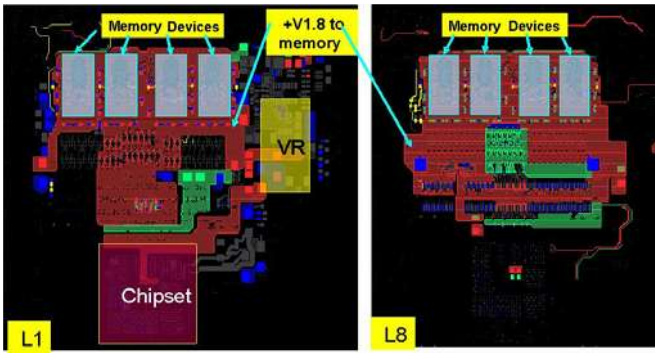


Figure 15: Power layout for memory down

INTEL® 945GMS DDR2 SDRAM SYSTEM MEMORY INTERFACE

The Intel 945GMS Express Chipset supports only a single channel 4-rank DDR2 interface, with a maximum memory support up to 2 GB (1 GB per two ranks) at 400 MHz and 533 MHz, respectively. The DDR2 memory interface consists of four signal groups: clock, command, control, and data. Figure 16 shows the block diagram of the memory interface. Intel gives design recommendations and routing guidelines for various SODIMM and onboard memory configurations for this chipset. In SFF platforms, implementing onboard memory along with the SODIMM provides a significant reduction in the space utilized and the z height. In this section we cover the signal integrity challenges inherent in enabling the 2rank-2rank DDR2 533 memory interface, the key parameters that need to be considered, and some of the best-known routing methods.

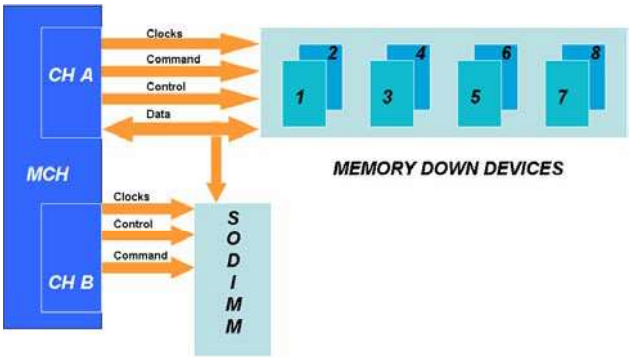


Figure 16: DDR2 memory interface overview

Signal Integrity and Design Recommendations

For the data signal group, being source synchronous, minimizing the flight time skew within the same group of signals is a key parameter that allows their high-frequency operation. As the interface speeds and memory density and loading get higher, the increased skew reduces the timing margins between the data and the strobe signals. In order to achieve good timing margins and also allow maximum routing flexibility in terms of trace lengths, the loading on the data lines needs to be restricted. For example, with a dual rank SODIMM and onboard memory configuration, a user could go with 1-rank on an onboard memory instead of two ranks to get the same memory size. This reduces the loading on the data bus from four SDRAMs to three. The 3-load scenario reduces skew between data and its strobe significantly. However, there is a tradeoff on the power dissipated, as all devices in the 3-load case need to be ON. We switch OFF all devices not being accessed in the 4-load case. However, by restricting the trace lengths between the SODIMM and the onboard memory to less than one inch, good timing margins can still be obtained with four loads on the data lines, but this requires the placement of the onboard memory underneath the SODIMM. Customers have demonstrated such designs by having an optimum cooling solution. Crosstalk is another important parameter to be considered, and it is observed that crosstalk could impact timing significantly with a worst-case timing hit of the order of 200 ps. Increasing the spacing between the data lines reduces crosstalk. Since real state on mobile platforms comes at a premium, designers should choose the right spacing to satisfy all aspects including signal integrity. Using smaller values of the on-die termination (ODT) and having a tighter tolerance on the ODT values yields better timing margins, but the tradeoff is increased power dissipation.

For the onboard memory clock signal group, we recommend a routing topology that yields good differential slew rates of more than 2.0 V/ns with monotonic rising and falling edges and positive noise margins. The signal integrity of the clocks has an impact on all the other signal groups. As the trace lengths of certain routing segments increase, CLK rings back into the DC thresholds causing slew rate degradation. This is overcome by restricting the lengths of certain segments, keeping in mind the layout feasibility, and choosing optimum values for the differential capacitance.

All the data lines that belong to the same group should be routed on the same internal layer for the entire length of the bus. This practice results in a significant reduction of the flight time skew, since the dielectric thickness, line width, and velocity of the signals will be uniform across a single layer of the stack-up. DQ Bit swapping within a byte lane is allowed whether the routing is to the SO-DIMM or onboard SDRAM components. Data Bit swapping works best for the memory down configurations that use a total of eight components where four components are on the Top and four are on the Bottom. With bit swapping the lengths to the Top and Bottom

SDRAM components can be routed to exact lengths for matching and also ease of routing. Figure 17 illustrates the bit swapping for data lines for the memory devices placed back to back.

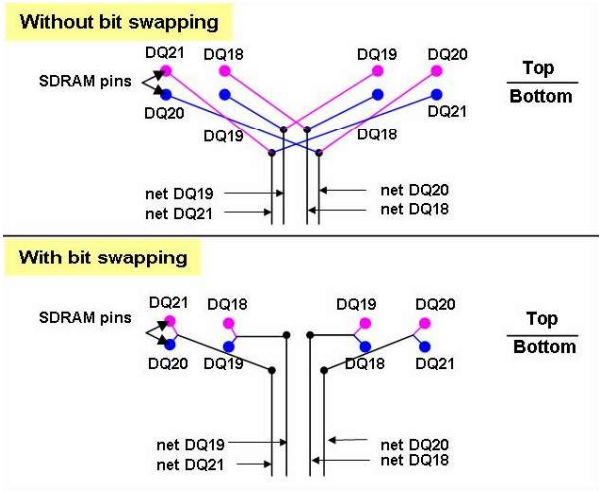


Figure 17: Bit swapping in data lines

GMCH POWER MANAGEMENT

The challenge in every new generation of chipsets is how to increase the platform performance while also increasing the platform battery life. Platform battery life is measured using several benchmarks. A typical benchmark is the Mobile Mark 2005 (MM05). Figure 18 shows system memory bandwidth versus time for this particular benchmark. It is apparent that the GMCH workload primarily consists of a relatively constant demand for a small percentage of read bandwidth, and a much smaller write bandwidth.

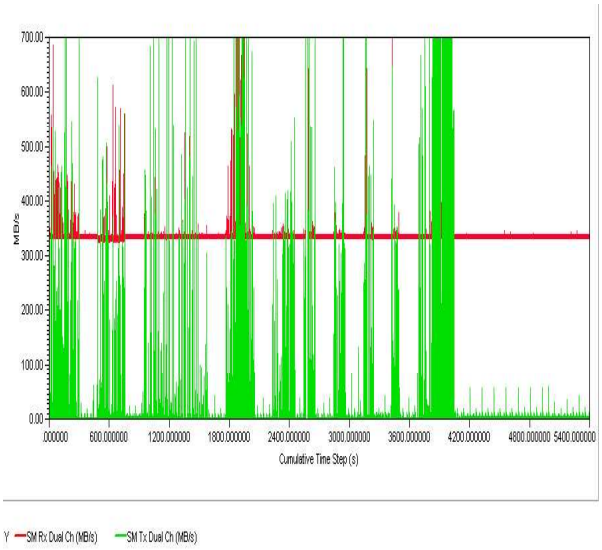


Figure 18: Memory traffic vs. time for MM05 OP

It turns out that while there is negligible usage of the graphics engine during the benchmark (therefore that power component can be ignored as long as there is good dynamic clock gating of this engine), the monitor/flatpanel needs to keep being updated at the required display refresh rate even if nothing is visually changing on the monitor/flat panel. Display refresh requires relatively power hungry accesses to the external memory, since the display frames are too large to store internally to the GMCH, and this represents the majority of the read bandwidth.

As the FSB and memory speeds increase, GMCH internal clock frequencies also increase, and hence their possible contribution to average power also increases, especially if a voltage increase is required to maintain critical timings.

Finally, another potentially large contributor to platform power is bus mastering from attached devices traffic (USB, PCIE, FIR, Audio).

These types of workloads are a small fraction of the MM05 benchmark, but a badly configured platform can easily suffer a significant power impact. This is because these types of workloads (especially if they are making unnecessary memory accesses when they are idle) have the potential to force the platform into a higher platform power state that is required to provide the short memory request latency and CPU snoops they need.

In summary, the solution to lowering the GMCH average power contribution is to do the following.

- Keep core voltages as low as possible.
- Minimize the impact of increasing I/O frequencies.
- Keep the overhead of display refresh power low.
- Minimize the impact of bus master traffic.

GMCH Power Management Features

As in previous generations of the Intel® mobile chipset [1], a large factor in achieving these goals is via main memory power management during normal operation and in low-power ACPI Cx states. This includes such power-management features as Dynamic Memory Row Power Management (DRPM) and Conditional Memory Self-Refresh (CMSR). CMSR mode includes dynamic gating of most GMCH clocks, since a MHC has no useful work to do whenever memory can be placed in a self-refresh state. In an analogous fashion, I/O circuits are also placed in lower power states.

One of the most efficient ways to control power is to minimize voltage. The 82945PM (targeted at larger form-factor platforms requiring up to two channels of higher speed memory), for example, needs to run some of its core logic at a higher voltage than the rest of the chipset core in

order to operate at higher FSB and DDR frequencies. The Intel 945GMS Express Chipset runs its core at the lowest voltage that can meet its performance requirements.

Given that display refresh must be kept serviced and is the predominant memory traffic, the GMCH attempts to maximize the amount of time spent in the CMSR state, and minimize the amount of time spent exiting the CMSR state. One way the GMCH does this is by using a large First In First Out (FIFO) Random Access Memory (RAM) to store very quick but large read bursts from memory so that the FIFO contents can be used to feed the slower display pipeline while the GMCH and memory are in the CMSR state. The self-refresh efficiency (= time spent in self refresh/total time) improves as a function of this FIFO size, as shown in Figure 19, although this is modulated as a function of the exit time from self refresh mode.

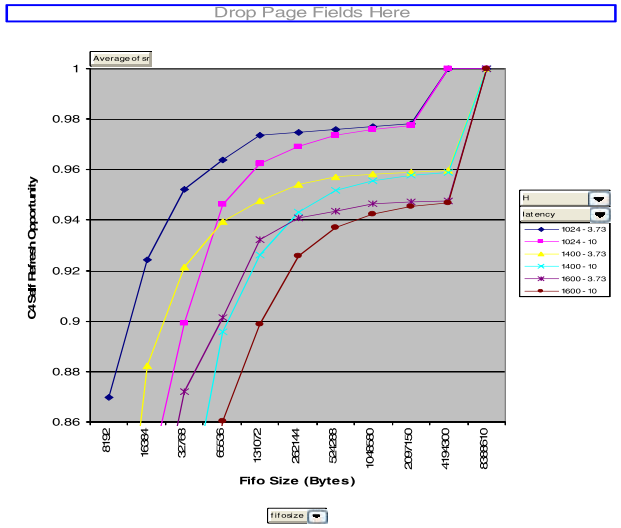


Figure 19: Self-refresh efficiency

Previous mobile chipset generations automatically jumped and stayed in the high-power CPU C2 state during bus mastering traffic. The only way back to a lower power C3 state had also required a jump back to the C0 (highest power of the C states) where the decision to go back to the C3 state could be made by the ACPI software.

The Intel 945GMS Express Chipset has increased the amount of time that the platform is in a lower power state by not always requiring the platform to go fully into the C2 state to handle snoops. Instead, only those portions of the platform required to handle snoop traffic are awakened momentarily to handle the snooping.

ICH Power Management

The ICH7M (82801GBM and 82801GHM) builds upon previous generations. It continues to implement the advanced platform power-management control functionality that was present on the previous generation. This functionality aggressively lowers the processor, GMCH, and clock chip power consumption during brief periods of idleness through control signals to the clock chip, processor voltage regulator chip, GMCH, and the processor. Additionally, the ICH7M integrates other features and new capabilities that are valuable in the SFF segment through power and board area savings.

