CONFIGURING AND TUNING FOR PERFORMANCE ON INTEL® 5100
MEMORY CONTROLLER HUB CHIPSET BASED PLATFORMS

Contributor

Perry Taylor
Intel Corporation

Index Words

Intel® 5100 Memory Controller Hub chipset
Intel® 5100 MCH
performance
tuning
memory bandwidth
single channel
dual channel
dual-rank
single-rank
single socket
dual socket
quad-core
dual-core
DDR2-533
DDR2-667
I/O device placement
platform tuning
thermal throttle
electrical throttle
global throttle
processor prefetching
hardware prefetch
second sector prefetch
adjacent sector prefetch
BIOS settings

Abstract

A system architect must make platform configuration choices based on multiple architecture tradeoffs between cost, power, and performance. Understanding the end impact on performance that these configuration decisions have is critical in designing competitive solutions around the Intel® 5100 Memory Controller Hub chipset (Intel® 5100 MCH chipset).

This article presents performance-related architecture topics for the Intel 5100 MCH chipset to assist system architects in designing a high performance solution. It will help engineers and architects make decisions with an awareness of performance implications, such as CPU population, memory configuration, and I/O device placement. This article also addresses performance tuning options for the Intel 5100 MCH chipset that can help increase performance for specific usage models.

Introduction

Intel® 5100 Memory Controller Hub chipset (Intel® 5100 MCH chipset) solutions provide board designers with a “blank slate,” allowing flexible design and layout of memory, front side bus, generation 1 PCI Express® capability and legacy I/O. While this design flexibility is desirable, it also allows for less than optimal performance configurations. Likewise, even with a board laid out for optimal performance, designers or end users can cripple system performance by populating the platform with lower performing processors, low throughput memory configurations, and poorly placed I/O endpoints. Not only the hardware selection but also hardware placement is important to system performance. Memory placement and I/O placement can have significant impact on performance as we will later demonstrate.

Once all hardware choices have been made, some additional performance may be possible with platform tuning for specific usage models.

This article attempts to help with these three areas related to performance:

- Intel® 5100 MCH configuration
- Hardware configuration
- Platform tuning

The information provided is intended to help designers and end users make performance aware decisions in regards to these three areas, allowing them to balance cost, power and thermals with performance needs on the Intel 5100 MCH chipset.
Performance Test Configuration

Performance data presented in this article is collected on the following system configuration:

- Williamsport Customer Reference Board (revision B)
- Dual Socket Intel® Xeon® processor E5410 2.33GHz (Quad-Core)
- Dual Socket Intel® Xeon® processor E5220 2.33GHz (Dual-Core)
- 2 memory channels, 2 DIMMs per channel, 4GB of system memory
- 4x1GB DDR2-667, dual rank, CL5
- BIOS version WSPTG015
- 32 Bit Linux* (Cent OS 4.4)

Intel® 5100 Memory Controller Hub Chipset Feature and Technology Overview

The Intel 5100 Memory Controller Hub (Intel 5100 MCH) chipset is a low power memory controller hub designed specifically for embedded applications. The Intel 5100 MCH chipset is derived from the Intel® 5000P Memory Controller Hub chipset (Intel® 5000P MCH chipset), a high performance server class chipset and as such, the Intel 5100 MCH chipset has many of the same features, technologies, and performance capabilities as the Intel 5000 Memory Controller Hub chipset (Intel® 5000 MCH chipset). The primary difference with the Intel 5100 MCH chipset as compared to the Intel® 5000 MCH chipset is the removal of the fully buffered DIMMs (FBDs) that were replaced with a native DDR2 controller. This architecture change reduces the total cost of platform ownership by reducing the overall platform power consumption while still delivering high performance.

Figure 1 details the platform block diagram.

Like the Intel 5000 MCH chipset, the Intel 5100 MCH chipset is designed with a dual independent front side bus (FSB) for improved bandwidth and efficiency over previous generations, supporting 667, 1066, and 1333 MT/s. There are six x4 (pronounced “by four”) Generation 1 PCI Express GB links available for direct connect I/O and a x4 direct media interface (DMI) link available to interface with an I/O Controller Hub (ICH). The six x4 PCI Express links can be combined to form various combinations of x8 links and/or x16.

