Part I - Shared Memory

1. Overview

1.1 Introduction

This document provides an insight look of architecture and implementation changes based on SiS 5511/2/3 PCI/ISA chipset. The additional pins and registers of 5511 for the shared memory are described in this part.

The shared memory architecture is achieved by allowing both GUI / VGA, and System DRAM controller to control system memory. For the shared memory application, the chipset always acts as the arbiter of memory bus masters . Whenever the GUI wants to access the memory bus, it requests the memory bus to the chipset first. The chipset grants the memory bus to the GUI, only if the memory bus is not needed by the chipset. Up to four pins are used between the chipset and the GUI, CPUCLK , VGAREQ#, VGAGNT#, and PREQ. Both asynchronous and synchronous operation modes are supported. The chipset also supports the two priority scheme. Other important key features such as direct access frame buffer and memory access latency are also supported. The system block diagram is shown in Figure 1.1.

1.2 Functional Block Diagram

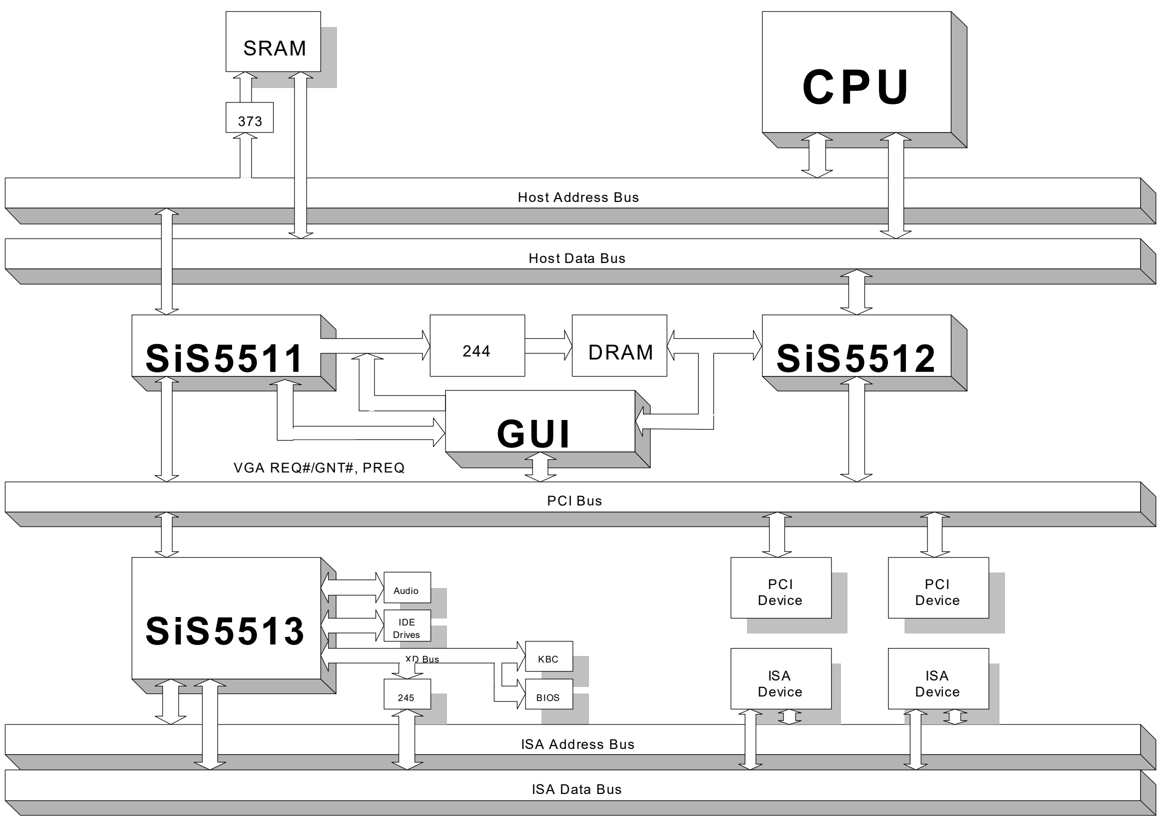


Figure 1.1 Shared Memory Architecture Block Diagram

2. Functional Description

2.1 System DRAM Controller

2.1.1 DRAM Arbiter

The system memory is default owned by the chipset. The chipset is responsible for regular memory access requests as is in the non-shared memory architecture. These regular memory access requests are from CPU, PCI devices and refresh. For the shared memory application, DRAM arbiter arbitrates graphics controller memory access request in additional to regular requests. DRAM arbiter receives bus requests and services in the following order: graphics controller's request, CPU, refresh, PCI master. The graphics controller memory bus request will not be granted while CPU, PCI master, or ISA master is accessing the memory. The maximum latency before graphics controller owning the memory bus is 30 CPUCLK for the high priority request and is 60 CPUCLK for the low priority request.

When graphics controller needs to access the system memory, it uses a pair of active low signals called VGAREQ# and VGAGNT# to arbitrate for the system memory. The arbitration is done by asserting the VGAREQ# active. After the chipset completes any outstanding memory cycles, it releases the memory bus to the graphic controller by asserting the VGAGNT# active. The basic handshaking protocol of VGAREQ# and VGAGNT# is shown in Figure 2.1, Figure 2.2 and Figure 2.3.

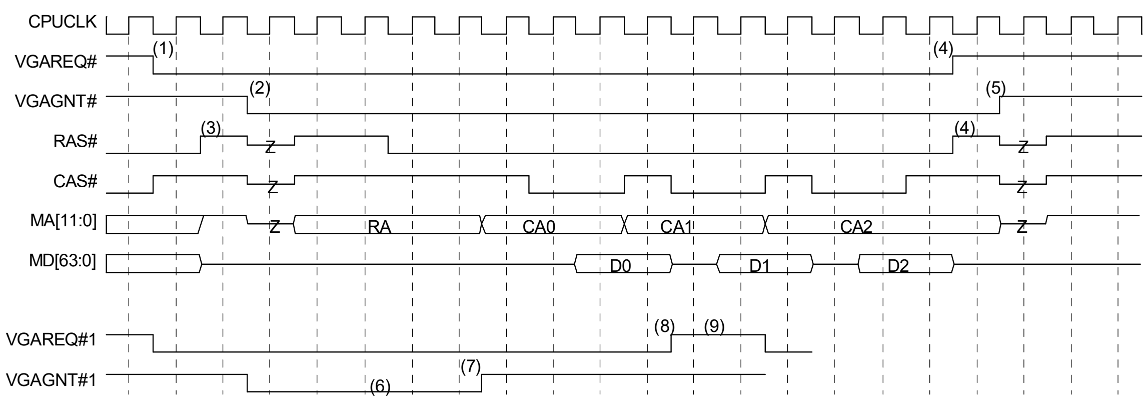


Figure 2.1 Arbitration Timing (low priority)

NOTE:

(1) Graphics controller issues VGAREQ# to request DRAM bus.
(2) If no one is accessing DRAM, 5511 asserts VGAGNT# two CPUCLKs later.
(3) 5511 drives RAS#, CAS#, MA, and RAMW# high one CPUCLK then tri-states them one CPUCLK to release DRAM bus to graphics controller.
(4) Graphics controller deasserts VGAREQ# when transfer finished, and drives RAS#, CAS#, MA, and RAMW# high one CPUCLK then tri-states them one CPUCLK.
(5) 5511 deasserts VGAGNT# one CPUCLK after VGAREQ# is deasserted.
(6,7) If 5511 wants to access DRAM when graphics controller is accessing DRAM, the VGAGNT# will be deasserted three CPUCLKs later.
(8) When VGAGNT# is deasserted, the graphics will release the DRAM bus after VGA cycle transfer is completed by deasserting VGAREQ#.
(9) VGAREQ# must be driven high two CPUCLKs at least.

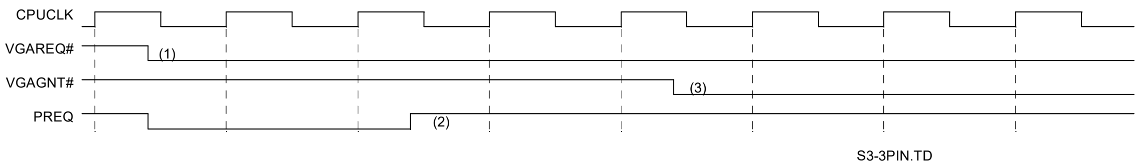


Figure 2.2 3-Pin Arbitration

NOTE:

(1) VGA requests memory control.
(2) VGA asserts PREQ to set the high priority.
(3) Chipset asserted the VGAGNT, and set as high priority of the VGAREQ#.