The ICH7M implements aggressive on-die clock gating. With many independent I/O subsystems integrated on the ICH7M die, the clock-gating is implemented in a distributed manner such that particular branches of a clock tree are shut off to those subsystems that are idle. This reduces idle and average power consumption.

The ICH7M also reduces the core operating voltage from 1.5 V on the previous generation to 1.05 V. This significantly reduces idle, average, and thermal power, all of which are critical to the SFF segment.

The ICH7M also integrates on-die regulators and power switching logic that avoid external components and allow for efficient power distribution. Because the ICH7M (and previous generations) contains system suspend/resume (for S3, S4, and S5 states) control logic, the system real-time clock, system CMOS, and the LAN MAC support multiple isolated power wells that must stay on when the main power supply is off. The ICH7M contains integrated voltage regulators that can supply low-voltage rails from the 3.3 V rail that is available externally in these suspended and off-power states. The platform only needs to supply the 3.3 V rail for each of these power wells (avoiding switches or VRs on the board), while the ICH7M reduces the voltage–and therefore power–on the silicon die. In the S0 state, the ICH shuts off the internal regulators and switches over to the equivalent voltage rails in the main power well. This guarantees that the efficient power supply infrastructure that is in place for the higher power S0 state is utilized, thereby minimizing on-die and platform power loss (idle, average, and thermal) due to power regulation inefficiencies in S0.

The ICH7M also adds features that reduce the system flash footprint on the board. First, the ICH7M adds a Serial Peripheral Interface (SPI) for the system flash as an alternative to the existing Low Pin Count (LPC) interface. This enables the use of high-volume, SFF, low-cost SPI flash devices from a variety of vendors. Secondly, the SPI flash controller in the ICH7M supports flash-sharing with an Intel® LAN NIC such that a single SPI flash component contains the LAN configuration information as well as the system BIOS data. This saves one non-volatile memory component on the board.

Silicon Results

As a result of the chipset power optimization work, the average power of the Mobile Intel 945GMS Express Chipset has gone down from that of the previous generation, all the while providing increased platform performance.

Table 1: Silicon results

	ICH MM05 Power	GMCH MM05 Power
Intel® Centrino® mobile technology platform (Previous Generation)	1.5 W	1.5 W
Intel® Centrino® Duo mobile technology platform	.67 W	1.2 W

CPU AND OTHER POWER MANAGEMENT FEATURES

Other than 945GMS and ICH power-management techniques discussed in the previous section, this platform also supports a few other power-management techniques discussed below.

Intel® Media Storage Manager (IMSM)

This is the power management technique used in the SATA interface of ICH7M. It uses the link power management of the SATA interface by changing the physical SATA link to lower power states. This technique can provide a power savings of up to 400 mW in AHCI mode.

Active State Power Management (ASPM)

An active state power management feature contained in the PCIe* specification is supported by all the PCIe x1 links of ICH7M. The two low-power states, L0s and L1, also called standby states, are supported by ICH7M in ASPM-enabled systems. L0s is a short entry and exit latency state where any transmit link can be placed into a low-power state if there is no traffic on the link. The L1 state is optimized for more power savings than the L0 state, but it comes with the cost of longer exit latencies. ASPM is supported by ICH7M but needs to be enabled by the software.

Processor Power Management

LV/ULV processors support C1/AutoHALT, C1/MWAIT, C2, C3, C4 for optimal power management. Figure 20 explains the processor power management states and signals that transition the state. Enhanced Intel SpeedStep® technology features multiple voltage/frequency operating points for optimal performance. The processor voltage regulator is designed for smooth voltage transition with low latency. Depending on the thermal sensor, the processor can change the core voltage with voltage identifiers. This provides better adaptive thermal management. Voltage regulator efficiency can be optimized with power status indicator pins that assert when the processor is in a lower power state.

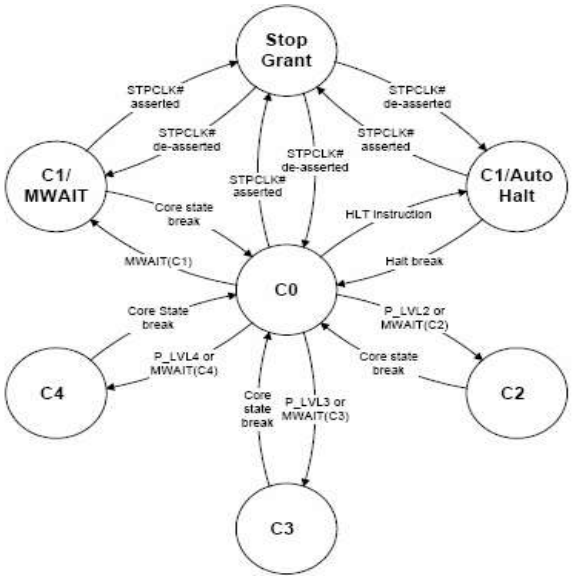


Figure 20: CPU power states

System Power Management

SFF platform supports S0 (active), S3 (stand by-suspend to RAM), S4 (hibernation-suspend to disk) and S5 (soft off).

FUTURE CHALLENGES

As the form factors of mobile platforms are getting smaller with faster processor and memory speeds, power management, packaging technology, platform power delivery, signal integrity, and the motherboard implementation have become more challenging than ever before. These challenges are discussed below.

Power Management Challenges

Even though excellent performance and power results were achieved on Intel Centrino Duo mobile technology systems that contain the GMCH and ICH, both performance and average battery life need to keep increasing.

There are three vectors to be considered:

- Graphics features and performance requirements are continuing to follow Moore's Law at a faster than average pace.
- We need to support ever increasing I/O bandwidths that have surpassed what can be done with unterminated I/O.
- Process technology is increasing leakage as a power component.

Packaging Challenges

To support additional functionalities on the chipset, the pins required on the chipset are continuing to increase. This increase in pin count and demand for reduced package sizes impose a big challenge in the substrate design and manufacturability.

- Smaller package ball pitches and reduced motherboard vias/spacing are key drivers for reducing package size.
- Smaller package size means smaller solder ball sizes due to reduced motherboard pad size and substrate pad size. This reduces the solder joint reliability.
- Smaller packages drive more layers on the package, in turn impacting the cost significantly.
- Decoupling capacitors need to be accommodated due to smaller packages.

Platform Challenges

In the sub and mini notebooks, one does not have the luxury of available real estate as the form factors are continually being reduced.

- The need to support additional features to improve performance yet still reduce the BOM cost and real estate.
- Signal integrity challenges in terms of timing margins and signal quality for higher and higher DDR/FSB frequencies need to be overcome. These impose stringent motherboard routing guidelines.
- Due to smaller package sizes and reduced pin pitches, the motherboard breakout becomes even more difficult.

SUMMARY

The Intel 915GMS chipset, which was the first of the smaller form factor chipsets, launched by Intel was very well received by customers, and this motivated Intel to come up with a better next-generation SFF chipset, the Intel 945GMS Express Chipset. Mobile Intel 945GMS comes in 27 mm x 27 mm package size with additional features (such as additional LVDS channel), higher FSB and DDR speed, optimized motherboard power delivery, platform power management, and innovations in chipset power management. All the above features make this platform ideal for mini and subnote segments of notebook PCs. With this platform Intel has delivered a best in-class SFF platform. Intel's customers will benefit significantly from Intel's strong focus on this market segment.

ACKNOWLEDGMENTS

The authors thank A. Khandekar and S. Pal for providing the power measurements used in this paper. We acknowledge all the teams who worked on this platform: Product Development, Package Development, Power Delivery and Signal Integrity, Boards Development, Technical Marketing, and all the validation teams.

REFERENCES

[1]"Innovation Brings Low Power Integrated Graphics to the Intel® Centrino® Mobile Technology Platform," *Intel Technical Journal*, Vol. 7, Issue 2, May 2003.

[2] "Intel 915GMS Chipset: in Mobile Platforms, Smaller is Better, *Intel Technology Journal* Vol. 09, Issue 01, Feb. 2005.

AUTHORS' BIOGRAPHIES

Suresh V. Subramanyam is the package design manager with Intel, India, supporting mobile chipsets. He has a B.S.M.E. degree from India and an M.S. degree in Materials Science & Engineering from Singapore. He joined Intel in 2003 after spending six years with HP/Agilent, Singapore. He started his career with Bharat Electronics, Bangalore, India, in 1988. His e-mail is suresh.v.subramanyam at intel.com.

Taninder S. Sijher received his M.S. degree from the University of Mississippi in 2004 and his B. Tech. degree from the Indian Institute of Technology, Bombay in 2001. He has been with Intel since 2004 working as a package electrical engineer with Mobility Group India and is responsible for the electronics of the next-generation Mobile GMCH packages. His e-mail is taninder.s.sijher at intel.com.

Sidharth Krishnama is a power delivery design engineer with Intel India, supporting mobile chipsets. He has a B.S.E.E. degree from India and an M.S.E.E degree from the University of Oklahoma, Norman. He started his career with Intel in 1999. His e-mail is sidharth.krishnama at intel.com.

Parthasarathy Ramaswamy obtained his Ph.D. degree from IISc., Bangalore, India and is working with Intel

Technology India Pvt. Ltd., Bangalore as an SIE manager. He is responsible for signal integrity and power-delivery solutions for the chipsets group. His areas of interests include high-speed PKG/Board designs, power and signal integrity studies, etc. His e-mail is ramaswamy.parthasarathy at intel.com.

Deepa Mohan joined Intel's Mobile Platforms group in 2003. She is a hardware design engineer in the Mobile Platform Architecture and Development team, designing Customer Reference boards for next-generation Intel® processors and chipsets. She has also worked on memory signal integrity simulations and technical marketing. Deepa received her B.E. degree in Electronics & Communication from R.V. College of Engineering, Bangalore, India in 2003. Her e-mail is deepa.mohan at intel.com.

Vikas Shilimkar is a hardware design engineer with Intel, India. He joined Intel in 2004 and works in the Mobile Platform Architecture and Development group. He works on customer reference motherboard designs for Intel's next-generation platforms. His specializations include platform and package power delivery. He holds a B.E. degree in Electronics and Telecommunication from COE Pune, India. His e-mail is vikas.s.shilimkar at intel.com.

Satish Prathaban is a hardware design engineer in the Mobile Platform Architecture and Development team. He has an M-Tech degree from IISc-Bangalore. He started working at Intel India in 2003 in the area of motherboard and package power delivery. His e-mail is satish.prathaban at intel.com.

Eric C. Samson joined Intel in 1983 to design communication chipsets. He currently focuses on power management architecture for chipsets. His e-mail is eric.c.samson at intel.com.

Michael N. Derr joined Intel in 1992 and has held various positions in the Client Chipset Development group over the years. Most recently, he has been working as an architect on I/O features and platform power management. His e-mail is michael.n.derr at intel.com.

Samir Gundawar is a hardware design engineer in the Mobile Platform Architecture and Development team. He has an M-Tech degree from IIT-Delhi. He started working at Intel India in 2003 after working in Wipro Technologies for four years. His e-mail is samir.v.gundawar at intel.com.

Copyright © Intel Corporation 2006.All rights reserved. Intel, Centrino, Core, and Intel SpeedStep are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

* Other names and brands may be claimed as the property of others.

This publication was downloaded from http://developer.intel.com/.

Legal notices at http://www.intel.com/sites/corporate/tradmarx.htm.