The Intel 5100 MCH chipset is designed with a native DDR2 memory controller supporting registered ECC DDR2533 and DDR2-667. Dual independent memory channels provide improved bandwidth and efficiency supporting up to 3 DIMM modules per channel. Intel 5100 MCH chipset supports single rank, dual rank, and quad rank DIMMs up to a maximum of 6 ranks per channel and a total capacity of 48 GB.

“Platform hardware and software tuning can help extract the last bit of performance out of a system, but the hardware configuration ultimately determines the performance potential (or pitfalls) of a platform.”

**Performance-Related Architecture Considerations**

The most important and often overlooked factor attributing to system performance is the hardware configuration. Platform hardware and software tuning can help extract the last bit of performance out of a system, but the hardware configuration ultimately determines the performance potential (or pitfalls) of a platform.

System architects may unintentionally end up designing a lower performing solution by choosing a less than desirable configuration in an attempt to save cost, power, thermals or design time. These sections address architecture considerations to help system architects and end users understand the performance impact of various tradeoffs when selecting CPU, memory, and I/O configurations.

**CPU**

The best processor configuration for computation performance will be the CPU with the maximum possible CPU frequency, cores/threads, sockets, and FSB speed. However this configuration does not always fit into the end budget, thermal, and power constraints. In this section we will explore performance impacts of choosing FSB frequency, dual-core versus quad-core; single socket versus dual socket, and finally a single socket quad-core compared to a dual socket dual-core on the Intel 5100 MCH chipset architecture.

**FSB Frequency**

As explained earlier, the Intel 5100 MCH chipset supports FSB of 667, 1067, and 1333 MT/s. For the best performance, processors supporting FSB of 1333 MT/s should be used. Note that higher FSB frequency not only results in higher effective FSB bandwidth and lower latency to memory but also makes the Intel 5100 MCH chipset core frequency proportionally faster. We will review FSB and memory frequency performance data later in the article.

**Quad-Core versus Dual-Core**

Here we illustrate the performance benefit of the quad-core architecture versus dual-core with the benchmark SPEC CPU2006. SPEC CPU2006 is a suite of tests many of which are CPU compute-bound while some are CPU to memory bandwidth-bound. For complete information on SPEC CPU2006 refer to: [http://www.spec.org](http://www.spec.org) [3]

The performance gain on quad-core will vary per sub-test and depends on the limiting factors mentioned above. Sub-tests limited by CPU-memory path bandwidth see little to no benefit from quad-core while those limited by CPU computation scale almost perfectly. The end score is an averaging of these sub-tests.

Lab testing shows that the quad-core platform configuration improves the SPECfp_rate_base2006(est.) score by an average of 40 percent and the SPECint_rate_base2006(est.) score by an average of 57 percent over the dual-core platform configuration. As expected, some computation-heavy sub-tests show nearly perfect scaling from dual-core to quad-core while memory intensive sub-tests show no scaling from dual-core to quad-core.
Single Socket versus Dual Socket
Now we show the performance benefit of a single socket architecture with a quad core CPU versus dual socket architecture with a quad-core CPU again using the benchmark SPEC CPU2006. This configuration compares four cores on one socket against eight cores across two sockets. Again, the scaling results come down to CPU computation power and memory bandwidth. Our computation power is doubling so we can expect about the same scaling here as with dual-core versus quad-core, but we also expect higher bandwidth with the additional FSB.

Tests show a 48 percent benefit on SPECfp_rate_base2006(est.) and 59 percent for SPECint_rate_base2006(est.) for this configuration.

Dual-Core and Dual Socket versus Quad-Core and Single Socket
Here we examine the performance difference between platforms configured with two dual-core processors versus a single socket populated with a quad-core. This is an interesting comparison since we are comparing an equal number of processor cores, but varying the number of physical processors and FSBs used.

Generally, the dual-core dual socket configuration yields higher performance but also increases system cost and power consumption. At first glance, the performance seems very similar with a 6 percent increase for SPECfp_rate_base2006(est.) and SPECint_rate_base2006(est.) is within test noise of 3 percent.

To better understand this we need to look at the sub-tests of SPEC CPU2006 Floating Point. Many of the tests perform the same on the two configurations. These tests are CPU-bound and performance is dependent on the processor core count and frequency. Some of the tests perform significantly better on the dual socket configuration. These tests are not CPU-bound, but rather memory-bound and dual socket configuration gives higher CPU-memory throughput. Figure 2 shows the tests and percentage of increase for those tests sensitive to CPU-memory throughput such as: 410.bwaves, 433.milc, 437.leslie3d, 450.soplex, 459.GemsFDTD, 470.lbm, 481.wlf, and 482.sphinx3. The remaining floating point tests are those that are computation-bound.