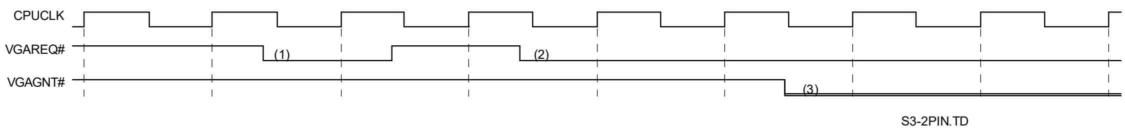


Figure 2.3 2-Pin Priority Arbitration

NOTE:

(1) VGA asserts the VGAREQ#.
(2) VGA deasserts the VGAREQ#, then asserts the VGAREQ# again.
(3) Chipset detects 2 times of VGAREQ# asserted, then asserts the VGAGNT# and sets the priority to the high priority. The maximum high priority latency for VGA to access DRAM is 30 CLKS.

2.2.2 DRAM Configuration

SiS 5511 implements several DRAM configuration registers (DBRs) to store information, such as DRAM size, DRAM type, and starting memory address. These registers are located in 5511's PCI configuration space, and graphics controller can read these registers through PCI interface. The graphics controller must read the DRAM configuration information for proper memory operation of RAS#, CAS#, and MA.

Shared memory location

When hooking with SiS 6205, the location of shared memory can be relocated in the whole memory. When hooking with other VGA controllers, the location of shared memory can only be at the top of the specialized bank. Since most VGA controllers can only support one set of CAS# signals, and which signals can only be connected to either even bank or odd bank. It is necessary to modify BIOS to support different shared memory location. The following examples are used to explain the different situation.

Example 1: 551x + 6205 (two banks of DRAM and the total size is 16MB, 1MB shared to VGA.

Shared memory is located at the top of overall system memory.

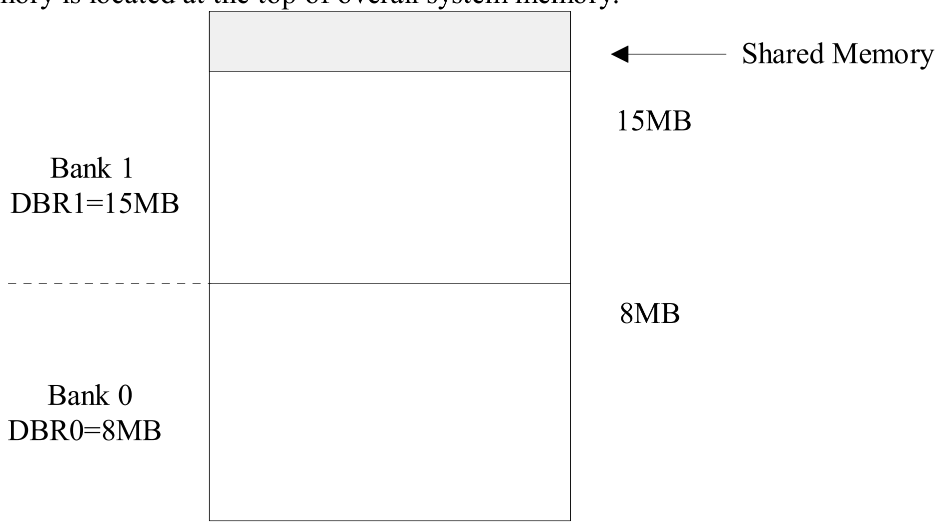


Figure 2.4 Shared Memory Location

Example 2: 551x + 6205 (two banks of DRAM and the total size is 16MB, 1MB shared to VGA.

shared memory is located at the top of the first bank

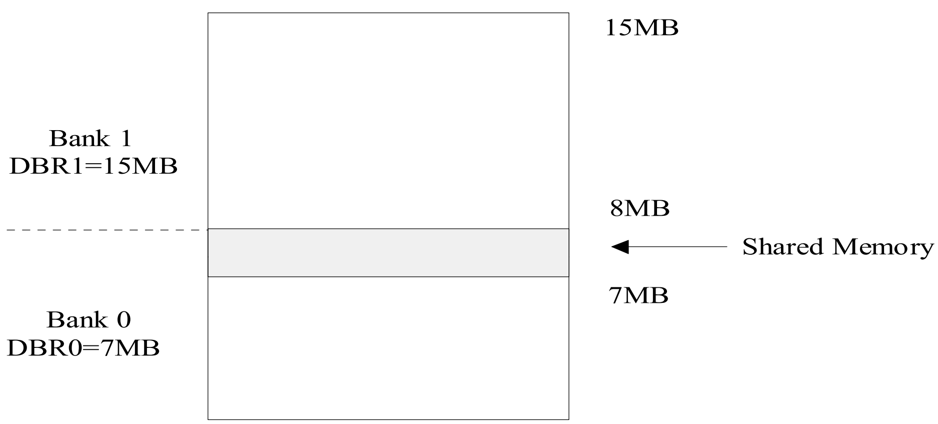


Figure 2.5 Shared Memory Location

Example 3: 551x + other VGAs

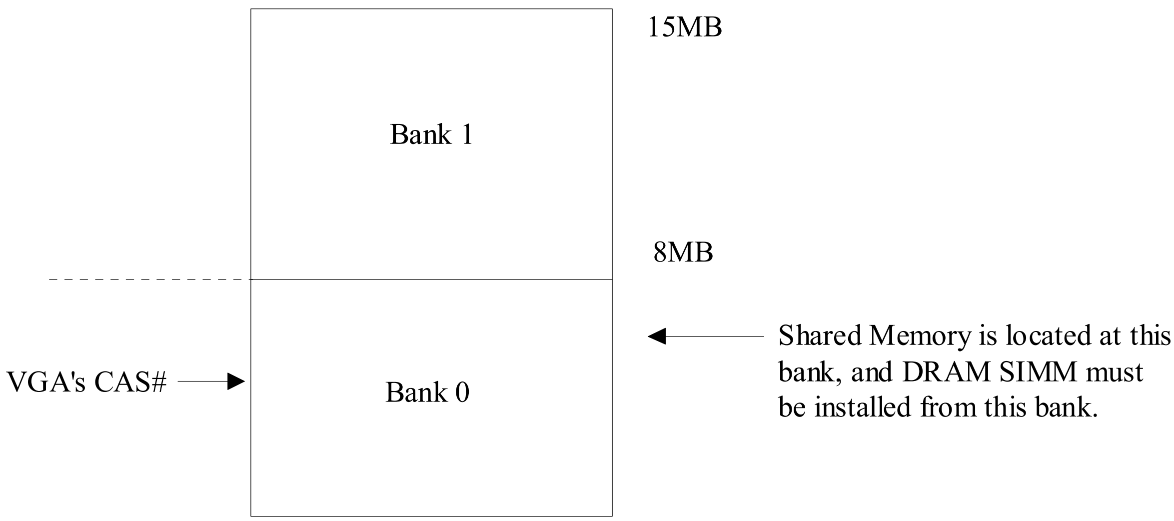


Figure 2.6 Shared Memory Location

Example 4: 551x + other VGAs

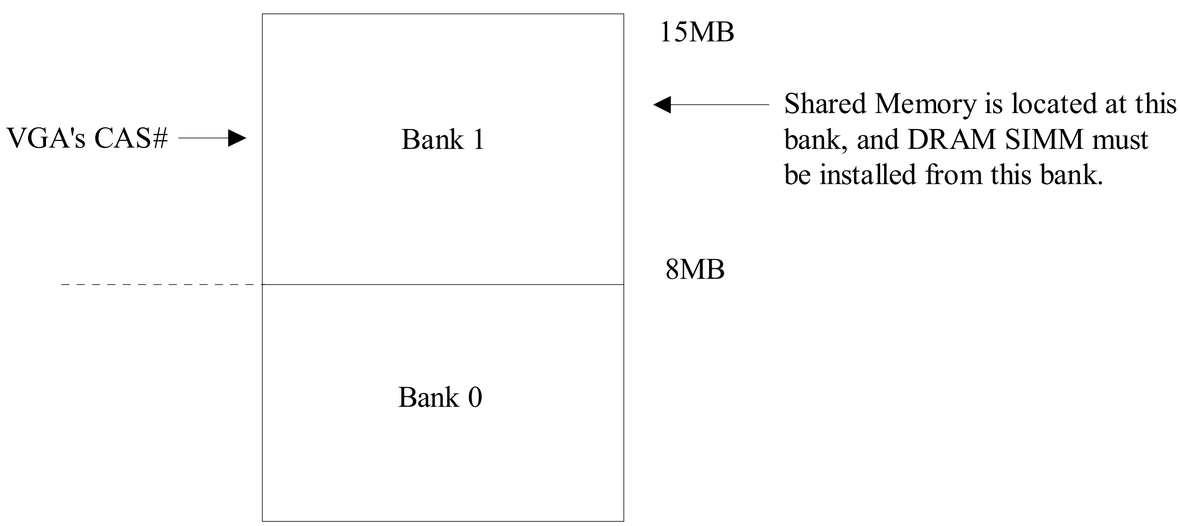


Figure 2.7 Shared Memory Location

2.2.3 DRAM Scramble Table

The DRAM scramble table contains information for memory address mapping. This table provides the translation between CPU host address and memory Row and Column address. Graphics controller needs to follow the DRAM scramble table for the proper MA mapping.