THIS PAGE INTENTIONALLY LEFT BLANK

WLAN System, HW, and RFIC Architecture for the Intel® PRO/Wireless 3945ABG Network Connection

Mark Ruberto, Mobility Group, Intel Corporation
Ra'anan Sover, Mobility Group, Intel Corporation
Jorge Myszne, Mobility Group, Intel Corporation
Alexander Sloutsky, Mobility Group, Intel Corporation
Yair Shemesh, Mobility Group, Intel Corporation

Index words: wireless, WLAN, Wi-Fi, hardware, HW, TX power calibration, closed loop, silicon, RFIC, front end module, platform integration

ABSTRACT

The corporate challenge of being "one generation ahead" has prompted us to find new ways to improve our time to market, size, and cost of our WLAN products. By identifying the key issues which challenged us in our previous-generation products and by shifting our design effort to a platform-level optimization, we moved to a top-down approach in our Si and hardware architecture design flow. By taking the product-level requirements down to the last component on the board using this new methodology, we enabled smoother and cleaner platform integrations, as well as smaller and cheaper solutions. This paper summarizes the different architectural and functional changes in our products, from the system level down to the board/Si level. In addition, we discuss some of the design issues to ensure robust and reliable RF designs in our radio chips. The Intel® PRO/Wireless 3945ABG Network Connection, which is a part of Intel® Centrino® Duo mobile technology, is the first Intel® WLAN product to be conceived in this way.

INTRODUCTION

The corporate challenge of being "one generation ahead" in our WLAN product line has challenged us to improve our time to market, product size, and cost. To achieve these goals, we take a top-down approach to our Si and hardware designs in order to optimize platform development. This top-down approach led us to consider the product-level requirements down to the last component on the board, which not only enabled smoother platform integration, but also cheaper products. In this paper, we describe the overall approach used in the development of the Intel PRO/Wireless 3945ABG Network Connection, which is part of Intel Centrino Duo mobile technology, and is the first Intel WLAN product to be conceived using this new top-down approach.

The Intel PRO/Wireless 3945ABG Network Connection conforms to the IEEE 802.11a/b/g/d/e/h/i standards and supports data rates from 1 Mbps up to 54 Mbps. It is capable of transmitting output powers up to 18dBm. It operates in the 2.400-2.484 GHz and 5.17-5.85 GHz frequency ranges and supports a Wake on Wireless LAN (WoWLAN) feature.

In this paper we summarize the different architectural/functional design approaches and tradeoffs taken from the system level down to the board and transistor level on Si. First, we summarize the system, board, RFIC, and front-end module architectures. Next, we discuss the design considerations and tradeoffs for each of the architectures. Finally, we summarize the high-volume testing challenges that had to be overcome to enable a reduction in production test time by a factor of 3, from 104 to 32 seconds per card for the wireless platform. The salient features and successes of this overall methodology can be seen in a reduction in the platform integration cycle time from 6 to 4 quarters, a "world class" wireless solution yield improvement through self calibration schemes to better than 98.5%, a modularity approach with built-in reliability checks to the RFIC design flow, migration to a smaller form factor (FF) platform solution (single-sided Mini card) with the lowest part count for this FF class, and a lower product cost achieved through customized board elements and vendor multisourcing.

ARCHITECTURE

System Architecture

The block diagram of the Intel PRO/Wireless 3945ABG Network Connection is shown in Figure 1. The system is composed of two chips, fully designed by Intel, and other third-party board components. One chip contains all the digital and mixed signal components including analog to digital converter (ADC) and digital to analog converter (DAC) while the other contains the RF sections. The media access control/baseband (MAC/BB) chip consists of the host interface, MAC processor, BB subsystem (OFDM and CCK modem), and the ADC/DAC. The RFIC chip consists of the synthesizer and full transmit/receive (TX/RX) chains, from BB to RF.

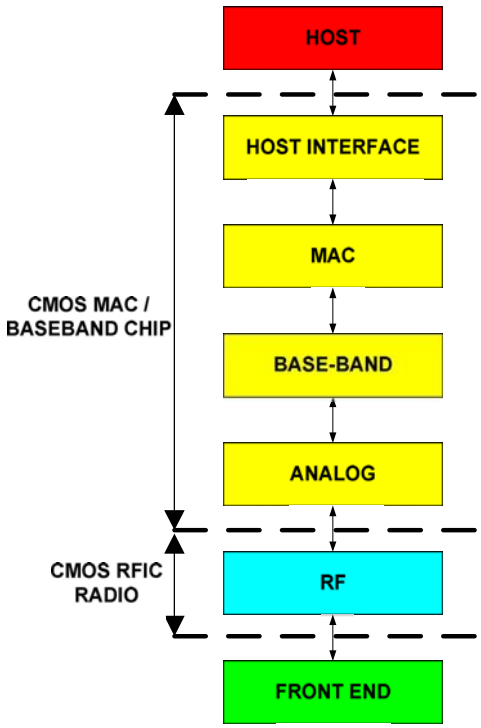


Figure 1: System block diagram of the Intel PRO/Wireless 3945ABG Network Connection

Board Architecture

The board architecture shown in Figure 2 was based on the silicon partitioning and so we had four main sections:

1. The MAC/BB with an associated EEPROM (non-volatile memory) used to store all board-specific information (MAC address, calibration data, regulatory skew information, etc.). The MAC/BB is also the only section that directly interfaces with the platform through the PCI Express* Mini Card interface.
2. The RFIC with an associated Xtal device used to generate all on-board frequencies and clock signals.
3. A bias network to enable bias selection, voltage regulation, and power management needed by the system.
4. An RF Front End (FE) section comprises antenna and transmit/receive switches, power amplifiers, low-noise amplifiers, and filtering devices.

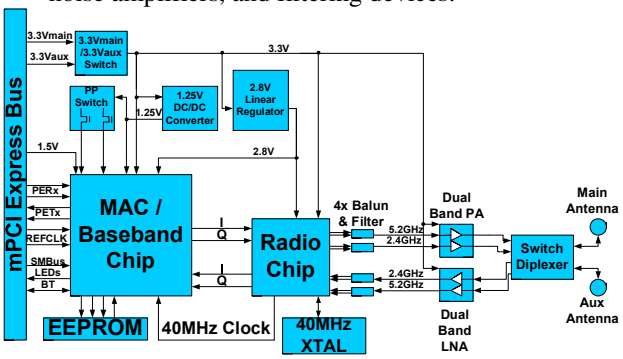


Figure 2: Block diagram of the board architecture

RFIC Architecture

The Si radio chip was designed using 0.18 um CMOS. The floorplan of the radio was designed together with the front end module (FEM) and package to minimize the spatial mismatch between the chip I/Os with the elements on the board. The architecture of the radio is shown in Figure 3 and is subdivided into six blocks: synthesizer, logic, RX BB, RX RF, TX BB, and TX/RF blocks. The logic section receives instructions from the MAC chip to set the system state. The instructions are then processed and output to the different blocks. The synthesizer is driven by an external clock. The loop basically can switch between two states to provide the LO signal for the low-band 802.11b/g (2.4-2.5 GHz) and the high-band 802.11a (4.9-5.95 GHz) up/down converters in the TX/RX chains. The dual-band RX chain consists of two separate RX architectures at the high- and low-bands, respectively. Note that the noise figure of the off-chip low noise amplifier (LNA) on board fixes the overall RX chain noise figure. A saturation detector at the input to the down converter is used to detect the power level in the receiver and if the power is too high, the attenuator at the input of the RX chain is triggered. The I/Q (in-phase and quadrature) signals from the down converter then pass into the BB section consisting of automatic gain control and active filters. The active filters will provide the out-of-band signal rejection depending on the system mode (CCK versus OFDM, etc.). The I/Q outputs from the BB section are passed off-chip via the board to the MAC chip.

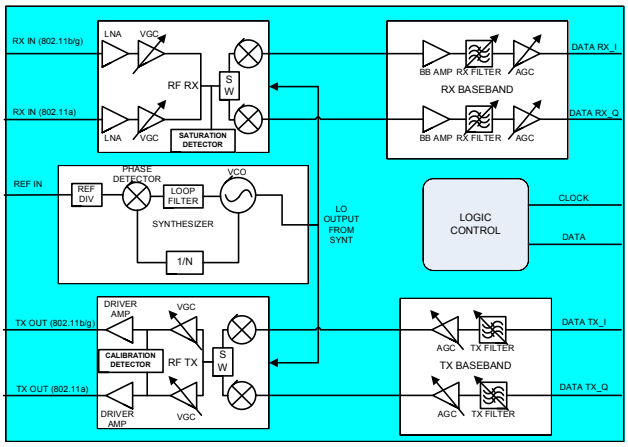


Figure 3: Block diagram of the RFIC architecture

On the TX side, the I/Q data signals from the MAC chip enter the radio chip via the TX BB section where they are filtered depending on the Wi-Fi standard needed, amplified via the digital gain control amplifier, and then upconverted to RF. The RF TX section consists of a high pass filter to reduce unwanted spurs at the lower bands followed by gain controlled amplifiers and a driver amplifier to the off-chip power amplifier (PA). A calibration detector at the input of the driver stage is used for calibrating the I/Q imbalance in the system.

Front End (FE) Architecture

The FE architecture shown in Figure 4 was basically defined at the system level which then drove the actual FE component content and requirements. The architecture chosen was a TX/RX architecture where the like TX and RX sections were all bundled together. This architecture also bundled similar technologies together according to the subgroups. For example, the PAs are of the same pseudomorphic high electron mobility transistor (PHEMT) GaAs technology, and the LNAs are of the same heterojunction bipolar transistor (HBT) GaAs technology. The filter passives are of a third technology and typically integrated in low temperature cofired ceramic (LTCC) technology. Therefore, the FE was subdivided into four category types: a) dual PA, b) dual LNA, c) switch-diplexer module, and d) balun filters. This "bundling" scheme ultimately created an industry trend of dual PA and LNA components appearing on the market place.

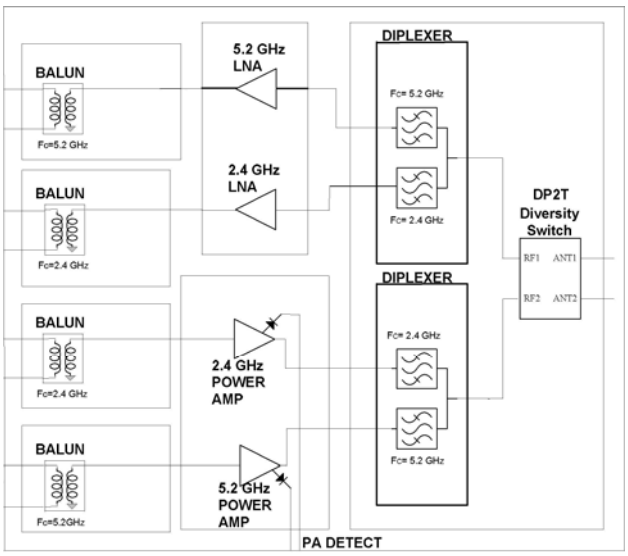


Figure 4: Front End architecture

DESIGN CONSIDERATIONS AND TRADEOFFS

System Tradeoffs

Once the marketing requirements for the product were released, the Intel PRO/Wireless 3945ABG Network Connection was partitioned down to the last detail. The architecture team developed the features and requirements of the different chips (MAC-BB, RF, and FE modules), while the system engineering team ensured that the different flows and features between the chips, board, and other components were synchronized and optimized. The two teams worked iteratively with each other such that the system team provided feedback on its findings/requirements to the architecture team in order for the architecture team to implement them in the chip design.

Another key point was to integrate the knowledge gained and avoid the pitfalls of previous projects. Since many of the system engineers led or were part of the integration of previous products, their input as to how to improve our product development approach was invaluable. Closing this loop was the only way to ensure that the next product integration cycle would be seamless and smooth.

There is a big difference between the design of the WLAN system of a few cards versus that of millions in production. All the manufacturing issues were taken into consideration from the start in order to guarantee a very robust and stable system. One important change was to utilize self-calibrations in order to compensate for component-to-component variance. From the early stages of the project, we obtained models about these variances and designed on-chip calibrations to compensate for them.

These calibrations guaranteed that millions of boards in production will behave with minimal performance deviations between them in accordance with the product specification. Ultimately, this contributed to very high yields in production. For example, in every WLAN system, the TX power needs to be calibrated. On the one hand, we want to transmit as much power as possible, but on the other hand, we cannot surpass regulatory limits. In previous projects, we used open loop calibration, but in the Intel PRO/Wireless 3945ABG Network Connection, we implemented a closed-loop TX power calibration shown schematically in Figure 5. It is important to note here that these self calibrations illustrate the kinds of things that can be done when designing the system from top to bottom. Other self calibration schemes were used to compensate for RF/analog parameter variances (I/Q imbalance, DC offset, gain, etc.), but for simplicity and cost reasons, these are compensated for in the digital domain.