![Figure 2: CPU2006fp performance, single socket quad-core versus dual socket dual-core. Source: Intel Corporation, 2009](image-url)
Based on SPEC CPU2006, we see that the computation performance of dual core with dual socket design when compared to a quad core single socket design is similar, but memory throughput improves on the dual socket design compared to single socket, showing a 27 percent improvement using 2 sockets on 437.leslie3d(est.).

**Memory**

The memory subsystem is a vital component to platform performance and often becomes the limiting factor of benchmark throughput. Therefore it is critical to populate memory with end performance in mind. This section explores the performance impact of populating one versus two channels, one DIMM per channel, two DIMMs per channel, three DIMMs per channel, dual rank DIMMs, single rank DIMMs, and memory frequency.

**Channel Population**

The Intel 5100 MCH chipset features two independent DDR2 channels with each channel having its own independent memory controller. In order to get the most benefit out of the memory system, it is vital to populate both channels. Memory configurations with two or more DIMMs should divide DIMMs equally between the channels.

Figure 3 illustrates the performance delta when two DIMMs are placed in one channel versus divided between channels. Memory bandwidth tests results are from an Intel internal benchmark that behaves much like Stream Benchmark, with higher memory efficiency. A 92 percent increase is observed from one channel to two channels with 1-GB, dual rank, DDR2-667 modules when CPU is issuing 66 percent read 33 percent write requests. From this data it is clear that utilizing both memory channels of the MCH is vital for memory performance. Based on the data in Figure 3, populating both memory channels is highly recommended.

**DIMMs per Channel**

Each memory channel on the Intel 5100 MCH chipset supports up to three DDR2 DIMMs. It is important to understand the performance impact of using one, two, or three DIMMs per channel to design a cost-effective product with high performance.

The estimated performance gains from one to two to three DIMM configurations with 1-GB, dual rank, 667 DDR2 modules is a 4.5 percent improvement from one DIMM to two DIMMs. Adding a third DIMM per channel does not typically increase bandwidth potential unless a benchmark is memory capacity limited. Populating three DIMMs per channel may potentially yield higher application/benchmark performance for capacity limited usage, but actually memory bandwidth will not increase. These data points indicate that utilizing both memory channels is much more important than populating multiple DIMMs on just one channel. Usage models requiring high memory throughput should populate two DIMMs per channel to gain the additional bandwidth while models with strict power and cost limits may consider using only one DIMM per channel. One must also consider the target software applications that will be executing on the platform. If maximum memory capacity is required then all three DIMMs per channel should be populated to achieve 4848 GB of total system memory.
Dual Rank versus Single Rank

Dual rank DIMMs are recommended over single rank for performance and to enable full rank interleaving of 4:1. Figure 4 shows the measured benefit of dual rank with 66 percent read 33 percent write traffic on various memory configurations. For the maximum configuration the dual rank memory provides an additional 6.5 percent throughput. Note: Three DIMMs per channel provides additional capacity, but not bandwidth (not shown in Figure 4).

DDR2-533 versus DDR2-667

When selecting memory frequency, the FSB frequency must also be considered due to the impact of memory gearing. Memory gearing refers to the frequency ratio between the front side bus and memory interface. When the ratio is not 1:1, additional memory latency occurs. Table 1 shows possible frequency ratios related to memory gearing.

<table>
<thead>
<tr>
<th>FSB Frequency</th>
<th>DDR2 Frequency</th>
<th>Memory Gearing via I/O Controller Hub (ICH)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1067 MT/s</td>
<td>533 MHz</td>
<td>1:1 Ratio</td>
</tr>
<tr>
<td>1067 MT/s</td>
<td>667 MHz</td>
<td>Not a 1:1 Ratio</td>
</tr>
<tr>
<td>1333 MT/s</td>
<td>533 MHz</td>
<td>Not a 1:1 Ratio</td>
</tr>
<tr>
<td>1333 MT/s</td>
<td>667 MHz</td>
<td>1:1 Ratio</td>
</tr>
</tbody>
</table>

Table 1: Memory gearing table

Testing shows that the reduced memory latency of 667 MHz over 533 MHz makes up for the negative impact of memory gearing. So a simple rule of thumb is that higher memory frequency provides higher performance. Figure 5 reports the relative CPU-memory bandwidth recorded with each combination in dual socket configuration. Note that there is no gain from a 1067/533 configuration to 1067/667, but we show an estimated 22 percent bandwidth improvement with the 1333/667 configuration.