SiS5511 supports two different memory address mapping, 64-bit non-interleave mapping (please refer to Table 2-7 in Part II) and 32-bit mapping (please refer to Table 2-9 in Part II) in the shared memory architecture.

Table 2-1 shows the address mapping when 551x hooks with other VGAs.

Table 2-1: Memory Address Mapping

	MA11	MA10	MA9	MA8	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
CAS				A11	A10	A9	A8	A7	A6	A5	A4	A3
RAS				A20	A19	A18	A17	A16	A15	A14	A13	A12

2.2.4 DRAM Access from CPU

Since CRT has very limited tolerance on memory access latency, it is very important for graphics controller to access DRAM at certain time interval. When graphics controller is granted the DRAM access, it is the responsibility of graphics controller to get as much needed data before the DRAM access is transferred to other requesters.

The implementation of SiS5511 allows CPU accesses L2 cache concurrently with graphics controller accesses DRAM. The maximum latency before CPU is granted for memory access is defined by the maximum access length by the graphics controller. This value depends on the graphics controller that SiS5511 is interfaced with.

Graphics controller may request memory access during CPU accessing system memory. The CPU access will be terminated upon the completion of one cache line. Next address signal will not be generated to CPU by 5511 under this condition.

2.2.5 Memory Type

The memory types supported by SiS5511 and SiS6205 are included in Table 2-7 and Table 2-9 in Part II, when SiS551x hooks with other VGAs, the Supported DRAM types are 256K x N(9x9), 512K x N(10x9), 1M x N(10x10, 11x9), 4M x N(11x11), and 16M x N(12x12), pull-up resistor(s) may be needed for high order bit(s) of MA bus.

2.2.6 DRAM Interface Changes for Shared Memory Support

For the shared memory system, the basic memory requirement is 4 M bytes. For shared memory 32 - bit SIMM configuration, 6205 can only work on 1M byte DRAM. Due to limited memory bandwidth, only the following resolutions are available:

Display Size	Colors Shades	Frame Rate
640x480	256	60, 72, 75
640x480	32K	60, 72, 75
640x480	64K	60, 72, 75
640x480	16M	60, 72, 75
800x600	256	56, 60, 72, 75
800x600	32K	56, 60, 72, 75
800x600	64K	56, 60, 72, 75
1024x768	256	87, 60, 70, 75
1280x1024	16	89, 60

It is not recommended to use 32-bit mode in shared memory system, because the performance degradation is significant.

3. BIOS

3.1 BIOS Interface

GUI shares the same BIOS interface with the system. Since the frame buffer is now shared with system memory, the BIOS interface is then moved to ISA bus. When VGA BIOS is accessed by a PCI master (either CPU through 5511 or 6205 as PCI master), 5513 will do subtractive decode and claims the PCI cycle. Depending on the read or write operation, data will be channeled through 5513 and put onto SD bus.

3.2 Initialization of Shared Memory

Upon power up, System BIOS will invoke the Video BIOS for the initialization of graphics controller. Video BIOS will then either 1) reads pre-defined CMOS locations to determine the frame buffer size, then boots graphics or 2) executes PCI configuration cycles to pull DRAM configuration information from the 5511 DRAM Bank Registers (DBRs) and DRAM Scramble Table out of the pre-defined chipset PCI configuration space. Once DRAM information is available to Video BIOS, booting of the graphics controller is proceeded as usual.

When control is returned from the Video BIOS back to System BIOS, System BIOS determines total system memory installed and reports available memory to the system as total memory size minus frame buffer size.

4. Pin Definition of 5511 Shared Memory Interface

CPUCLK:

The CPUCLK is generated from the clock generator and is fixed at 50, 60, or 66Mhz. The chipset memory controller uses this input as the reference clock. For the GUI, this pin is used for two purposes.

Synchronous mode:

The GUI memory clock is synchronous with CPUCLK (or MCLK is equal to CPUCLK). The interface signals of VGAREQ# and VGAGNT# are synchronized with CPUCLK, therefore,.5511 needs no synchronization. This mode is recommended for a better performance.

Asynchronous mode:

The GUI memory controller runs with MCLK while the chipset memory controller runs with CPUCLK. Since MCLK and CPUCLK are two independent clocks, VGAREQ# and VGAGNT# need to be synchronized with CPUCLK. The synchronization can be done by either VGA controller or 5511.

VGAREQ# (pin 193):

Graphics controller issues VGAREQ# to 5511 to request DRAM bus.

VGAGNT# (pin 190 or 191):

5511 grants DRAM bus to graphics controller by asserting VGAGNT#.

NOTE The main memory is shared between the chipset and GUI through the VGAREQ# and VGAGNT# handshaking protocol. Whenever the GUI wants to access the memory, it generates the VGAREQ# active to the chipset. The chipset grants the memory bus to the GUI by asserting the VGAGNT# active low. Note that the SiS chipset supports both Trident's one-priority and two-priority scheme. The chipset may request the memory bus while the memory bus is granted to the GUI. This is done by the chipset to deassert the VGAGNT# from active low to inactive high while the VGAREQ# is still active (low). The GUI, when detects this situation, shall release the memory bus back to the chipset as soon as possible by deasserting the VGAREQ# to inactive high.

MA[11:9]:

Need external pull-high resistors.

RAS#:

All the RAS# signals are floated by the chipset next clock after the VGAGNT# is active low. In order to reach the best system performance; the chipset automatically detects the fastest DRAM (such as EDO, instead of fast page mode) and assigns that particular DRAM as the logical bank 0 memory (address). This is also the memory that is shared by the chipset and GUI. The GUI RAS# signal therefore need to connect to the memory bank RAS# input.

CAS[7:0]#:

All the CAS[7:0] signals are floated by the chipset next clock after the VGAGNT# is active low.

RAMW#:

This signal is floated by the chipset next clock after the VGAGNT# is active low.

PREQ (pin 192)

This signal is used to hook with S3 VGA Controller. It is described in Register 84.

5. New Registers Definition

Register 88h (R/W): DRAM Boundary Register (DBR) Resolution Selection

Bits 7:0	**DRAM Boundary Address A20**
Bit 7	**DRAM Bank Register 3-1**
Bit 6	**DRAM Bank Register 3-0**
Bit 5	**DRAM Bank Register 2-1**
Bit 4	**DRAM Bank Register 2-0**
Bit 3	**DRAM Bank Register 1-1**
Bit 2	**DRAM Bank Register 1-0**
Bit 1	**DRAM Bank Register 0-1**

Bit 0 **DRAM Bank Register 0-0**

NOTE: This register must be used together with Register 70h, 72h, 74h, 76h, 78h, 7Ah, 7Ch, 7Eh.

Register 89h (R/W): CPU Interface Register

Bit 7 **PCI master data transfer interruption. When enabled, the PCI data transfer is disconnected and the memory bus is granted to the GUI as soon as any outstanding CPU write back cycle is completed. If disabled, the memory bus will not be released to the GUI until all the pending posted write data is flushed to the DRAM.**

0: Enable

1: Disable

Bit 6 **Shared memory protocol operation selection**

0: Synchronous operation

1: Asynchronous operation

Bit 5 **GUI access memory latency**

0: 30 CPUCLK maximum.

1: 60 CPUCLK maximum.

Bit 4 **DRAM address mapping table**

0: Use SiS table

1: Non SiS table (such as Trident GUI table).

Bits 3:1 **Shared memory size**

000: 1M

001: 0.5M

010: 1.5M

011: 2M

100: 2.5M

101: 3M

110: 3.5M

111: 4M

Bit 0 **Shared memory mode**

0: Disable

1: Enable

Register 0A0h (R/W): DRAM Boundary Register Resolution Selection (DBR)

This register defines the resolution of DBR7-0 down to 512K byte.