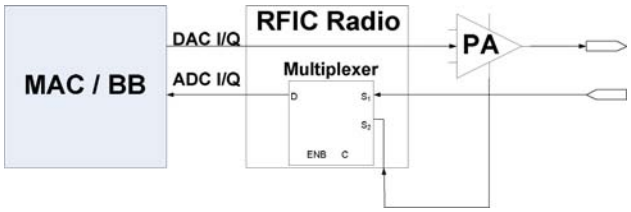


Figure 5: Closed-loop TX power calibration scheme

RFIC Design Considerations

The Radio Frequency Integrated Circuit (RFIC) designs used in the radio chip for the Intel Centrino Duo mobile technology are basically a concatenation of different functional blocks (amplifier, mixer, etc.) to form transmit or receive subsystems. The optimal (and real-estate savings) method for designing these subsystems would be to provide an impedance match between each of the blocks. Essentially the design then becomes one complete, standalone subsystem block. The other extreme is to design modular subsystem blocks, matching them all to the same impedance level and concatenating the blocks. Although not necessarily optimal in terms of real estate and power consumption, the system design is robust. Basic circuit theory dictates that if a design has a high impedance between two cascaded circuit blocks, the voltage is transferred between them. This is more useful for analog circuits and mixers. On the other hand, if the impedance is low, current is predominantly transferred between the two blocks. Microwave engineering usually looks at power transfer using impedance levels at 50 ohm, with power gain and match (return loss) as the primary metrics. If each of the blocks is matched to 50 ohm (or 100 ohm for differential circuits), these modular blocks can usually be seamlessly concatenated, without any significant increase in loss over the bandwidth in question. Moreover, these blocks can better withstand any changes in process variations or model inaccuracies; hence, they become predictable both in design and performance.

The prime consideration in our silicon RF CMOS designs was how to succeed in getting the product out to the market on time, reliably. In order to reduce design risk for this product, we leveraged the element of predictability described above. Thus, our approach here was to modularize all of the radio chip RF sub-blocks by designing each block independently to 50 ohm (or a 100 ohm differential) and to concatenate them. Analog designers typically utilize parasitic extraction to model coupling, capacitive loading, and fanout between the myriad of signal interconnects between circuit blocks. With RF designs at frequencies as high as 6 GHz and the need to check for harmonic performance and circuit stability up to 20 GHz, this "analog" approach is not accurate enough at these frequencies to account for distributed effects and unwanted coupling. Thus, the entire RF portions of the circuit layout were simulated in an electromagnetic (E/M) simulator to account for these effects. The output from the E/M simulator is then brought back into the circuit simulator for further analyses capturing all the E/M effects in the design simulations for maximum accuracy. This approach is illustrated in Figure 6.

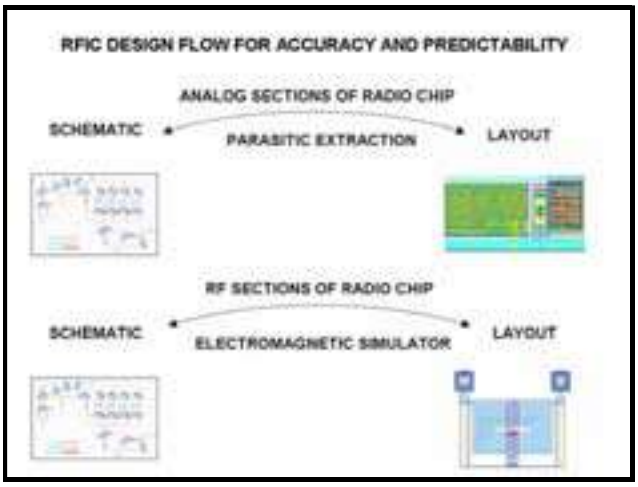


Figure 6: RFIC design flow for performance accuracy and predictability

Once the designed circuit blocks are concatenated into the RF RX/TX subsystems, which are then simulated, the circuit block and overall radio chip layout underwent reliability simulations such as aging, time dependent dielectric breakdown, IR drops in the power supply lines, and electromigration degradation checks in the inductors. For aging, the dominant aging effects of hot carrier injection in NMOS and PMOS, as well as negative bias temperature instability in PMOS transistors, were

considered. In the circuit simulations, the aged transistor models were then used to recompute the aged RF performance on the radio chip circuit blocks [1]. Electromigration effects were determined by parasitic extraction tools in looking at whether the inductor elements have enough metallization to carry the DC reliably.

Board Tradeoffs

The Intel PRO/Wireless 3945ABG Network Connection WLAN solution was required to fit on to a new FF, called Mini Card (PCI-SIG PCI Express Mini Card CEM), to be used on the Intel Centrino Duo mobile technology platform [2]. This new FF is about half the size of a Mini PCI card Type A used in the previous-generation WLAN solution called the Intel® PRO/Wireless 2915ABG Network Connection. An additional constraint was to have a single-sided solution, which meant that we had less than a quarter of the original board real estate available to implement the solution. This required us to rethink our board strategy and review our architecture. The new board architecture and strategy were driven by a number of key directives:

1. Keep it simple (simple and direct interfaces to optimize routing).
2. Optimize package selection and pinouts (to enable component butting).
3. Design for High-Volume Manufacturing (HVM) from the beginning.
4. Optimize for low-cost Printed Circuit Board (PCB) technology (minimum number of layers, through-hole vias, industry-standard design rules).
5. Optimize bias networks for simplest implementation (minimum number of power rails, unification of power rails, efficient power conversion schemes, minimum number of parts).
6. Outsource all components to multiple vendors to enable a significant cost reduction through competition between vendors.

Keeping the design as simplistic as possible is probably the main directive that drove the design. The intent was to design all of the components in such a way that their actual implementation on the board would be very much like the drawing of its block diagram. The interfaces between the different blocks/packages/silicon would need to coincide with each other to prevent unnecessary crossover routing on the board. All of the RF routing was limited to the top side only so as to minimize regulatory infringements (radiation, emissions) and enable easy certification by the various regulatory bodies. Keeping it simple also meant that we needed to drastically reduce the part count. The product requirement document clearly indicated that a single hardware (HW) skew would be supported, such that one HW build configuration made all of the logistics much simpler.

Another key directive was the optimum package selection for the silicon chipset where the pinouts between the MAC and RFIC chips were aligned to each other allowing for direct interconnect on the board. This approach enabled us to save significant board space.

The strategy of Design For Manufacturing (DFM) of HVM from the beginning of the design enabled us to identify potential limitations early in the program cycle. Once identified, the DFM infringements were dealt with in various manners. Some required a re-layout of specific sections on the product board. Some required the Manufacturing Engineering Group to re-examine outdated design rules for validity as the PCB fabrication and assembly technology matured over time. Some of the rules were modified quickly, and some rules required more extensive investigation including experimental assembly runs to check the validity of the rule intended to maintain low DPM requirements. By implementing the DFM/HVM rules from the beginning of the design, the 3945ABG was mass produced with very high production yields greater than 98.5%. This significantly reduced the product cost.

Product cost is always the driving factor in our development solutions. The PCB was found to be a significant cost percentage of the Bill of Material (BOM). As such we researched the PCB parameters that are "cost adders." We identified that the PCB material is a cost adder, whereas FR4-based materials are the cheapest. The type/class is another cost adder, whereas Through Hole Via (THV) technology would yield the lowest cost PCB. The number of layers and the exposed pad plating were also cost adders. Multiple vendors were requested to provide their technological capabilities with respect to the proposed PCB stackup and material. Once we found the lowest common denominator tolerances of multiple vendors, we specified the PCB so all vendors could comply with their existing capabilities and thus enabled cost reduction through competition and outsourcing to multiple vendors. The net result is that the Intel PRO/Wireless 3945ABG Network Connection incorporates a 4-layer FR4 with THV to achieve the lowest possible cost PCB solution that met our needs.

Our previous-generation WLAN board design had nine power rails, which was a tremendous cost adder because of the increased board real estate. The new FF Mini Card with the limited board real estate dictated that we re-examine the bias network strategy and drastically simplify it. The only way this could be done on the Intel PRO/Wireless 3945ABG Network Connection and the Si chipsets was to unify the bias voltages or generate them directly on die without adding any more circuitry to the board. Since the MAC and RFIC chips are fabricated on different CMOS processes, it was difficult to find a perfect unification of the power rails between the chips. A

compromise was reached that enabled the unification of the Analog/RF voltage with a single voltage to be used by both chips. In this way, the 2.8 V would be generated on the board by means of a linear regulator. The digital core voltage of the RFIC needs to be 1.8 V due to process requirements, but since it is a low current consumption rail, it was generated and regulated on-die. On the other hand, the core voltage of the MAC/BB chip (1.25 V) is one of the larger power consumers and therefore it was generated on-board by means of a DC/DC converter in order to save power. Another specific voltage is required for another dedicated circuit in the RFIC. The 2.5 V on-die regulator was dedicated to supply the voltage to the internal VCO, which requires a very clean and stable voltage supply to meet the RF performance of the system. Because our system is intended for very low-power consumption platforms, additional means were used to further reduce the leakage currents in idle and disabled modes of operation. The use of power plane separation switches enabled us to shut off the bias supply to the two largest digital circuit blocks (the MAC and PHY) to eliminate leakage currents. The final component in our bias network is a bias selection switch used to select between the 3.3 V main and the 3.3 V auxiliary power rails from the platform to enable wake-on WLAN (WoWLAN) modes during system (Sx) states. The 1.5 V power rail from the platform is dedicated to the PCI Express engine and connects directly to the MAC chip.

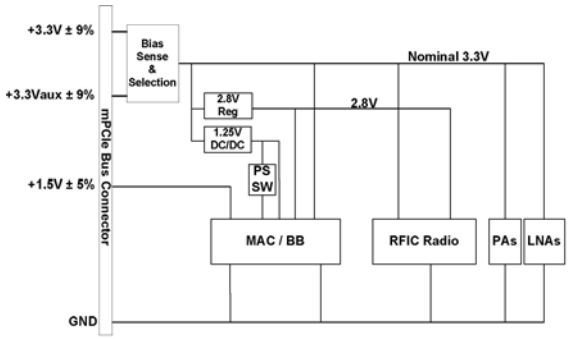


Figure 7: Bias network on the Intel PRO/Wireless 3945ABG Network Connection card

The other key components, such as the FE components, were specified by Intel as per our system needs. Specification documents were distributed among the potential vendors. Having multiple vendors providing us with the same "black box" component enabled us to outsource the product to multiple vendors to ensure a continuous supply base and a cost reduction through competition between the vendors. This strategy was initially targeted for the more expensive RF components (e.g., the power amplifiers, the low noise amplifiers, the switches and the passive filtering devices), but we expanded this strategy to cover other key components such as the EEPROM, the Xtal, and the bias network components. This multi-sourcing scheme not only enabled lower overall cost (through vendor competition), but also helped the design team to better understand the expected material variation resulting in a more robust "HVM-friendly" design that has built-in margins to compensate for expected material variations. Therefore, the design is less susceptible to yield fluctuations.

One of the key unknowns with respect to the new Mini Card FF was PCI Express noise leakage into the receiver and how this would affect product performance. The first generation PCI Express standard high-speed signals generate energy as high as our 2.4-2.5 GHz frequency range of operation. There was concern that leakage of the PCI Express signal from the Mini Card interface section may reach the RF/RX chain and interfere with reception and performance. A series of investigations were done to examine what critical items in the design needed to be done to minimize this leakage. This study included electromagnetic simulations of the package and bond wire scheme. After having done this investigation, the silicon design of the PCI Express engine needed to become a very compact and bias independent section of the die. This also affected the pinout, and it required that the PCI Express interfaces in the silicon be as close as possible to the Mini Card interface connection including a ground pin ring that encompassed the active pins to shield the PCI Express pins from the other sections of the package. A unified ground plane scheme in the PCB was used, which yielded the best isolation behavior from one end of the board to the other. The shielding scheme was limited to the radio section only to further reject any stray leakage which could leak into our sensitive receiver.