PCI Express*

Recall from the block diagram in Figure 1 that the PCI Express (PCIe) ports of the Intel 5100 MCH chipset can be configured in many ways. There are six x4 PCIe lanes available directly from the MCH. These can be configured as: a single x16 link with a 1x8 link, three x8 links, or six x4 links. There are also six x1 links available from the I/O controller hub (ICH), which can be combined for form one x4 and two x1 links. This section addresses performance related to these choices.

Lane Width

PCI Express lane width should be chosen based on required bandwidth of the I/O device. Peak PCI Express bandwidth efficiency is about 81 percent for reads and 86 percent for writes on the Intel 5100 MCH chipset.

Another important concept with link width is transaction latency. Transmit time of PCI Express packets increases as the link width decreases. For this reason, performance can benefit from link width even when higher bandwidth is not demanded.
For best performance, place I/O devices as close to the Intel 5100 MCH chipset as possible. Hence the use of direct MCH attached PCIe interfaces is recommended for performance sensitive IO devices. On the other hand, having relatively higher latency to memory, the PCIe slots on the I/O Controller Hub are recommended for less performance-sensitive applications. Note that increased latency also impacts throughput; how much will depend on how many outstanding transactions are pending. Tables 2 and 3 show the relative latency and bandwidth measured on a x4 PCI Express link via Intel 5100 MCH chipset and I/O Controller Hub. Note: The bandwidth measurement are carried out with up to 32 outstanding requests, hiding much of the latency impact on bandwidth while latency measurements are carried using one outstanding transaction at a time. Figure 6 illustrates recommended I/O device placement for best performance.

A another important concept in I/O device placement is I/O Unit (IOU) balance within the MCH. As shown in Figure 7, the Intel 5100 MCH chipset’s available PCI Express slots are divided between IOU0 and IOU1. IOU0 contains the x4 DMI link and 2x4 or 1x8 PCI Express links. IOU1 contains 4x4 or 2x8 PCI Express links. Populate PCI Express slots with the intent of balance loading traffic between the IOUs. Figure 7 also illustrates an example ordering preference to ensure IOU balance.

### Out of the Box Performance

Prior to product release, Intel conducts performance testing and analysis to determine ideal default chipset and processor settings. This work results in the best “out of the box” performance for general usage models and in most cases additional tuning is not necessary.

Additional performance might be achieved depending on the CPU and memory configuration used, or the specific usage model of interest. Some of these additional tuning options are presented in the following section.
Accessing the Intel® 5100 Memory Controller Hub Chipset Control Registers

The following sections about hardware tuning require an understanding of chipset control registers. Inside the Intel 5100 MCH chipset there are registers related to status, capabilities and control. These registers are defined in the datasheet [2]. Changing values within these registers can control specific behaviors of the Intel 5100 MCH chipset. Registers are mapped to the PCI configuration space and they can be accessed with the correct physical address. The addressing nomenclature of the PCI configuration space is: bus, device, function, offset, and bits.

Linux provides the commands “lspci” and “setpci” for listing and changing the PCI configuration space. For example, let us say we want to disable FSB1. According to Table 4 we need to set bit 30 for bus 0, device 16, function 0 at offset 78h.

<table>
<thead>
<tr>
<th>Register</th>
<th>Field</th>
<th>Bus</th>
<th>Device</th>
<th>Function</th>
<th>Offset</th>
<th>Bit</th>
<th>Value</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>FSB1</td>
<td>[1]</td>
<td>0</td>
<td>16</td>
<td>0</td>
<td>78h</td>
<td>30</td>
<td>1</td>
<td>FSB1 Disabled</td>
</tr>
</tbody>
</table>

Table 4: Disable FSB 1 example. Source: Intel Corporation, 2009

lspci –s 0:10.0 –xxx dumps config space for bus 0, device 10h, function 0. From this output we can locate the register at offset 78h (highlighted below in bold text).