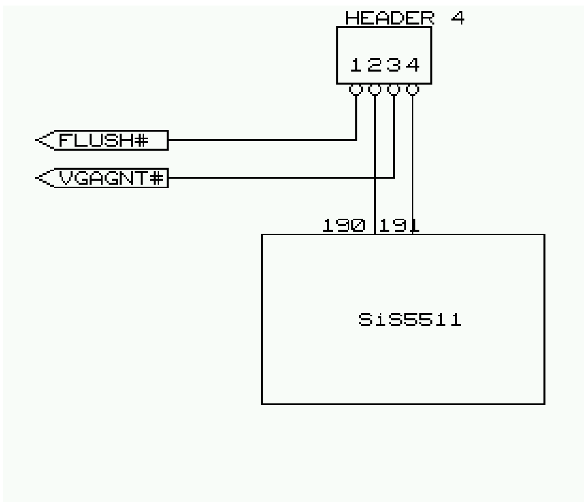
Register 0A1h (R/W): GUI Interface Register

Bits 7:6 Reserved.

Bit 5 VGAGNT# output select.

0: VGAGNT# from INTR1 pin.

1: VGAGNT# from INTR2 pin.



NOTE: 12, 34: ON VGAGNT# from INTR2
23: ON VGAGNT# from INTR1

Figure 5.1 VGAGNT# Signal Output Select

Bits 4:2 The bank corresponding to the share memory.

Bit 1 Trident VGAGNT# minimum pulse width of 16 CPUCLKs.

0: Disable

1: Enable

Bit 0 Priority scheme selection

0: two-priority scheme.

1: one-priority scheme.

Register 0A2h (R/W): Direct Access Register

This register defines the address range that allows the CPU to direct accesses shared memory without going through the PCI bus.

Bits 7-3 **These bits define the direct access address of A23-A19.**

Bit 2 **Direct access area read enable.**

Bit 1 **Direct access area write enable.**

Bit 0 **Direct access area PCI master access enable.**

Register 0A3h (R/W): Direct Access Register

Bits 7:0 **Define shared memory hole area A31~A24**

NOTE: Shared memory hole concept

When CPU accesses off-board DRAM, it will be forwarded to PCI side then write to or read from shared memory area data through VGA chip. If we define a shared memory hole area , any logical area, we can access the shared memory area by remapping the logical area to physical area of the shared memory through system chip (5511C). We can save some time wasted on PCI bus.

Register 84

Bit 0 **3-pin priority support (PREQ)**

0: Not Support

1: Support

NOTE: When using S3 VGA Controller, pin 192 is used as PREQ.

Part II SiS5511/2/3 Pentium/P54C PCI/ISA Chipset

1. 5511/5512/5513 Overview

SiS5511	PCI/ISA Cache Memory Controller (**PCMC**)
SiS5512	PCI Local Data Buffer (**PLDB**)
SiS5513	PCI System I/O (**PSIO**)

A whole set of the SiS5511, 5512, and 5513 provides fully integrated support for the Pentium PCI/ISA system. The chipset is developed by using a very high level of function integration and system partitioning. With the SiS5511, SiS5512, and SiS5513 chipset, only about 9 TTLs (include 3 DRAM address buffers) are required to implement a low cost, high performance, Pentium PCI/ISA system. Figure 1.1 shows the system block diagram.

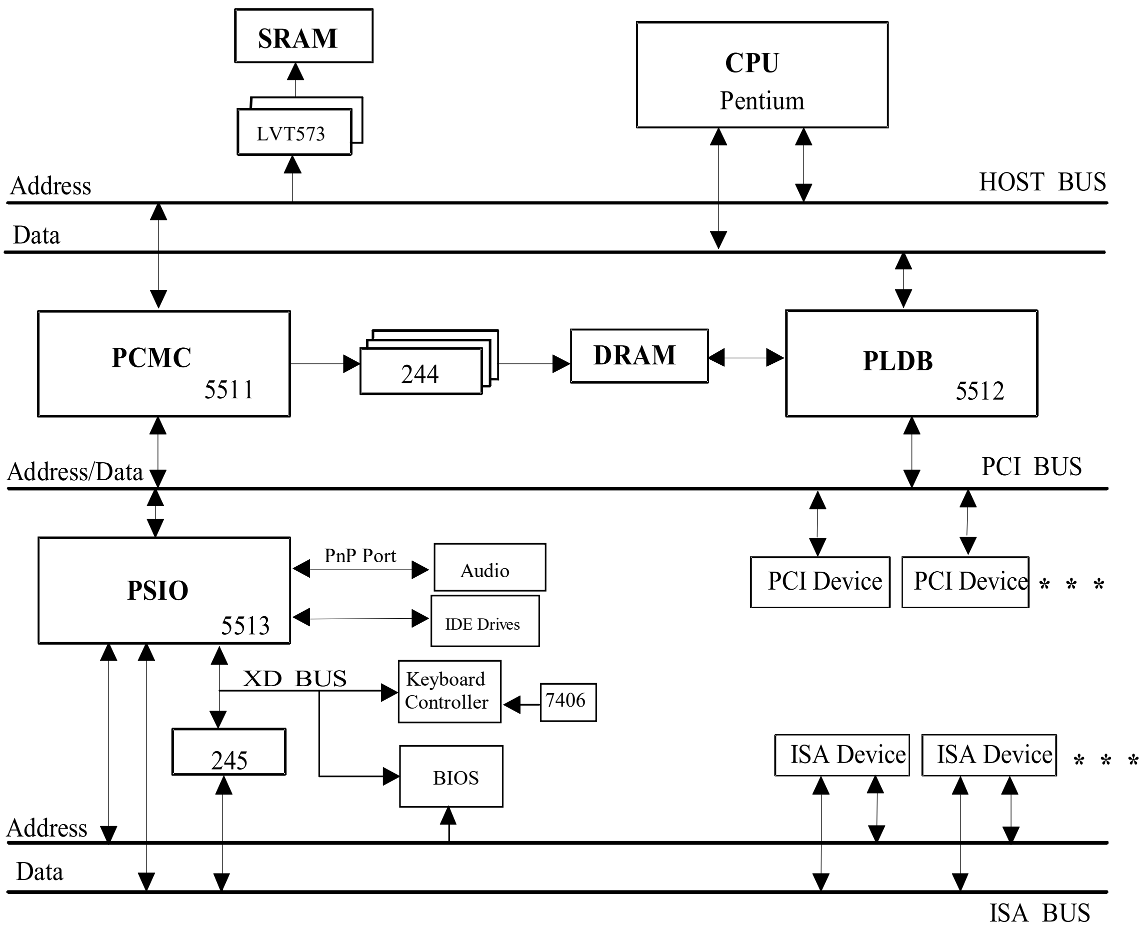


Figure 1.1 System Block Diagram

2. SiS5511 PCI/ISA Cache Memory Controller

2.1 SiS5511 Introduction

2.1.1 Features

- **Supports Intel Pentium CPU and other compatible CPU at 66/60/50MHz (external clock speed)**
- **Supports Shared Memory Architecture**
  - Direct Memory Accesses
  - Shared Memory Area 0.5M, 1M, 1.5M, 2M, 2.5M, 3M, 3.5M, 4M.
- **Supports VGA Shared Memory Function**
- **Supports Slice MP Protocol**
- **Supports the Pipelined Address Mode of Pentium CPU.**
- **Integrated Second Level (L2) Cache Controller**

- Write Through and Write Back Cache Modes
- 8 bits or 7 bits Tag with Direct Mapped Cache Organization
- Supports Standard, Burst and Pipelined Burst SRAMs.
- Supports 64 KBytes to 1 MBytes Cache Sizes.
- Pipeline Burst SRAM 3-1-1-1-1-1-1-1
- Cache Read/Write Cycle of 3-2-2-2 or 4-2-2-2 Using Standard SRAMs at 66 MHz.
- Cache Read/Write Cycle of 3-1-1-1 Using Burst or Pipelined Burst SRAMs at 66 MHz.

- **Integrated DRAM Controller**
  - Supports 4 Banks of SIMMs, the memory size is from 2MBytes up to 512Mbytes. (5511 decodes memory space up to 1 Gbytes actually, but limited by current DRAM modules 512Mbytes is the maximum now.)

- Supports 256K/512K/1M/2M/4M/16M x N 70ns Fast Page Mode and EDO DRAM
- Supports 4K Refresh DRAM
- Supports 3V or 5V DRAM.
- Supports Symmetrical and Asymmetrical DRAM.
  - Supports Half-Populated (32 bits) Configuration for Bank 0
  - Supports Concurrent Write Back
  - Bank Interleave Mode for 6-2-2-2 Read Cycle
  - Supports FP DRAM 6-3-3-3 Burst Read Cycle.
  - Supports EDO Type DRAM.
  - Supports 6-2-2-2 Burst Read Cycle.
  - Supports 50 Mhz 5-2-2-2 DRAM Cycle.
  - Supports X-2-2-2/X-3-3-3 Burst Write Cycle.
  - Supports Read Cycle Power Saving Mode.
  - Table-free DRAM Configuration, Auto-detect DRAM size, Bank Density, Single /Double sided DRAM, EDO/ FP DRAM for each bank
  - Supports CAS before RAS "Intelligent Refresh"
  - Supports Relocation of System Management Memory
  - Optional Parity Checking
  - Programmable CAS# Driving Current
  - Fully Configurable for the Characteristic of Shadow RAM (640 KByte to 1 MByte)

- **Two Programmable Non-Cacheable Regions**

- **Option to Disable Local Memory in Non-Cacheable Regions**
- **Shadow RAM in Increments of 16 KBytes**
- **Supports SMM Mode of CPU.**
- **Supports CPU Stop Clock.**
- **Supports Break Switch.**
- **Provides High Performance PCI Arbiter.**
  - Supports 4 PCI Master.
  - Supports Rotating Priority Mechanism.
  - Hidden Arbitration Scheme Minimizes Arbitration Overhead.