Other noise sources needed to be taken into account, such as voltage bounce and current surges associated with transitions from one state to another. These noises could adversely affect the performance of the product because they disrupt key parameters like the frequency stability of the local oscillator used for all of the up and down frequency conversions. Special care needed to be taken in the PCB layout to ensure proper ground return paths for key components, such as the power amplifiers. This was done by strategically placing decoupling capacitors in the entire circuit and sequencing the transition events (instead of instantaneous transitions) so as to reduce their effect on the key frequency sources in the system.

FEM Tradeoffs

The Mini Card FF drove us to re-examine how to save space on the board. One of the largest consumers of board space in the Intel PRO/Wireless 2915ABG Network Connection product generation was the RF FE section. Therefore, it became apparent that if the FE section did not shrink drastically, we would not be able to fit it into a single sided Mini Card FF. The sheer number of

components dictated that some sort of integration was required. The question was how much could be integrated without jeopardizing the program schedule or reliability. Going from a discrete solution to a fully integrated solution with multi-sourcing was too high a risk for the program. Therefore, we chose to implement a block-level integration scheme.

We also realized that going to this block-level integration would require defining custom parts that suited our system needs. When looking at the FE architecture, it became obvious that by bundling the technologies together, we also attained the best integration as well. For example, taking the two PAs (one for each band) and integrating them on the same GaAs die, along with their respective bias networks, into a single package device would ultimately yield the smallest solution. For example, the PA circuit alone, which was over 20 components in the Intel PRO/Wireless 2915ABG Network Connection generation, was shrunk down to five components (one integrated Dual Band PA and four decoupling caps) in the Intel PRO/Wireless 3945ABG Network Connection solution by the use of building blocks.

The same approach was also used to shrink other sections of the FE such as the dual LNA device, the switch-diplexer module, and the two balun filter devices. The integration of these components in this manner enabled us to implement a cheaper, single-sided solution for our Intel PRO/Wireless 3945ABG Network Connection WLAN product. The net effect of this FE module shrinkage is outlined in red in Figure 8, going from the FE of the previous generation (Intel PRO/Wireless 2915ABG Network Connection) to the current Intel PRO/Wireless 3945ABG Network Connection product.

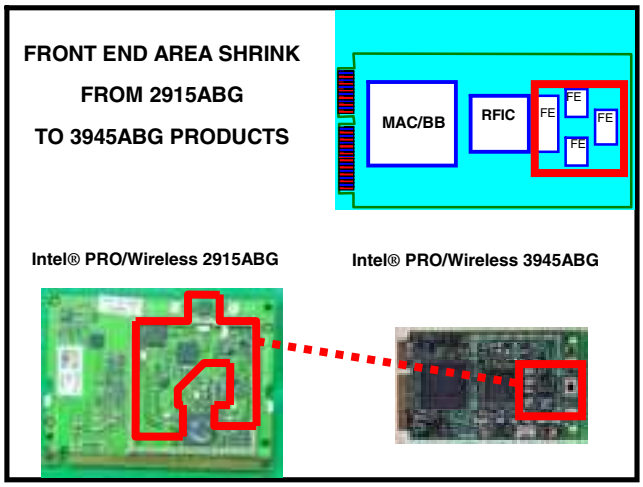


Figure 8: Front End module area compression from 2915ABG to 3945ABG

The main performance tradeoff between the various FE components was the out-of-band behavior. It was obvious that if we required one component to provide the entire out of band rejection needed for the system performance, it would make that particular component either very expensive, very large, or both. Therefore, when considering the RF line-up definition for the in-band performance, we also needed to distribute these out-of-band requirements between the three basic sections (balun filter, the active device, and the diplexer). Thus, we were able to simplify the requirements for each of the sections while achieving the overall system requirement. In some specialized cases, more weight needed to be placed on one section over another to compensate for known deficiencies or to maximize performance. For example, the diplexer rejection at frequencies below 2 GHz was stricter to prevent the LNA from saturating due to cell phone activity near the notebook platform. On the other hand, second- and higher-order harmonic signals needed to be reduced out of the PA to meet regulatory certification requirements, a requirement on the diplexer out of the PAs.

MANUFACTURING TESTING

One of the main bottlenecks in reducing product cost and time to market was the need to significantly reduce production testing time. The new approach described here for the Intel PRO/Wireless 3945ABG Network Connection enabled a dramatic reduction in test time by a factor of 3 from 104 to 32 seconds per card at the functional testing stage as compared to the previous WLAN product (the Intel PRO/Wireless 2915ABG Network Connection).

The Wireless HVM process flow involves several stages (see Figure 9), each of which is capable of detecting specific failures.

Structural Test uses X-ray laminography technology, which provides a virtual 3D image "slice" that blurs out all but the selected plane of focus. These 3D cross-section images of solder joints are then evaluated.

Functional Test (FT) station is the first stage where the Intel PRO/Wireless 3945ABG Network Connection device is inserted into the PC. FT has several goals:

1. Check for HW failures of the system and sub-components.
2. Characterize the device's physical parameters that have an impact on transmit and receive path performance. Tests belonging to this category are called *calibration procedures.*

3. EEPROM programming: HW definitions and results of the calibration tests are stored in the device's EEPROM.

The next stage of the HVM flow is **Final Assembly**. This is the stage where the wireless device is packed and wrapped with a Label Paper sticker.

The **Final Tester** is used to verify the programming of the EEPROM, including the MAC address for the device, conduct performance tests on 802.11a/b/g radios, and an association test with an Access Point (AP). After the Final Test is performed, samples of passed units are tested in Ongoing Quality Monitoring (OQM).

The **OQM Station** runs the same tests as in the Final Tester, but on a defined percentage of the manufactured product. The main requirement of the whole HVM process is that at this station, the Defects Per Million (DPM) level will be below 500 DPM.

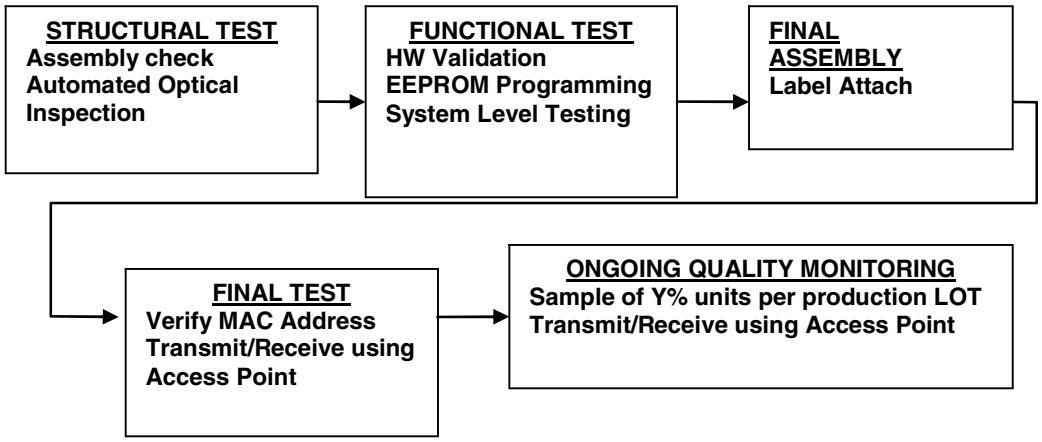


Figure 9: Intel PRO/Wireless 3945ABG Network Connection high-volume manufacturing process flow

Testing time and coverage are a major component of the manufacturing cost and quality of the wireless device, since they have a direct impact on the amount of equipment and testers that is used in the manufacturing line. In order to achieve so dramatic a reduction in testing time relative to our previous wireless products, several approaches were taken:

1. **Working over Product Network Driver (NDIS).** Our previous Intel® Wireless LAN products were tested using a specialized version of Validation Driver that was used to directly control the embedded software running on the product board. In the Intel PRO/Wireless 3945ABG Network Connection project, the role of the network driver was significantly enhanced and most of the features previously handled by the embedded software were moved to the driver level. Therefore, it made sense that the test and validation during FT should also be redone. This ultimately yielded a shorter test time of the product board. For example, the reset test flow through the Network Driver was optimized for speed. Over the Network Driver, it was 10 times faster than testing over an application layer-based test.

2. **Utilizing driver system flows**. Additional time savings were achieved by optimizing the time of validation flows. Using the Network Driver also made it possible to take advantage of real WLAN system-flows, such as online DC and TxIQ calibrations, and checking the embedded software image after reset. This enabled us to remove application-layer tests that did not contribute to the test suite coverage.

3. **Using** a **closed-loop TX power calibration algorithm.** By using this we dramatically reduced the amount of required measurement points in the FT tester, as compared to previous projects. This allowed the calibration to be performed in a much shorter time (by a factor of 2.5).

4. **Speeding up the equipment.** A special and dedicated effort was made to find the test equipment bottlenecks and to speed up the RF testing equipment used in the FT. Most of the time savings came from speeding up the Power Meter and Spectrum Analyzer equipment data acquisitions.

With regard to test coverage, in our previous projects of wireless HVM testing, there was always doubt as to how close the resemblance was between the driver and the FT software, since they represent two different flows. Using this new approach, this difference was eliminated because the FT was actually running using the same driver as the

product, giving us confidence in test quality and reliability. Overall test coverage was increased due to the fact that we were using the Product Network Driver in executing real-life system flows, such as online calibrations, real-time Tx/Rx flows, reset flow, and up-to-date PHY initialization routines. Having the network driver in manufacturing allowed logging of various system parameters across all manufactured boards, which helped us to monitor product health across builds.

Overall, by using the network driver for the Intel PRO/Wireless 3945ABG Network Connection FT made significant improvements to the Functional Tester Development Program in terms of time and coverage, and this technique will be used in our future Intel® Wi-Fi products.

CONCLUSION

This paper summarizes and reviews the work and design approach of many groups in the development of the Intel Centrino Duo mobile technology Intel PRO/Wireless 3945ABG Network Connection card for the Intel Centrino Duo mobile technology platform. The combination of taking a top-down approach to the system design, the application of self-calibration schemes, the modularity approach taken in the RFIC design, the use of custom board and FE elements to reduce part count, and the significant improvement of production testing time, has resulted in a reduction of platform integration from 6 to 4 quarters with product yields better than 98.5% with lower product cost and a smaller FF. The same approach is now being applied to our next-generation wireless products.

ACKNOWLEDGMENTS

The work described in the paper was made possible by the contributions of many people. We acknowledge Shmuel Ravid, Tzahi Hod, Yaniv Mazor, Yossi Levi, Leonardo Hilel, Alla Gevorkov, Richard Perry, James Mowery, Eli Rajchert, Eli Laks, Jacob Solomon, David Carasso, Ayelet Alon, Shay Dubrovski, Shaul Shayevich, Hila Pinchas, Tzvi Maimon, Yishai Eilat, Sharon Betzalel, Georgi Normatov, Shai Gross, Sharon Zaguri, and Assaf Ben-Bassat.

A special thanks to all the teams that spent many long hours to make the launch of the Intel Centrino Duo mobile technology Intel PRO/Wireless 3945ABG Network Connection a reality.

REFERENCES

[1] M. Ruberto, T. Maimon, Y. Shemesh, A. Desormeaux, W. Zhang, C. Yeh, "Consideration of Age Degradation in the RF Performance of CMOS Radio Chips for High Volume Manufacturing," *IEEE RFIC Symposium*, Long Beach, CA, June 2005, pp. 549-552.

[2] "PCI Express Mini Card Electromechanical Specification," Revision 1.1; March 28, 2005, PCI-SIG.

AUTHORS' BIOGRAPHIES

Mark Ruberto received his Ph.D. degree in Electrical Engineering from Columbia University in New York in 1991. He worked as a postdoctoral research scientist at the Technion in Haifa in the area of microwave optoelectronics. After 10 years as an RF and Microwave design engineer in various Israeli companies, he joined Intel in 2003 as a senior RFIC design engineer. Mark designs RF CMOS circuits for Intel's wireless products. He has also led projects in the RF model/collateral development for an Intel® RF process, and in the integration of an RF reliability validation flow into the design of Intel's CMOS radio chips. Mark authored over 30 technical papers, holds one patent, and is a Senior Member of the IEEE. His e-mail is mark.ruberto at intel.com.