[root]# lspci -s 0:10.0 -xxx

00:10.0 Host bridge: Intel Corporation: Unknown device 65f0 (rev 90)
00:  86 80 f0 65 00 00 00 00 90 00 00 06 00 00 80 00
10:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
20:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
30:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
40:  00 00 ff 07 00 08 00 00 00 00 00 00 fe 00 00 00 00
50:  00 00 02 00 00 00 04 04 00 10 11 01 00 00 31 33
60:  00 12 08 01 00 00 00 00 ff ff ff ff 00 00 00 00
70:  8c  c0  c2  0f 00 00 00 00 8c  c0  c2  0f 00 00 00 00
80:  01 00 80 00 04 02 80 00 00 00 00 00 00 00 00 00
90:  02 01 80 00 08 03 80 00 00 00 00 00 00 00 00 00
a0:  00 00 00 00 00 00 00 00 00 00 00 04 80 00 40 06 80 00
b0:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 20 05 80 00
c0:  00 00 00 00 5a 5a 5a 5a 5a 5a 5a 5a 5a 00 00 00 00
d0:  00 00 00 00 00 00 00 00 00 00 0c 2c 00 00 03 01 00 00
e0:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
f0:  dc  7f 40 00 06 81 58 81 f4 08 d8 03 80 00 00 00

“Inside the Intel 5100 MCH chipset there are registers related to status, capabilities and control.”
“Remember that this is little endian, requiring us to flip it backwards per byte.”

Remember that this is little endian, requiring us to flip it backwards per byte, resulting in: 0F E2 C0 8C. For completeness, this is broken down to bit level with bit 30 highlighted:

Default Setting: 0000 1111 1110 0010 1100 0000 1000 1100
New Setting: 0100 1111 1110 0010 1100 0000 1000 1100

Working backwards, the result is 4F E2 C0 8C, showing that we need to set offset 7Bh = 4Fh as follows:

setpci -s 0:10.0 7b=4f

Hardware Tuning Recommendations

There are several hardware level settings that can be changed through BIOS menus or chipset control registers that can help improve performance. Here we discuss a few of these options for the Intel 5100 Memory Controller Hub.

Intel® Processor Prefetching

Intel® Core™ microarchitecture has two hardware level prefetch mechanisms to help reduce CPU memory read latency. These prefetchers bring data into processor L2 cache before the processor requires it in an attempt to produce a cache hit rather than a miss, resulting in increased performance.

There are two prefetchers available within the CPU architecture and can be set within BIOS, the hardware prefetch, also known as Data Prefetch Logic or DPL and the L2 Streaming Prefetch (L2S), also known as Adjacent Sector Prefetch.

Tuning processor prefetching is a topic within itself and is beyond the scope of this article. Please refer to the following paper for further information on processor prefetching:


Hardware Prefetch

The default recommendation is to enable hardware prefetch. However, depending on individual usage models, hardware prefetch may also fetch more cache lines than are needed by an application. This can result in increased memory bus utilization and may affect performance under usage models that require heavy memory bandwidth. It is left to the user to choose hardware prefetch settings that best suit the application under consideration. It is beneficial to test with HW Prefetching ON and also OFF in order to determine the optimal performance for the specific usage model.

L2 Streaming Prefetch

L2S improves performance under some usage models with sequential memory addressing and/or spatial locality. L2S may be enabled/disabled via BIOS setting. Again, the effect of L2S on the performance is application-specific.
FSB Tuning for Single Socket Configurations
In single socket configurations, the second, unused socket should not be populated with a processor and BIOS should disable the unused FSB. The motivation is to improve performance by allowing the bus to switch to in-order mode rather than deferred mode; this reduces transaction latency and protocol overhead.

Table 4 above shows the bit in the FSBC[1] register that can be checked to verify that BIOS is disabling the second FSB in single socket configurations. For complete register definition, please see the Intel 5100 MCH Chipset Datasheet [2].

Memory Tuning
Memory timing settings are set to optimum values by default, providing the best performance possible within specification. DRAM timing registers should not be modified by end users except through preset BIOS settings.

For recommended memory settings, the following parameters should be selected in BIOS:

• MCH Channel Mode: Channel Interleave
• Channel Dependent Sparing: Disabled
• Channel Specific Sparing: Disabled
• Rank Interleave = 4:1
• Channel 0: Enabled
• Channel 1: Enabled
• DIMM Calibration Reuse: Enabled
• Read Completion Coalesce: Auto

PCI Express® Tuning with Maximum Payload Size
The Intel 5100 MCH chipset supports a PCI Express Maximum Payload Size (MPS) of 128 bytes and 256 bytes. The default and recommended setting is 128. A 128 byte payload size allows opportunistic split completion combining (coalescing) for read requests, a feature not supported with a 256-byte MPS.