- Supports Concurrency between CPU to Memory and PCI to PCI.

- **Integrated PCI Bridge**

- Supports Asynchronous PCI Clock.
- Translates the CPU Cycles into the PCI Bus Cycles
- Provides CPU-to-PCI Read Assembly and Write Disassembly Mechanism
- Translates Sequential CPU-to-PCI Memory Write Cycles into PCI Burst Cycles.
- Zero Wait State Burst Cycles.
- Provides A Prefetch Mechanism Dedicated for IDE Read.
- Supports Advance Snooping for PCI Master Bursting.
- Maximum PCI Burst Transfer from 256 Bytes to 4 KBytes.

- **208-Pin PQFP.**
- **0.6µm CMOS Technology.**

2.1.2 Functional Block Diagram

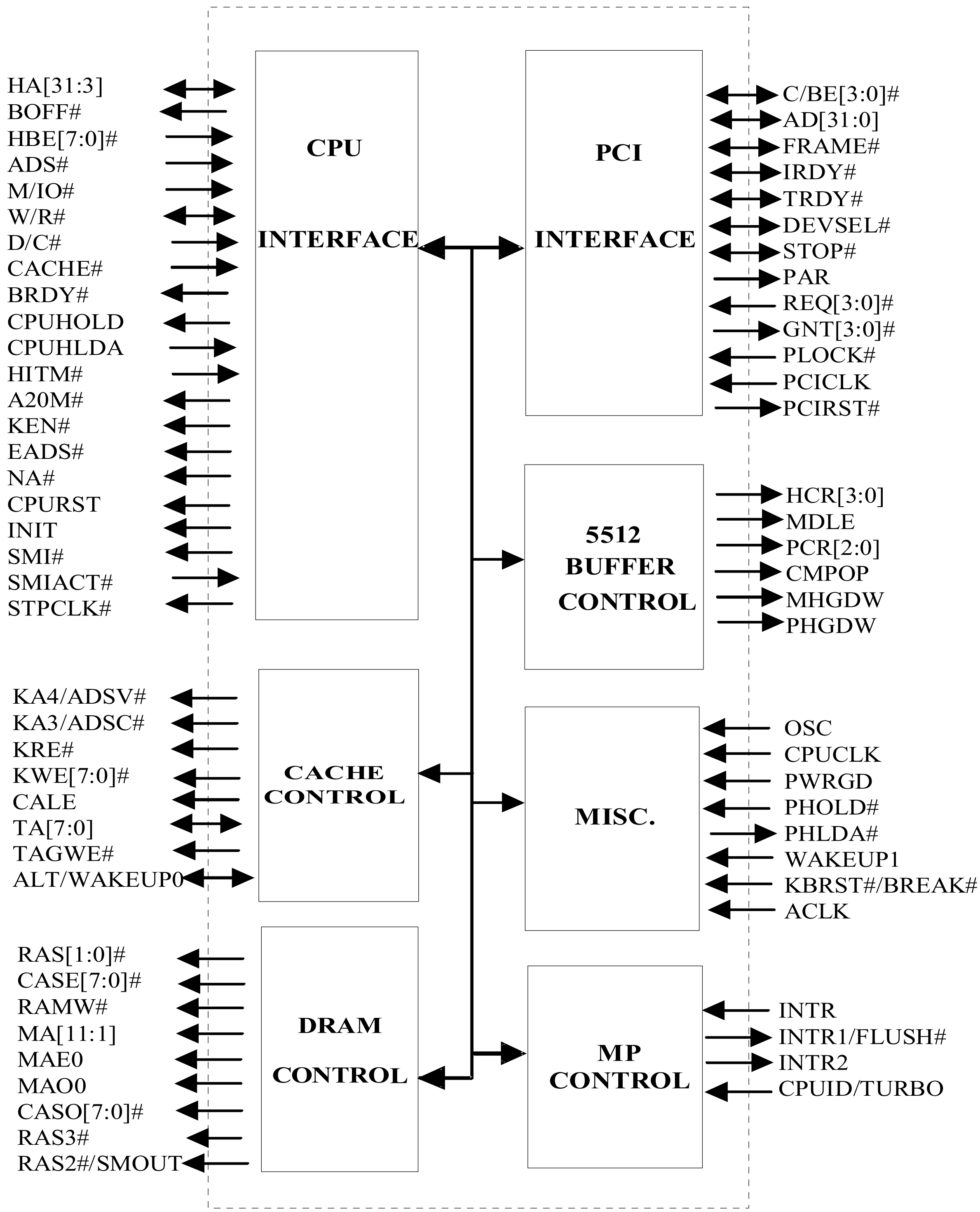


Figure 2.1 SiS5511 Function Block Diagram

2.2 Functional Description

The SiS5511(PCMC) bridges between the host bus and the PCI local bus. The SiS5511 (PCMC) monitors each cycle initiated by the CPU, and forwards it to the PCI bus if the CPU cycle does not target at the local memory. For the CPU or the PCI to the local memory cycles, the built-in Cache and DRAM Controller assumes the control to the secondary cache, DRAMs, and the SiS5512 (PLDB). The SiS5511 (PCMC) also guides the SiS5512 (PLDB) for correct data flow. All of the Green PC functions are provided.

2.2.1 CPU Interface

The SiS5511 is designed to support Pentium CPU host interface at 66.667/60/50MHz. The host data bus and the DRAM bus are 64-bit wide.

The SiS5511 supports the pipelined addressing mode of the Pentium CPU by issuing the next address signal, NA#. NA# is only generated in the following cases:
(a) Burst read L2 cache or DRAM, (b) Single read DRAM and (c) Burst write L2 cache.

The PCMC supports the CPU L1 write back (WB) or write through (WT) cache policies and the PCMC L2 WB or WT cache policies. The L1 cache is snooped by the assertion of EADS# when the CPU is put in the HOLD state.

The PCMC issues CPUHOLD to the Pentium CPU in response to the assertion of PCI master requests(REQ[3:0]#, and PHOLD#). Upon receiving the CPUHLDA from the CPU, it does not immediately assert GNT[3:0]# or PHLDA# until both the CPU to PCI posted write buffer and the memory write buffer are empty. During inquire cycles, the CPUHOLD may be negated temporarily to allow the CPU to write back the inquired hit modified line to L2 or DRAM.

In order to support Dual Processor application the SLiC/MP interrupt control module has been integrated into 5511. The SLiC/MP interrupt control module consists of IPI control block and the IPI vector register. The detailed function of the SLiC/MP interrupt controller is described below.

IPI Control

An Inter-processor interrupt (IPI) can be initiated by performing a non-cacheable memory write operation to the IPI dispatch register. An IPI dispatch instruction always takes 3 CPU clocks. For example, CPU1 dispatches an IPI interrupt by asserting ADS# in T1. The SLiC module will responds this IPI by asserting INTR2 and BRDY# in T2 and T3, respectively. The SLiC module will provide the IPI vector to the interrupted CPU in 2nd interrupt acknowledge cycle. Because the SLiC module operates in Auto-End-of-Interrupt mode, it will deassert INTRx after 2nd interrupt acknowledge cycle. Even though there is an 8259 interrupt pending, the SLiC module will also deassert INTRx for 3 CPU clocks after 2nd interrupt acknowledge cycle. Notice that if the consecutive IPIs are dispatched to the same CPU before the first IPI acknowledge cycle is finished, these IPIs will be treated as one IPI.
The priority of IPI is higher than that of the interrupts from 8259. The IPI will get service from SLiC even though the INTRx is asserted by 8259 interrupt requests. For example, the INTRx pin is

asserted by 8259 interrupt request. However, the IPI arrives before the interrupted CPU initiate the INTA cycles. The SLiC will provide the IPI vector to the interrupted CPU instead of 8259 interrupt vector.

IPI Vector Register

This register contains the vector which will provide to IPIed CPU directly by SLiC module at 2nd INTA cycle. The operating system is responsible for initializing this register. This register is actually located in 5512.