Ra'anan Sover received his B.Sc.E.E. degree from the Technion, Israel in 1988. From 1988-2000, he was employed by MicroKim Ltd. as a senior RF engineer. He developed multifunction Hybrid RF synthesizers and control devices. From May 2000 he has worked at Intel as a senior RFIC design engineer and has co-designed the synthesizer block of Intel's first WLAN RFIC. He now serves as the HW/RF architect of the RF System and Integration Team of WNG. He is a Senior Member of the IEEE. He recently co-authored the award winning "Best of DTTC" paper for 2005 entitled "Silicon-backplane-based integrated RF front-end modules for WLAN applications." His e-mail is raanan.sover at intel.com.

Jorge Myszne received his B.Sc. degree in Electrical Engineering from "Universidad Mayor de la Republica" in Montevideo, Uruguay in 1998. He worked for almost three years in MicroSur Ltd. and joined Intel in 2000. He worked in the DSP/PHY group for four years as a senior DSP engineer and in 2004 he moved to system engineering. He was responsible for the 3945ABG product integration and is now the system engineer leading Intel's future WLAN products. Jorge authored and co-authored several technical papers and holds seven pending patents. His e-mail is jorge.myszne at intel.com.

Alexander Sloutsky is the software lead in the RF System Testing Intel Wireless group. Alexander graduated from Israeli Technion University with a B.Sc. degree in Computer Engineering. He started his career at Intel in the processors department as a student in the 3D HW design group seven years ago. He joined the Wireless

Networking Department in 2000 and started his work as a software engineer in the System Validation Tools group. From 2003, his team has been responsible for the development of functional tests for wireless manufacturing and infrastructures for hardware validation and testing. His e-mail is alexander.sloutsky at intel.com.

Yair Shemesh received the his B.Sc.E.E. (Cum Laude) degree in 1985) and his M.Sc.E.E. degree from the Technion in Israel. From 1986 to 2000 he worked for RAFAEL, the Israeli armaments development agency. There, he was a microwave engineer and later a team leader developing microwave circuits and modules in various technologies. Yair designed numerous MMICs with all the major foundries of the time. From 1994 to 2002 he was employed by Alpha Industries Inc, Woburn, MA, (Now Skyworks Inc.) as a senior principal engineer responsible for development of GaAs RFICs and modules mostly for the high-volume cellular market. In 2002 he joined Intel in Haifa Israel where he manages the RFIC design team, developing all Intel's WLAN radio chips. Yair authored and co-authored over 20 technical papers and patents and is a senior member of the IEEE. His e-mail is yair.shemesh at intel.com.

Copyright © Intel Corporation 2006. All rights reserved. Intel and Centrino are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

* Other names and brands may be claimed as the property of others.

This publication was downloaded from http://developer.intel.com/.

Legal notices at http://www.intel.com/sites/corporate/tradmarx.htm.

MIMO Architecture for Wireless Communication

Ilan Hen, Mobility Group, Intel Corporation

Index words: Alamouti scheme, beamforming, capacity, error probability, fading, inter-symbol-interference, LDPC codes, MIMO channels, MRC, OFDM

ABSTRACT

In this paper, we consider the use of multiple-input multiple-output (MIMO) architecture for wireless communication systems. We show how by employing MIMO architecture, architecture in which transmission and reception are carried out through multiple antennas, one can design a superior wireless communication system with respect to reliability, throughput, and power consumption.

INTRODUCTION

In recent years, wireless communication devices have become more and more popular. However, at the same time, the design of faster, more reliable, and power-efficient wireless communication systems has become evermore difficult. Wireless channels, as opposed to wireline channels, exhibit highly irregular amplitude behavior due to what is known as *fading*. The fading, essentially caused by the reception of multiple reflections of the transmitted signal (illustrated in Figure 1), is a key inherent problem of wireless channels, which, unfortunately, cannot be avoided. Fading causes the received signal power to change rapidly in time, making the task of information extraction from the received signal a fairly complicated endeavor. Furthermore, once the information is extracted, its reliability, manifested through error probability, is often poor.

In this paper, we demonstrate how by exploiting the spatial diversity, namely, using multiple antennas, one can improve reliability, increase transmission throughput, and reduce transmission power. We also briefly discuss the benefits of using MIMO architecture along with orthogonal frequency division multiplexing (OFDM) modulation, and low-density parity check (LDPC) coding.

First, we consider the reliability issue. We present a basic model for the wireless single-input single-output (SISO, single transmit antenna, single receive antenna) channel, and show how the corresponding error probability is critically damaged by fading. We then consider the single-input multiple-output (SIMO, single transmit antenna, multiple receive antennas) channel and describe the concept of maximal ratio combining (MRC) as a way to exploit the receive diversity offered by this type of channel. We calculate the error probability achieved by the MRC, showing it to be much smaller than the one corresponding to the SISO channel, in which no spatial diversity exists. Next, we consider the multiple-input single-output (MISO, multiple transmit antennas, single receive antenna) channel, and we present some mechanisms that exploit the transmit diversity offered by this channel. Specifically, the beamforming technique and Alamouti's [3] scheme are analyzed. Bringing together transmit and receive diversity, the MIMO channel is introduced. The beamforming technique and Alamouti-based scheme are shown to achieve full diversity, i.e., they take full advantage of both transmit and receive diversity provided by the MIMO channel. We discuss the performance of the aforementioned spatial diversity techniques, and we draw some conclusions as to when one should be preferred over the other.

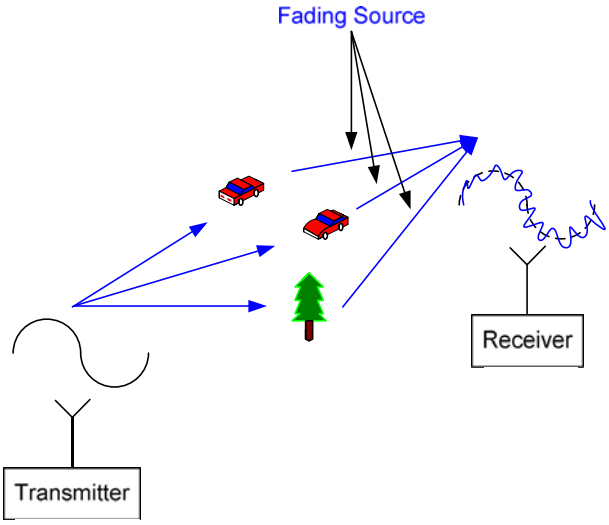


Figure 1: Wireless channel–fading problem due to multiple reflections

Improved reliability is not the only outcome of using multiple antennas. About ten years ago, a remarkable theoretical result regarding the capacity of MIMO channels [1] suggested that the transmission rate over wireless channels can be dramatically increased when using multiple antennas. It turns out that the ability to transmit and receive through multiple antennas does not only reject fading; better yet, it actually harnesses the fading itself in favor of increased throughput. We present the capacity of wireless MIMO channels, showing it to be greater than that of the wireline SISO channel. Moreover, the capacity formula is used to demonstrate how one can reduce transmission power by using multiple antenna systems.

Finally, we briefly explain how, in principal, integration between MIMO architecture, OFDM modulation, and LDPC coding (essentially the basic building blocks for the most advanced wireless communication standards, e.g., 802.11n, 802.16), can give rise to a superior wireless communication system.

The outline of the paper is as follows. We first present a basic model for the wireless SISO channel. We then consider the MRC technique, the beamforming technique, and Alamouti's scheme. After that, we present the capacity of MIMO channels and then discuss the benefits of using MIMO architecture along with OFDM modulation and LDPC coding. We conclude with some remarks on the performance of MIMO architecture and how the Intel® Centrino® mobile technology benefits by embracing it.

THE WIRELESS CHANNEL

In this section, we present a basic model for the wireless channel and show its performance to be inferior to that of the wireline channel.

The traditional wireline channel is modeled by the equation

$$y = x + n \tag{1.1}$$

where

x is the channel input. x is a complex number, referred to as *symbol*, representing two bits of information, i.e., it can take up to four different values according to the mapping (in general, x may be chosen to represent more than two bits of information):

$$\begin{aligned} "00" &\rightarrow x = -\sqrt{E_s} - j\sqrt{E_s} \\ "01" &\rightarrow x = -\sqrt{E_s} + j\sqrt{E_s} \\ "10" &\rightarrow x = \sqrt{E_s} - j\sqrt{E_s} \\ "11" &\rightarrow x = \sqrt{E_s} + j\sqrt{E_s} \end{aligned} \tag{1.2}$$

We assume that all the above possible realizations of x are equally probable.

n is the channel noise, accounting for the thermal noise induced by different parts of the receiver. n is modeled as a zero mean, complex Gaussian random variable with variance σ^2 per dimension, i.e., the real part of n and the imaginary part of n are zero mean, statistically independent Gaussian random variables with variance σ^2. Note that

$$\begin{aligned} En &= 0 \\ E|n|^2 &= E(nn^*) = 2\sigma^2 \end{aligned} \tag{1.3}$$

"E" and "*" denote statistical expectation and complex conjugate, respectively. The channel signal-to-noise-ratio (SNR) is given by

$$\begin{aligned} SNR &= \frac{E|x|^2}{E|n|^2} \\ &= \frac{E_s}{\sigma^2}. \end{aligned} \tag{1.4}$$

The receiver observes the channel's output y and decides which symbol, out of the four possible ones, was sent. The receiver tries to produce decisions with the best possible reliability. We measure reliability through error probability. Let $\hat{x}(y)$ be the receiver decision. The error probability, denoted here by $P_r\{\varepsilon\}$, is the probability that $\hat{x}(y)$ is different than x,

$$P_r\{\varepsilon\} = P_r\{\hat{x}(y) \neq x\}. \tag{1.5}$$

What receiver should we be using in order to achieve minimal error probability? The optimal receiver [2] decides that symbol x was sent, if the Euclidian distance between x and y is the smallest among all possible distances (total of four in our case). The optimal receiver, referred to as *the maximum likelihood (ML)* receiver, achieves error probability [2] satisfying

$$P_r\{\varepsilon\} \le \exp\left\{-\frac{SNR}{2}\right\}. \tag{1.6}$$

Throughout the paper, we provide upper bounds on the error probability. However, these bounds are tight for the mid-to-high SNR range [2], which is the interesting SNR range when targeting reliable, high-throughput communication. As we can see, for the wireline channel, the error probability decreases exponentially fast with the SNR. Does this excellent behavior remain intact when transmitting over wireless channels? Unfortunately, the answer is no. The fading, as shown next, dramatically increases the error probability.