Under specific I/O usage models that perform high percentages of inbound writes with large payloads and few inbound reads, it will benefit performance to disable coalescing and increase MPS to 256 bytes. This allows I/O devices to send up to 256 bytes of data per write packet, improving write throughput but limits the maximum read completion size to 64 bytes, reducing read throughput potential.

Table 5 defines the register changes required in the Intel 5100 MCH chipset to implement a 256-byte MPS tweak. Note that end devices must also be changed to 256-byte MPS.
Throttling Mechanisms

During performance benchmarking it is sometimes useful to disable the throttling technologies of the Intel 5100 MCH chipset to verify that performance limits are not throttle related.

Thermal Throttle

Thermal throttle allows for dynamic frequency scaling based on defined thermal thresholds, but the Intel 5100 MCH chipset does not implement thermal throttling.

Other platform components may support thermal throttle, such as the CPU. Please refer to component-specific documentation on how to disable thermal throttle.

Global Throttle

Global throttle allows for software controlled throttling on memory activations for a long time window, as shown in Table 6. If the number of activations to a memory rank exceeds the specified limit, then further requests are blocked for the remainder of the activation window. Setting the following bit values to zero will disable global throttle features.

Electrical Throttle

Electrical throttling is a mechanism that limits the number of activations within a short time interval that would otherwise cause silent data corruption on the DIMMs. Disable electrical throttle with the configuration settings listed in Table 7.

---

**Throttling Mechanisms**

During performance benchmarking it is sometimes useful to disable the throttling technologies of the Intel 5100 MCH chipset to verify that performance limits are not throttle related.

**Thermal Throttle**

Thermal throttle allows for dynamic frequency scaling based on defined thermal thresholds, but the Intel 5100 MCH chipset does not implement thermal throttling.

Other platform components may support thermal throttle, such as the CPU. Please refer to component-specific documentation on how to disable thermal throttle.

**Global Throttle**

Global throttle allows for software controlled throttling on memory activations for a long time window, as shown in Table 6. If the number of activations to a memory rank exceeds the specified limit, then further requests are blocked for the remainder of the activation window. Setting the following bit values to zero will disable global throttle features.

**Electrical Throttle**

Electrical throttling is a mechanism that limits the number of activations within a short time interval that would otherwise cause silent data corruption on the DIMMs. Disable electrical throttle with the configuration settings listed in Table 7.

---

**Table 5:** Enable 256-byte maximum payload size. Source: Intel Corporation, 2009

<table>
<thead>
<tr>
<th>Register</th>
<th>Field</th>
<th>Bus</th>
<th>Device</th>
<th>Function</th>
<th>Offset</th>
<th>Bit(s)</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>PEXCTRL[7:2,0]</td>
<td>COALESCE_EN</td>
<td>0</td>
<td>7-2,0</td>
<td>0</td>
<td>48h</td>
<td>10</td>
<td>0</td>
</tr>
<tr>
<td>PEXDEVCTRL[7:2,0]</td>
<td>MPS</td>
<td>0</td>
<td>7-2,0</td>
<td>0</td>
<td>74h</td>
<td>5-7</td>
<td>001b</td>
</tr>
</tbody>
</table>

**Table 6:** Disable global throttle. Source: Intel Corporation, 2009

<table>
<thead>
<tr>
<th>Register</th>
<th>Field</th>
<th>Bus</th>
<th>Device</th>
<th>Function</th>
<th>Offset</th>
<th>Bit(s)</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>THRTHIGH</td>
<td>THRTHIGHLM</td>
<td>0</td>
<td>16</td>
<td>1</td>
<td>65h</td>
<td>7-0</td>
<td>0</td>
</tr>
<tr>
<td>THRLOW</td>
<td>THRLOWLM</td>
<td>0</td>
<td>16</td>
<td>1</td>
<td>64h</td>
<td>7-0</td>
<td>0</td>
</tr>
<tr>
<td>GBLACT</td>
<td>GBLACTLM</td>
<td>0</td>
<td>16</td>
<td>1</td>
<td>60h</td>
<td>7-0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Table 7:** Disable electrical throttle. Source: Intel Corporation, 2009