The SLiC module contains 7 memory mapped registers (one is located in 5512). These registers must be accessed by non-cacheable memory read/write operations. These memory mapped registers are relocatable by changing the content of the global register. for example, if we change the content of global register from F(hex) to C(hex). The address of IPI dispatch register will change from FFC00000h to CFC00000h. For more detailed information, please refer to register description.

2.2.2 Cache Controller

The built-in L2 Cache Controller uses a direct-mapped scheme, which can be configured as either in the write through or write back mode. Standard asynchronous, burst and pipelined burst SRAMs are supported.

SiS5511 supports SRAM types auto-detection and auto-sizing. Table 1 shows the cache sizes that are supported by the SiS5511 when using asynchronous SRAM, with the corresponding TAG RAM sizes, data RAM sizes, and cacheable memory sizes. Tables 2 and 3 summarize the recommended speed setting when either the standard SRAMs or the Burst SRAMs are used.

Table 2-1 Cache Size

Cache Size	Data RAM	Tag RAM	Alter RAM	Cacheable Size
64K	8Kx8x8	2Kx8	2Kx1	16M
256K	32Kx8x8	8Kx8	8Kx1	64M
512K	64Kx8x8	16Kx8	16Kx1	128M
1M	128Kx8x8	32Kx8	32Kx1	256M

The PCMC also provides an alternative to save the dirty SRAM chip. This is accomplished by sharing the alter bit with tag address bits in the same 8-bit wide TAG RAM. System uses this implementation supports 7 tag address bits and 1 dirty bit. By doing so, the cacheable local memory sizes are reduced to half of the original sizes as indicated in Table 1.

In reality, the L2 Cacheable DRAM Size is determined by:
1) Max. L2 Cacheable Size as described in table 1.
2) Noncacheable Area defined in register 56h, 57h, 58h and 59h and
3) C, D, E, F Segment Cacheability defined in registers 80h~86h.

But, the L1 Cacheable size is only determined by 2), 3), and the maximum DRAM size, i.e., 512M bytes. Thus, the cycles with address ranging over the L2 Cacheable Size but within the 512M bytes can also be cacheable to L1. The behavior of KEN# is ruled by the L1 Cacheability. Note that only code of C, D, E, F segment is cacheable to L1/L2, and the data portion of C, D, E, F segment is not cacheable to L1/L2.

Table 2-2 Asynchronous SRAM Speed Settings

	Data RAM Speed	Tag RAM Speed	Read Performance	Write Performance
66 MHz	15ns	12ns	3-2-2-2	3-2-2-2
66 MHz	15ns	15ns	4-2-2-2	4-2-2-2
60 MHz	15ns	12ns	3-2-2-2	3-2-2-2
60 MHz	20ns	20ns	4-2-2-2	4-2-2-2
50 MHz	20ns	20ns	3-2-2-2	3-2-2-2
50 MHz	20ns	20ns	4-2-2-2	4-2-2-2

Table 2-3 Synchronous SRAM Speed Settings

	Data RAM Speed			Tag RAM Speed			Read Performance	Write Performance
	66 MHz	60 MHz	50 MHz	66 MHz	60 MHz	50 MHz		
Burst SRAM	15ns	15ns	20ns	12ns	12ns	20ns	3-1-1-1	3-1-1-1
				15ns	15ns	20ns	4-1-1-1	4-1-1-1
Pipeline SRAM	15ns	15ns	20ns	12ns	12ns	20ns	3-1-1-1	3-1-1-1
				15ns	15ns	20ns	4-1-1-1	4-1-1-1

NOTE:(1) The SRAM parameters of data RAMs showed in above table are "cycle time".
(2) Use asynchronous SRAM for Tag RAM.

Table 2-4 Cache Signals and their Relative Clock Sources

Asynchronous SRAM	KRE#	CPUCLK
	KWE#	Cache hit: ACLK, otherwise: CPUCLK
	KA3	Read cache hit: ACLK
	KA4	Otherwise: CPUCLK
Synchronous SRAM	KRE#	Always refers to CPUCLK
	KWE#	
	ADSC#	
	ADSV#	

SRAM Address Mapping

Table 2-5 TAG=8-bit

	64K	256K	512K	1M
TA7	HA23	HA23	HA23	HA23
TA6	HA22	HA22	HA22	HA22
TA5	HA21	HA21	HA21	HA21
TA4	HA20	HA20	HA20	HA20
TA3	HA19	HA19	HA19	HA27
TA2	HA18	HA18	HA26	HA26
TA1	HA17	HA25	HA25	HA25
TA0	HA16	HA24	HA24	HA24

Table 2-6 TAG=7-bit

	64K	256K	512K	1M
TA6	HA22	HA22	HA22	HA22
TA5	HA21	HA21	HA21	HA21
TA4	HA20	HA20	HA20	HA20
TA3	HA19	HA19	HA19	HA23
TA2	HA18	HA18	HA23	HA26
TA1	HA17	HA23	HA25	HA25
TA0	HA16	HA24	HA24	HA24

NOTE: TA7 acts as ALT.

2.2.3 DRAM Controller

The 5511 can support up to 512MBytes (4 banks) of DRAM. Each bank could be single or double sided 64 / 72 bits (with / without parity) FP (Fast Page mode) DRAM or EDO (Extended Data Output) DRAM. Half populated bank is also supported for bank 0.

The installed DRAM type can be 256K, 512k, 1M, 2M, 4M or 16M bit deep by n bit wide DRAM, and both symmetrical and asymmetrical type DRAM are supported. It is also permissible to mix the EDO DRAM and FP DRAM bank by bank and the corresponding DRAM timing will be switched automatically according to register setting. If the FP DRAM is installed symmetrically, then the 5511 will auto-configure it to Bank-interleave mode to speed the system performance.

DRAM Configuration

The SiS5511 can support single sided or double sided DRAM modules for each bank. The basic configurations are shown below.

Double-sided DRAM:

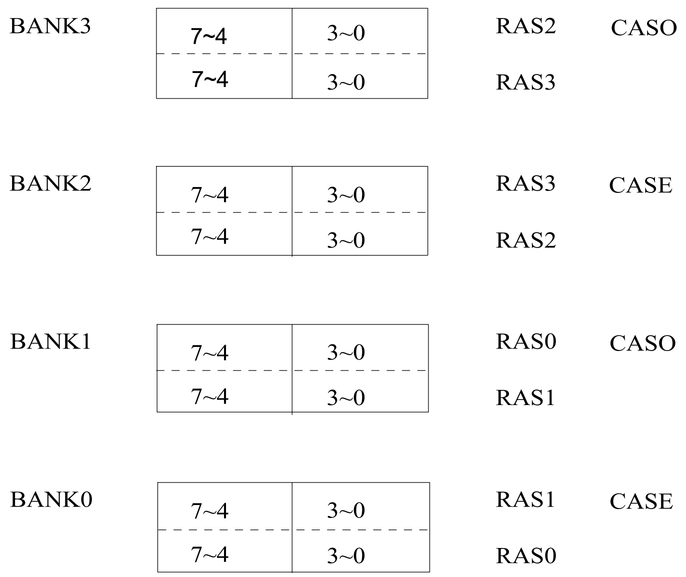


Figure 2.2 SiS5511 Supports Double Sided DRAM

For single-sided DRAM

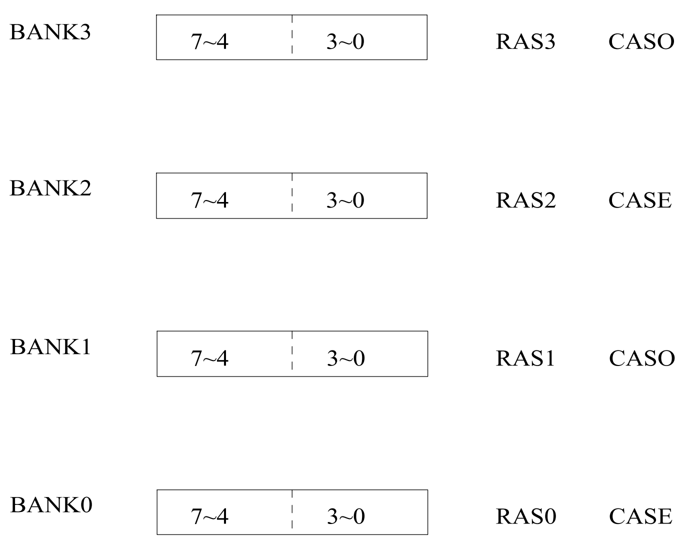


Figure 2.3 SiS5511 Supports Single Sided DRAM

The DRAM address MA[11:1] are connected to each bank and MAE0, MAO0 only go to the respective bank. The CASE[7:0]# are connected to Bank 0, 2 and CASO[7:0]# are connected to Bank 1, 3. Thus the CASE and CASO can be interleaved during the burst cycle in interleave mode.