The wireless channel model is similar to the wireline channel model, but with the input amplitude modified to account for the fading. The wireless channel is modeled with the equation

$$y = hx + n \tag{1.7}$$

where h represents the fading. We consider an environment in which there is no line of sight (NLOS) between the transmitter and the receiver. For this kind of environment, h is modeled as a zero mean, complex Gaussian random variable with variance 0.5 per dimension. Suppose for a moment, that the fading h is a fixed deterministic number rather than a random variable. In that case, the channel SNR would be given by

$$\begin{aligned} SNR(h) &= \frac{E|hx|^2}{E|n|^2} \\ &= SNR|h|^2 \end{aligned} \tag{1.8}$$

and, as for the wireline channel, the error probability satisfies

$$\begin{aligned} P_r\{\varepsilon \mid h\} &\le \exp\left\{-\frac{SNR(h)}{2}\right\} \\ &= \exp\left\{-\frac{|h|^2\, SNR}{2}\right\}. \end{aligned} \tag{1.9}$$

Let

$$z = |h| \tag{1.10}$$

then

$$P_r\{\varepsilon \mid z\} \le \exp\left\{-\frac{z^2 SNR}{2}\right\}. \tag{1.11}$$

The above result accounts for the error probability for a given realization of the fading h. In order to obtain the error probability $P_r\{\varepsilon\}$ we must average the conditioned error probability (1.11) with respect to the probability low of z

$$P_r\{\varepsilon\} = \int P_r\{\varepsilon \mid z\} f(z) dz \tag{1.12}$$

$f(z)$ is the probability density function of z. Since h is Gaussian distributed, z is the squared root of the squared sum of two independent Gaussian random variables, which means [2] z is *Rayleigh* distributed

$$\begin{aligned} P_r\{z \le a\} &= \int_0^a f(\beta)\, d\beta \\ f(\beta) &= 2\beta \exp\{-\beta^2\}, \quad \beta \ge 0. \end{aligned} \tag{1.13}$$

Averaging (1.11) with respect to the Rayleigh distribution we have

$$\begin{aligned} P_r\{\varepsilon\} &\le \int_0^\infty \exp\left\{-\frac{z^2 SNR}{2}\right\} 2z \exp\{-z^2\} dz \\ &= \frac{1}{1+\frac{SNR}{2}}. \end{aligned} \tag{1.14}$$

It should be clear now how severe the damage caused by the fading is: instead of having an error probability that decreases exponentially fast with the SNR (1.6), we have an error probability which is only inversely proportional to the SNR. In the next section, we show how by using multiple antennas the situation can be rectified to some extent.

MIMO SYSTEMS RELIABILITY

In this section, we consider various spatial diversity techniques aimed at reducing the error probability.

Receive Diversity

Consider the SIMO channel depicted in Figure 2.

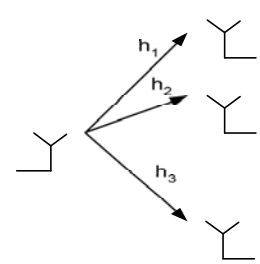


Figure 2: SIMO channel

Let N be the number of receive antennas. The signal received in antenna i is given by

$$y_i = h_i x + n_i, \quad i = 1, 2, \cdots, N \tag{1.15}$$

where h_i and n_i are the fading and noise, respectively, as experienced by antenna i. We assume the fading is independent, which is the case, provided the antennas are sufficiently spaced from each other.

Consider the following weighted combination of the antennas' inputs

$$\begin{aligned} y &= \sum_{i=1}^{N} \alpha_i y_i \\ &= x \sum_{i=1}^{N} \alpha_i h_i + \sum_{i=1}^{N} \alpha_i n_i \end{aligned} \tag{1.16}$$

where the α_i's are some deterministic numbers. The SNR of the above channel is given by

$$\begin{aligned} SNR(h_1, \cdots, h_N) &= \frac{E\left| x \sum_{i=1}^{N} \alpha_i h_i \right|^2}{E\left| \sum_{i=1}^{N} \alpha_i n_i \right|^2} \\ &= SNR \frac{\left| \sum_{i=1}^{N} \alpha_i h_i \right|^2}{\sum_{i=1}^{N} |\alpha_i|^2} \end{aligned} \tag{1.17}$$

It is straightforward to verify (applying the Cauchy-Schwartz inequality) that by setting

$$\alpha_i = h_i^* \tag{1.18}$$

the SNR is maximized. The weighted combination of the antennas' inputs with the above α_i's is referred to as *maximal ratio combining*. Substituting (1.18) into (1.17), the maximal SNR is given by

$$SNR(h_1, \cdots, h_N) = SNR \sum_{i=1}^{N} |h_i|^2 . \tag{1.19}$$

Thus, the error probability obtained by the ML receiver when applied to the MRC output satisfies

$$\begin{aligned} P_r\{\varepsilon \mid h_1, \cdots, h_N\} &\le \exp\left\{ -\frac{SNR(h_1, \cdots, h_N)}{2} \right\} \\ &= \exp\left\{ -\frac{SNR \sum_{i=1}^{N} |h_i|^2}{2} \right\}. \end{aligned} \tag{1.20}$$

Let

$$z_i = |h_i|, \quad i = 1, 2, \cdots, N \tag{1.21}$$

then

$$P_r\{\varepsilon \mid z_1, \cdots, z_2\} \le \exp\left\{ -\frac{SNR \sum_{i=1}^{N} z_i^2}{2} \right\}. \tag{1.22}$$

z_i's are statistically independent, Rayleigh distributed random variables. Thus, their joint density is simply given by the product of their individual densities

$$f(z_1, \cdots, z_2) = \prod_{i=1}^{N} 2 z_i \exp\{-z_i^2\}. \tag{1.23}$$

Averaging (1.22) with respect to (1.23) yields

$$\begin{aligned} P_r\{\varepsilon\} &\le \int_0^\infty .. \int_0^\infty \exp\left\{ -\frac{SNR \sum_{i=1}^{N} z_i^2}{2} \right\} \\ &\quad \times \prod_{i=1}^{N} 2 z_i \exp\{-z_i^2\} \, dz_1 .. dz_N \\ &= \frac{1}{\left(1 + \frac{SNR}{2}\right)^N}. \end{aligned} \tag{1.24}$$

As we can see, by using N receive antennas we have managed to substantially reduce the error probability. Note that in order to perform MRC, the receiver has to know the fading, or, in other words, the receiver has to have access to the *channel state information* (CSI). This is usually done by sending some known signal through the channel, called *pilot*, and measuring the channel's response. Cleary, such a procedure does not allow for having perfect CSI, but rather approximate CSI. However, empirical results indicate that using MRC with

approximate CSI, instead of perfect CSI, slightly deteriorates performance. In general, the performance of spatial diversity techniques is measured using two terms: *diversity order*

$$\text{diversity order } = -\lim_{SNR\to\infty} \frac{\ln P_r\{\varepsilon\}}{\ln SNR} \quad (1.25)$$

and *antenna gain*

$$\text{antenna gain } = \frac{E\left(SNR\left(h_1,\cdots,h_N\right)\right)}{SNR}. \quad (1.26)$$

MRC achieves diversity order of N and antenna gain of N. If we draw a curve of the error probability as a function of the SNR on the logarithmic axis, the diversity order is the slope of the curve, and the antenna gain is the left-hand horizontal shift of the curve with respect to the curve

$$\frac{1}{\left(1+\frac{1}{N}\frac{SNR}{2}\right)^N}.$$

In some cases, it is not practical to have multiple antennas at the receiver. Consider for example handled devices: their small form factor does not allow for the positioning of multiple antennas that are spaced far enough from each other. Once the antennas are close, the fading seen by them is not independent, and then the error probability can not be made small as indicated by (1.24). Can we achieve the performance of MRC but with multiple antennas at the transmitter? The answer is yes. Essentially, the reason that MRC works is that it increases the SNR. By applying transmit diversity we can also increase the SNR and in turn decrease the error probability.

Transmit Diversity

Consider the MISO channel depicted in Figure 3. Let M be the number of transmit antennas. The received signal is given by

$$y = \sum_{j=1}^{M} h_j x_j + n \quad (1.27)$$

where h_j is the fading corresponding to transmit antenna j, and x_j is the symbol sent through antenna j. Again, we assume that the fading is independent. Suppose that we transmit

$$xw_j, \quad j = 1,2,\cdots,M \quad (1.28)$$

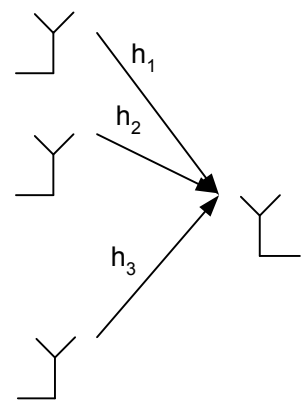


Figure 3: MISO channel

where w_j's are some weighting factors satisfying

$$\sum_{j=1}^{M} \left|w_j\right|^2 = 1. \quad (1.29)$$

The above constraint ensures we are not increasing the transmission power. Substituting (1.28) into (1.27) we have

$$y = x\sum_{j=1}^{M} h_j w_j + n. \quad (1.30)$$

The SNR of the above channel is given by

$$SNR\left(h_1,\cdots,h_M\right) = \frac{E\left|x\sum_{j=1}^{M} h_j w_j\right|^2}{E\left|n\right|^2} = SNR\left|\sum_{j=1}^{M} h_j w_j\right|^2 \quad (1.31)$$

As before, we would like to maximize the SNR. Setting

$$w_j = \frac{h_j^*}{\sqrt{\sum_{j=1}^{N}\left|h_j\right|^2}} \quad (1.32)$$

the SNR is maximized to the value

$$SNR\left(h_1,\cdots,h_M\right) = SNR\sum_{j=1}^{M}\left|h_j\right|^2. \quad (1.33)$$

Following the exact same steps as in the case of the MRC, we readily obtain

$$P_r\{\varepsilon\} \le \frac{1}{\left(1+\frac{SNR}{2}\right)^M}. \qquad (1.34)$$

The procedure described in equation (1.28) with the optimal weighting factors of (1.32) is referred to as *transmit beamforming*. It is so named because the signal x is being formed before being transmitted. Transmit beamforming achieves a diversity order of M and an antenna gain of M, the same as MRC with M receive antennas. However, note that for transmit beamforming, the transmitter must have the CSI. This presents us with a bit of a problem, since in order for the transmitter to have the CSI, the receiver must send it to the transmitter, unavoidably reducing the throughput. Can we achieve transmit diversity without having to provide the transmitter with the CSI? Yes, we can, using Alamouti's scheme.

Alamouti's scheme consists of two transmit antennas and one receive antenna. It achieves the error probability

$$P_r\{\varepsilon\} \le \frac{1}{\left(1+\frac{SNR}{4}\right)^2} \qquad (1.35)$$

while only requiring CSI to be at the receiver. It does so by employing transmission and reception mechanisms stretched across space and time. Alamouti's scheme achieves a diversity order of 2 and an antenna gain of 1, as opposed to an antenna gain of 2 for MRC 1×2 and transmit beamforming 2×1. This means that MRC 1×2 and transmit beamforming 2×1 outperform Alamouti's scheme by $3db$ (the SNR term in (1.35) is divided by 4 and not 2 as for MRC and transmit beamforming). However, as explained earlier, MRC needs the antennas to be sufficiently spaced, and transmit beamforming needs to know the CSI at the transmitter.

Transmit/Receive Diversity

Consider the MIMO channel depicted in Figure 4. Let M and N be the number of transmit and receive antennas, respectively. The received signal at antenna i is given by

$$y_i = \sum_{j=1}^{M} h_{ij} x_j + n_i, \quad i = 1, 2, \cdots, N. \qquad (1.36)$$

h_{ij} is the fading corresponding to the path from transmit antenna j to receive antenna i.