<table>
<thead>
<tr>
<th>Register</th>
<th>Field</th>
<th>Bus</th>
<th>Device</th>
<th>Function</th>
<th>Offset</th>
<th>Bit(s)</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>MTR[1:0][3:0]</td>
<td>ETHROTTLE0</td>
<td>0</td>
<td>22</td>
<td>0</td>
<td>15Ah</td>
<td>9</td>
<td>0</td>
</tr>
<tr>
<td>MTR[1:0][3:0]</td>
<td>ETHROTTLE0</td>
<td>0</td>
<td>22</td>
<td>0</td>
<td>158h</td>
<td>9</td>
<td>0</td>
</tr>
<tr>
<td>MTR[1:0][3:0]</td>
<td>ETHROTTLE0</td>
<td>0</td>
<td>22</td>
<td>0</td>
<td>156h</td>
<td>9</td>
<td>0</td>
</tr>
<tr>
<td>MTR[1:0][3:0]</td>
<td>ETHROTTLE0</td>
<td>0</td>
<td>22</td>
<td>0</td>
<td>154h</td>
<td>9</td>
<td>0</td>
</tr>
<tr>
<td>MTR[1:0][3:0]</td>
<td>ETHROTTLE0</td>
<td>0</td>
<td>21</td>
<td>0</td>
<td>15Ah</td>
<td>9</td>
<td>0</td>
</tr>
<tr>
<td>MTR[1:0][3:0]</td>
<td>ETHROTTLE0</td>
<td>0</td>
<td>21</td>
<td>0</td>
<td>158h</td>
<td>9</td>
<td>0</td>
</tr>
<tr>
<td>MTR[1:0][3:0]</td>
<td>ETHROTTLE0</td>
<td>0</td>
<td>21</td>
<td>0</td>
<td>156h</td>
<td>9</td>
<td>0</td>
</tr>
<tr>
<td>MTR[1:0][3:0]</td>
<td>ETHROTTLE0</td>
<td>0</td>
<td>21</td>
<td>0</td>
<td>154h</td>
<td>9</td>
<td>0</td>
</tr>
</tbody>
</table>
Conclusion

When designing a solution with the Intel 5100 MCH chipset the designer should make architecture decisions based on CPU population, FSB frequency, memory population, memory frequency, I/O device selection, and placement with an understanding of how it will impact end performance. All areas of architecture should be carefully thought out with the end solution in mind to balance power, performance, cost and thermals.

Disclaimers

Performance tests and ratings are measured using specific computer systems and/or components and reflect the approximate performance of Intel products as measured by those tests. Any difference in system hardware or software design or configuration may affect actual performance. Buyers should consult other sources of information to evaluate the performance of systems or components they are considering purchasing. For more information on performance tests and on the performance of Intel products, visit Intel Performance Benchmark Limitations at:

Intel, and the Intel logo, Intel Core, and Xeon are trademarks of Intel Corporation in the U.S. and other countries.

Intel processor model numbers are not a measure of performance. Processor numbers differentiate features within each processor family, not across different processor families.

SPECint*_rate_base2006 and SPECfp*_rate_base2006 are capacity-based metrics used to measure throughput of a computer that is performing a number of tasks. This is achieved by running multiple copies of each benchmark simultaneously with the number of copies set to set to the number of logical hardware cores seen by the operating system. SPEC* CPU2006 provides a comparative measure of compute intensive performance across the widest practical range of hardware. The product consists of source code benchmarks that are developed from real user applications. These benchmarks depend on the processor, memory and compiler on the tested system. SPEC, SPECint, SPECfp, SPECrate are trademarks of the Standard Performance Evaluation Corporation. For more information go to: www.spec.org/spec/trademarks.html. [1]

All SPEC CPU2006 data in this document is estimated based on measurements of Intel internal reference platforms; the data is being provided as described in the CPU2006 Run Rules 4.5 Research and Academic usage of CPU2006.

Copyright © 2009 Intel Corporation. All rights reserved. *Other names and brands may be claimed as the property of others.
References


Author Biography

Perry Taylor: Perry Taylor is a senior performance engineer in the Embedded and Communications Group at Intel Corporation. Perry has been at Intel for 8 years and in that time has focused on performance analysis and tuning of Intel® architecture and has received multiple divisional awards for his contributions.

Copyright

Copyright © 2009 Intel Corporation. All rights reserved.
Intel, the Intel logo, and Intel Atom are trademarks of Intel Corporation in the U.S. and other countries.
*Other names and brands may be claimed as the property of others.