The SiS5511 can also support two way bank interleaved mode for FP DRAM. That means if DRAM's type and size of bank pair are the same, the bank interleaved mode is applicable to bank pair. There are two bank pairs, bank pair 1 includes bank 0 and bank 1, bank pair 2 includes bank2 and bank 3. In single-sided interleave mode, the RAS0# and RAS1 or RAS2# and RAS3# will be asserted at the same time. In double-sided interleave mode, there is only one RAS# will be asserted at one time to avoid data contention.

For the half-populated bank, it is only supported in the even side of Bank 0 for single sided DRAM.

There are several DBRs (DRAM Bank Register). The DRAM type is recorded in DBR which includes the status of FP/EDO, Double/Single sided, Half/Full populated and Symmetrical/Asymmetrical DRAM for each bank. If the even SIMM and odd SIMM DRAM type are different, the type of the smaller size DRAM is recognized. The accumulated DRAM density is programmed to DBRs. Each of them corresponds to the different side of each bank.
DBR0-0 = Total amount of even Side of bank 0
DBR0-1 = Total amount of bank 0
DBR1-0 = Total amount of bank 0 + even Side of bank 1
DBR1-1 = Total amount of bank 0 + bank 1
DBR2-0 = Total amount of bank 0 + bank 1 + even Side of bank 2
DBR2-1 = Total amount of bank 0 + bank 1 + bank 2
DBR3-0 = Total amount of bank 0 + bank 1 + bank 2 + even Side of bank 3
DBR3-1 = Total amount of bank 0 + bank 1 + bank 2 + bank 3

Restrictions:
We don't support the following DRAM combinations:
1. Different DRAM types (EDO or FP) in different SIMMs of each bank,
2. EDO interleaved mode,
3. Double-sided half-populated DRAM,
4. Interleaved mode between two different DRAM configurations (single-sided/double-sided, DRAM size, or symmetry/asymmetry).

DRAM Address Mapping

The following tables show the different address mapping for different DRAM configuration.

Table 2-7 Non-Interleave 64 Bits (FP, EDO/BEDO)

	256K Sym.		1M Sym.		4M Sym.		16M Sym.		S3 SYM	
MA	CAS	RAS	CAS	RAS	CAS	RAS	CAS	RAS	CAS	RAS
0	4/3	12	4/3	22	4/3	22	4/3	22	3	12
1	11/4	13	11/4	13	11/4	24	11/4	24	4	13
2	3/11	14	3/11	14	3/11	14	3/11	26	5	14
3	5	15	5	15	5	15	5	15	6	15
4	6	16	6	16	6	16	6	16	7	16
5	7	17	7	17	7	17	7	17	8	17
6	8	18	8	18	8	18	8	18	9	18
7	9	19	9	19	9	19	9	19	10	19
8	10	20	10	20	10	20	10	20	11	20
9	NA	NA	12	21	12	21	12	21	*NA*	*NA*
10	NA	NA	NA	NA	13	23	13	23	*NA*	*NA*
11	NA	NA	NA	NA	NA	NA	14	25	*NA*	*NA*

	512K Asym.		1M Asym.		2M Asym.		4M Asym.		S3 Asym	
MA	CAS	RAS	CAS	RAS	CAS	RAS	CAS	RAS	CAS	RAS
0	4/3	12	4/3	12	4/3	22	4/3	22	3	12
1	11/4	13	11/4	13	11/4	13	11/4	13	4	13
2	3/11	14	3/11	14	3/11	14	3/11	14	5	14
3	5	15	5	15	5	15	5	15	6	15
4	6	16	6	16	6	16	6	16	7	16
5	7	17	7	17	7	17	7	17	8	17
6	8	18	8	18	8	18	8	18	9	18
7	9	19	9	19	9	19	9	19	10	19
8	10	20	10	20	10	20	10	20	*NA*	20
9	NA	21	NA	21	12	21	12	21	*NA*	11
10	NA	NA	NA	22	NA	23	NA	23	*NA*	*NA*
11	NA	NA	NA	NA	NA	NA	NA	24	*NA*	*NA*

	Asym. 12x8	Asym. 12x8	Asym. 12x9	Asym. 12x9
MA	CAS	RAS	CAS	RAS
0	4/3	22	4/3	22
1	10/4	13	11/4	13
2	3/10	14	3/11	14
3	5	15	5	15
4	6	16	6	16
5	7	17	7	17
6	8	18	8	18

7	**9**	**19**	**9**	**19**
8	**NA**	**20**	**10**	**20**
9	**NA**	**21**	**NA**	**21**
10	**NA**	**12**	**NA**	**23**
11	**NA**	**11**	**NA**	**12**

Noninterleave, 64 bits, support for S3 Trio64V+

MA
0
1
2
3
4
5
6
7
8
9
10
11

Table 2-8 Interleave, 64 bit

	256K Sym.		**1M Sym.**		**4M Sym**		**16M Sym.**	
MA	CAS	RAS	CAS	RAS	CAS	RAS	CAS	RAS
0	**4**	**21**	**4**	**22**	**4**	**22**	**4**	**22**
1	**11**	**13**	**11**	**23**	**11**	**24**	**11**	**24**
2	**12**	**14**	**12**	**14**	**12**	**25**	**12**	**26**
3	**5**	**15**	**5**	**15**	**5**	**15**	**5**	**27**
4	**6**	**16**	**6**	**16**	**6**	**16**	**6**	**16**
5	**7**	**17**	**7**	**17**	**7**	**17**	**7**	**17**
6	**8**	**18**	**8**	**18**	**8**	**18**	**8**	**18**
7	**9**	**19**	**9**	**19**	**9**	**19**	**9**	**19**
8	**10**	**20**	**10**	**20**	**10**	**20**	**10**	**20**
9	**22**	**27**	**13**	**21**	**13**	**21**	**13**	**21**
10	**24**	**23**	**24**	**27**	**14**	**23**	**14**	**23**
11	**26**	**25**	**26**	**25**	**26**	**27**	**15**	**25**

	512K Asym.		**1M Asym.**		**2M Asym.**		**4M Asym.**	
MA	CAS	RAS	CAS	RAS	CAS	RAS	CAS	RAS
0	**4**	**22**	**4**	**23**	**4**	**22**	**4**	**22**

1	**11**	**13**	**11**	**13**	**11**	**24**	**11**	**25**
2	**12**	**14**	**12**	**14**	**12**	**14**	**12**	**14**
3	**5**	**15**	**5**	**15**	**5**	**15**	**5**	**15**
4	**6**	**16**	**6**	**16**	**6**	**16**	**6**	**16**
5	**7**	**17**	**7**	**17**	**7**	**17**	**7**	**17**
6	**8**	**18**	**8**	**18**	**8**	**18**	**8**	**18**
7	**9**	**19**	**9**	**19**	**9**	**19**	**9**	**19**
8	**10**	**20**	**10**	**20**	**10**	**20**	**10**	**20**
9	**27**	**21**	**27**	**21**	**13**	**21**	**13**	**21**
10	**24**	**23**	**24**	**22**	**27**	**23**	**27**	**23**
11	**26**	**25**	**26**	**25**	**26**	**25**	**26**	**24**

	Asym. 12x8	**Asym. 12x8**	**Asym. 12x9**	**Asym. 12x9**
MA	CAS	RAS	CAS	RAS
0	**4**	**22**	**4**	**22**
1	**11**	**23**	**11**	**24**
2	**10**	**14**	**12**	**14**
3	**5**	**15**	**5**	**15**
4	**6**	**16**	**6**	**16**
5	**7**	**17**	**7**	**17**
6	**8**	**18**	**8**	**18**
7	**9**	**19**	**9**	**19**
8	**27**	**20**	**10**	**20**
9	**25**	**21**	**25**	**21**
10	**24**	**12**	**24**	**23**
11	**26**	**13**	**26**	**13**

Table 2-9 Non-interleave 32 Bits

	256K Sym		**1M Sym.**		**4M Sym.**		**16M Sym**	
MA	CAS	RAS	CAS	RAS	CAS	RAS	CAS	RAS
0	**4**	**12**	**4**	**12**	**4**	**22**	**4**	**22**
1	**2**	**13**	**2**	**13**	**2**	**13**	**2**	**24**
2	**3**	**14**	**3**	**14**	**3**	**14**	**3**	**14**
3	**5**	**15**	**5**	**15**	**5**	**15**	**5**	**15**
4	**6**	**16**	**6**	**16**	**6**	**16**	**6**	**16**
5	**7**	**17**	**7**	**17**	**7**	**17**	**7**	**17**
6	**8**	**18**	**8**	**18**	**8**	**18**	**8**	**18**
7	**9**	**19**	**9**	**19**	**9**	**19**	**9**	**19**
8	**10**	**11**	**10**	**20**	**10**	**20**	**10**	**20**
9	**22**	**21**	**11**	**21**	**11**	**21**	**11**	**21**
10	**24**	**23**	**24**	**23**	**12**	**23**	**12**	**23**
11	**20**	**25**	**22**	**25**	**24**	**25**	**13**	**25**