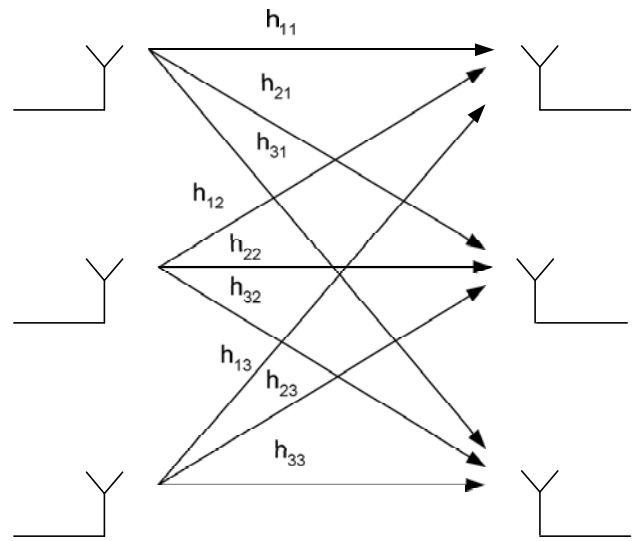


Figure 4: MIMO channel

As before, we assume the fading is independent. n_i is the noise corresponding to receive antenna i. Let

$$\underline{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}, \quad \underline{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_M \end{bmatrix}, \quad \underline{n} = \begin{bmatrix} n_1 \\ n_2 \\ \vdots \\ n_N \end{bmatrix}, \qquad (1.37)$$

$$H = \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1M} \\ h_{21} & h_{22} & \cdots & h_{2M} \\ \vdots & \vdots & \cdots & \vdots \\ h_{N1} & h_{N2} & \cdots & h_{NM} \end{bmatrix}. \qquad (1.38)$$

then

$$\underline{y} = H\underline{x} + \underline{n}. \qquad (1.39)$$

We now describe the procedure of transmit/receive beamforming. The transmitter sends

$$\underline{v}x \qquad (1.40)$$

where $\underline{v}$ is a vector of size $M\times 1$ satisfying

$$E\left(\underline{v}^*\underline{v}\right) = 1. \qquad (1.41)$$

For vectors and matrices "*" denotes the Hermitian conjugate, i.e., the vector, or matrix, is first transposed and then complex conjugated, entry by entry. The received signal is then

$$\underline{y} = H\underline{v}x + \underline{n}. \qquad (1.42)$$

The receiver multiplies the received signal with a $N \times 1$ sized vector $\underline{u}$, creating the channel

$$\underline{u}^* \underline{y} = \underline{u}^* H \underline{v} x + \underline{u}^* \underline{n}. \quad (1.43)$$

The above channel SNR is given by

$$\begin{aligned} SNR(H) &= \frac{E\left|\underline{u}^* H \underline{v} x\right|^2}{E\left|\underline{u}^* \underline{n}\right|^2} \\ &= SNR \frac{\left|\underline{u}^* H \underline{v}\right|^2}{\left|\underline{u}^* \underline{u}\right|^2} \end{aligned} \quad (1.44)$$

How should one choose $\underline{v}$ and $\underline{u}$ such that the SNR is maximized? The maximizing vectors are derived from the *singular value decomposition (SVD)* of H [4], and the maximal SNR satisfies

$$\frac{1}{\min\{N,M\}} SNR \|H\|^2 \le SNR(H) \le SNR \|H\|^2 \quad (1.45)$$

where

$$\|H\|^2 = \sum_{i=1}^{N} \sum_{j=1}^{M} \left|h_{ij}\right|^2 . \quad (1.46)$$

The error probability is then bounded by

$$P_r\{\varepsilon \mid H\} \le \exp\left\{-\frac{SNR\|H\|^2}{2\min\{N,M\}}\right\}. \quad (1.47)$$

Averaging the error probability with respect to the fading yields

$$P_r\{\varepsilon\} \le \frac{1}{\left(1+\frac{SNR}{2\min\{N,M\}}\right)^{MN}}. \quad (1.48)$$

Thus, for transmit/receive beamforming we have a diversity order of MN, referred to as *full diversity*. The antenna gain on the other hand satisfies

$$\max\{M,N\} \le \text{antenna gain} \le MN . \quad (1.49)$$

Transmit/receive beamforming requires CSI at the receiver as well as in the transmitter. For a 2×2 setting, transmit/receive diversity can also be achieved by using 2×2 Alamouti-based scheme (obtained by an extension of the Alamouti's scheme) which achieves a diversity order of 4, an antenna gain of 2, and requires CSI only at the receiver. In Table 1, we summarize the antenna gain and diversity order for the different channel configurations.

MIMO SYSTEMS CAPACITY

In this section, we present the capacity of wireless MIMO channels and show that it is greater than that of the wireline SISO channel.

The capacity of a communication channel is the maximum throughput at which data can be sent over the channel while maintaining a low probability of error. The capacity is measured in bits per channel use. Clearly, we would like to transmit over channels with high capacity.

Table 1: Diversity order and antenna gain for various spatial channels

System	Antenna Gain	Diversity Order
SISO	1	1
SIMO CSI RX (MRC 1xN)	N	N
MISO CSI RX (ALAMOUTI 2x1)	1	M
MISO CSI RX & TX (TX BEAMFORMING Mx1)	M	M
MIMO CSI RX (ALAMOUTI-BASED 2x2)	N	MN
MIMO CSI RX & TX (TX/RX BEAMFORMING MxN)	$\le MN$	MN

The capacity of the wireline SISO channel (1.1) is given by [2]

$$\begin{aligned} C_{SISO} &= \log_2\left(1+\frac{E|x|^2}{E|n|^2}\right) \\ &= \log_2\left(1+\frac{P}{2\sigma^2}\right) \end{aligned} \quad (1.50)$$

where P is the transmission power

$$P = E|x|^2 . \quad (1.51)$$

As we can see, for the wireline SISO channel, the capacity can be increased only if the transmission power is increased. This is not the case for wireless MIMO channels as shown next. The capacity of the wireless MIMO channel (1.39) is given by [1]

$$C_{MIMO} = \log_2 \det\left(I + \frac{P}{2\sigma^2}\frac{1}{M}Q \right) \quad (1.52)$$

where

$$P = \sum_{j=1}^{M} E\left|x_j\right|^2 \quad (1.53)$$

is the total transmission power radiating from the transmit antennas

$$Q = \begin{cases} HH^* & \text{if } N < M \\ H^*H & \text{if } N \geq M \end{cases} \quad (1.54)$$

and I is the identity matrix of size

$$\left(\min\{M,N\} \times \min\{M,N\}\right) \quad (1.55)$$

"$\det$" denotes the determinant of the matrix. Averaging (1.52) with respect to the Rayleigh distribution of the fading yields [1]

$$C_{MIMO} = \min\{M,N\}\log_2\left(1 + \frac{P}{2\sigma^2}\right) \quad (1.56)$$

(It should be pointed out that the above expression is an approximation; however. for mid-to-high values of P it is a fairly accurate one). We define the *multiplexing gain* as

$$\text{multiplexing gain} = \frac{C_{MIMO}}{C_{SISO}}. \quad (1.57)$$

Under the same transmission power P, the multiplexing gain is

$$\text{multiplexing gain} = \min\{M,N\}. \quad (1.58)$$

Thus, by using multiple antennas we can dramatically increase the throughput. If our main goal is power saving, and not increased throughput, we can use the MIMO architecture and have the same throughput as for the SISO channel, but with a much reduced transmission power. Usually, MIMO systems are used to simultaneously achieve both increased throughput and reduced power. In that case, we achieve multiplexing gain that is smaller than $\min\{M,N\}$.

MIMO SYSTEMS, OFDM, AND LDPC CODES

In this section, we briefly discuss the benefits of using MIMO architecture together with OFDM modulation and LDPC coding.

Wireless channels, in addition to the problem of fading, also suffer from the problem of *inter-symbol-interference (ISI)*. ISI is caused by the reception of a small number of reflections from remote objects (as opposed to the large number of reflections from nearby objects that causes fading). ISI causes the receiver to receive the original signal, overlapped by some delayed versions of the signal (illustrated in Figure 5). Traditionally, different types of *equalizers* were used to reject ISI. However, the complexity of good equalizers is usually high, making their employment fairly problematic. OFDM modulation [4] provides a fairly strong and simple ISI rejection mechanism. It essentially introduces a *guard interval* in between symbols, in which the interference can reside without critically distorting the original signal. The use of OFDM modulation within MIMO structured systems creates a strong system that has the ability to successfully reject fading as well as ISI.

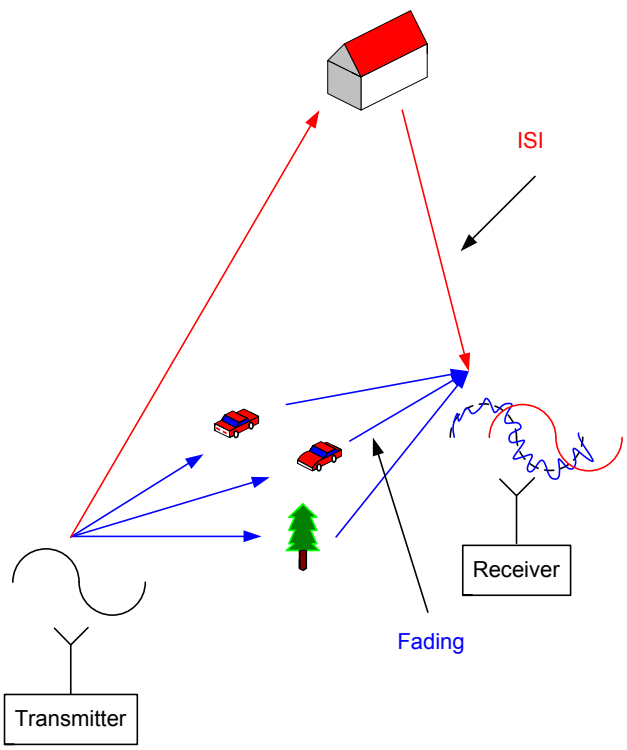


Figure 5: Wireless channel: ISI and fading problems

The increased capacity of MIMO channels can be translated into increased throughput provided that proper coding is used prior to transmission. The coding procedure essentially pads the transmitted data with some

protection bits that help the receiver decide whether errors occurred during transmission. LDPC codes are highly efficient (with respect to encoding and decoding complexity) capacity-approaching codes. Using LDPC codes helps to fulfill the high-throughput potential of MIMO systems in a highly efficient manner.

The upcoming IEEE 802.11n Wireless Local Area Networks (WLAN) standard uses MIMO architecture, along with OFDM and LDPC coding. The standard has the ability to provide a stunning throughput of up to 600 Mbps as apposed to 54 Mbps provided by the older, non-MIMO IEEE 802.11a standard.

CONCLUSION

Wireless channels key problem is fading. The fading induces rapid changes in the SNR and in turn critically damages reliability. The ability to transmit and receive through multiple antennas enables us, while applying various spatial diversity techniques, to reject fading and ultimately have substantially improved reliability. Using multiple antennas does not only reject fading; better yet, it actually harnesses the fading itself in favor of increased capacity. The increased capacity, under proper coding, eventually translates into increased throughput. Employing multiple antennas also allows for power saving at the expense of reduced throughput or reduced reliability. In any case, the reduced throughput and reliability are well above those of the original single antenna channel. The joint use of MIMO architecture, OFDM modulation, and LDPC coding, creates a highly resilient communication system, which successfully rejects fading and ISI and fulfills the high-throughput potential offered by MIMO systems.

One of the key ingredients for current and future versions of Intel Centrino mobile technology is the ability to provide mobile users with fast and low-power communications. Intel achieves that goal by incorporating MIMO architecture into its line of wireless communication products.

ACKNOWLEDGMENTS

We wish to thank all the reviewers for their useful comments.

REFERENCES

[1] E. Teletar, "Capacity of multi-antenna Gaussian channels," *AT&T-Bell Labs, Tech. Rep*., June 1995.

[2] A. J. Viterbi and J.K. Omura, *Principles of Digital Communication and Coding*, McGraw-Hill, Inc., New York, USA, 1979, pp. 54-64.

[3] S.M. Alamouti, "A Simple Transmit Diversity Technique for Wireless Communications," *IEEE Journal on Selected Areas in Communications*, vol. 16, October 1998, pp. 1451-1458.

[4] M. Jankiraman, *Space-Time Codes and MIMO Systems*, Artech House, London, UK, 2004, pp. 98-100, and 165-191.

AUTHOR'S BIOGRAPHY

Ilan Hen is a lead algorithmic engineer with Intel's Mobile Wireless Group. He received his B.Sc. and M.Sc. degrees in Electrical Engineering from the Technion, Israel Institute of Technology, in 1998 and 2002, respectively. He is currently completing his studies for his Ph.D degree in Electrical Engineering. His research interests include information theory, communications, signal processing, and chaos theory. His e-mail is ilan.hen at intel.com.

Copyright © Intel Corporation 2006. All rights reserved.
Intel and Centrino are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

This publication was downloaded from http://developer.intel.com/.

Legal notices at http://www.intel.com/sites/corporate/tradmarx.htm.

THIS PAGE INTENTIONALLY LEFT BLANK

For further information visit:

developer.intel.com/technology/itj/index.htm

Copyright © 2006 Intel Corporation. All rights reserved.
Intel is a trademark or registered trademark of Intel Corporation or its subsidiaries in the United States and other countries.
For a complete listing of trademark information visit: www.intel.com/sites/corporate/tradmarx.htm