	512K Asym.		1M Asym.		2M Asym.		4M Asym.	
MA	CAS	RAS	CAS	RAS	CAS	RAS	CAS	RAS
0	**4**	**12**	**4**	**12**	**4**	**22**	**4**	**22**
1	**2**	**13**	**2**	**13**	**2**	**13**	**2**	**13**
2	**3**	**14**	**3**	**14**	**3**	**14**	**3**	**14**
3	**5**	**15**	**5**	**15**	**5**	**15**	**5**	**15**
4	**6**	**16**	**6**	**16**	**6**	**16**	**6**	**16**
5	**7**	**17**	**7**	**17**	**7**	**17**	**7**	**17**
6	**8**	**18**	**8**	**18**	**8**	**18**	**8**	**18**
7	**9**	**19**	**9**	**19**	**9**	**19**	**9**	**19**
8	**10**	**20**	**10**	**20**	**10**	**20**	**10**	**20**
9	**22**	**11**	**22**	**21**	**11**	**21**	**11**	**21**
10	**24**	**23**	**24**	**11**	**24**	**12**	**24**	**23**
11	**21**	**25**	**23**	**25**	**23**	**25**	**25**	**12**

	Asym. 12x8	Asym. 12x8	Asym. 12x9	Asym. 12x9
MA	CAS	RAS	CAS	RAS
0	**4**	**12**	**4**	**22**
1	**2**	**13**	**2**	**13**
2	**3**	**14**	**3**	**14**
3	**5**	**15**	**5**	**15**
4	**6**	**16**	**6**	**16**
5	**7**	**17**	**7**	**17**
6	**8**	**18**	**8**	**18**
7	**9**	**19**	**9**	**19**
8	**23**	**20**	**10**	**20**
9	**25**	**21**	**25**	**21**
10	**24**	**10**	**24**	**12**
11	**22**	**11**	**23**	**11**

DRAM Performance

All the DRAM cycles are synchronous with the CPU clock. The following table shows the different possible speed settings that depend on different DRAM type, RAS# setting, CAS# setting, and so forth.

Table 2-10 DRAM Performance

Cycle Type	**DRAM type**	**66, 60 Mhz**	**50 Mhz**	**Note**
Read Page Hit	EDO	6-2-2-2/ 6-3-3-3	6-2-2-2/ 6-3-3-3	
	FP	6-2-2-2/ 6-3-3-3/ 7-4-4-4	6-2-2-2/ 6-3-3-3/ 7-4-4-4	*1
Read Row miss	EDO	9-2-2-2/ 9-3-3-3	9-2-2-2/ 9-3-3-3	*2
	FP	9-2-2-2/ 9-3-3-3/ 10-4-4-4	9-2-2-2/ 9-3-3-3/ 10-4-4-4	*1
Read Page Miss	EDO	13-2-2-2/ 14-3-3-3	12-2-2-2/ 13-3-3-3	*3
	FP	12-2-2-2/ 13-3-3-3/ 14-4-4-4	12-2-2-2/ 12-3-3-3/ 13-4-4-4	*1 *3
Post Write	EDO	4-1-1-1	4-1-1-1	
	FP	4-1-1-1	4-1-1-1	
Write Retire Rate (Buffer to DRAM)	EDO	2/3	2/3	
	FP	3/4	3/4	*1

*1 X-2-2-2 is for Interleave mode, X-3-3-3 is for CAS active 2 CPU clocks.
*2 It is for RAS to CAS time of 3 CPU clocks.

*3 It is for RAS pre-charge time of 4 CPU clocks, RAS to CAS time of 3 CPU clocks.
There is a one level built-in CPU to Memory post-write buffer with 4 Quad Word deep (CTMFF). All the write access to DRAM will be buffered. For the CPU read miss / Line fill cycle, the write-back data from the second level cache will be buffered first, and the PCMC will start to read data from DRAM at the same time. The buffered data are written to DRAM right after the read cycle. With this concurrent write back policy, many wait states are eliminated. However, any other cycle targeting DRAM will be pending until the CTMFF is empty.

For the read access, there will be either single or burst read cycle to access the DRAM which depends on the cacheability of the cycle. If the current DRAM configuration is half-populated bank, then the SiS5511 will assert 8 consecutive cycles to access DRAM for the burst cycle. For the single cycle that only accesses DRAM within a DWord, the 5511 will only issue one cycle to access DRAM. For the single cycle that accesses one Qword or cross DWord boundary, the 5511 will issue two consecutive cycles to access DRAM.

Refresh cycle

The refresh cycle will occur every 15.6 us. It is timed by a counter of 14Mhz input. The CASE[7:0]# and CASO[7:0]# will be asserted at the same time. The RAS[3:0]# are asserted sequentially. In order to reduce the impact of performance, the "Intelligent Refresh" will only refresh those populated banks.

Characteristics of Shadow RAM

The SiS5511 defines the characteristics of any 16K memory block between 640 KByte to 1 MByte address range through register 80h to 86h. Through these registers, the memory blocks can be programmed not only to be directly accessible by the CPU or PCI Bus Master (combined with another enable bit for PCI Master accessible), but also their cacheability attributes. There are three bits: Read Enable, Write Enable, and Cache Enable, in each registers to define the corresponding memory blocks as normal read/write DRAM function, these bits also specify the cacheability of these blocks to the first/second level cache.

Table 2-11 shows the attributes of these enable bits, Table 2-12 is the attribute bits assignments and the attribute definitions, and Table 10 represents the registers and their corresponding memory segments.

Table 2-11 Attributes of enable bits

Read Enable	When this bit is set to 1, the CPU read cycles that access to the corresponding memory block are regarded as normal DRAM read cycles. Otherwise, the read cycles are directed to the PCI bus.
Write Enable	When this bit is set to 1, the CPU write cycles that access to the corresponding memory block are regarded as normal DRAM write cycles. Otherwise, the write cycles are directed to the PCI bus.

Cache Enable	When this bit is set to 1, the corresponding memory block is programmed to be L1/L2 cacheable. Note that the cacheable function is for code portion only, and the cacheability works only if Read Enable bit is also enabled.

Table 2-12: Attribute Bit Assignments and Attribute Definitions

Read Enable	Cache Enable	Write Enable	Attribute	Definition
0	0	0	Disable	Cycles are transferred to PCI bus.
0	0	1	Write Only	Write cycles are conducted to DRAM in normal manners, and read cycles are passed to PCI bus for termination.
0	1	0	Disable	Cycles are transferred to PCI bus.
0	1	1	Write Only	Write cycles are conducted to DRAM in normal manners, and read cycles are passed to PCI bus for termination.
1	0	0	Read Only	Read cycles are conducted to DRAM in normal manners, and write cycles are passed to PCI bus for termination.
1	0	1	Read/Write	Normal DRAMRead/Write cycles.
1	1	0	Read/Cacheable	Normal DRAM read cycles and code portion is cacheable to L1/L2.
1	1	1	Read/Write/ Cacheable	Normal DRAM Read/Write cycles and code portion is cacheable to L1/L2.

NOTE: When PCI master access enable bit is set, the PCI master Read/Write cycles are served as the same as the descriptions in Table 12.

Table 2-13 Registers and Corresponding Memory Blocks

Reg	Bits	Attribute				Memory Block
80h	7:4	Read Enable	Cache Enable	Write Enable	Reserved	0c0000-0c3fffh
	3:0	Read Enable	Cache Enable	Write Enable	Reserved	0c4000-0c7fffh
81h	7:4	Read Enable	Cache Enable	Write Enable	Reserved	0c8000-0cbfffh
	3:0	Read Enable	Cache Enable	Write Enable	Reserved	0cc000-0cffffh
82h	7:4	Read Enable	Cache Enable	Write Enable	Reserved	0d0000-0d3fffh
	3:0	Read Enable	Cache Enable	Write Enable	Reserved	0d4000-0d7fffh
83h	7:4	Read Enable	Cache Enable	Write Enable	Reserved	0d8000-0dbfffh
	3:0	Read Enable	Cache Enable	Write Enable	Reserved	0dc000-0dffffh
84h	7:4	Read Enable	Cache Enable	Write Enable	Reserved	0e0000-0e3fffh
	3:0	Read Enable	Cache Enable	Write Enable	Reserved	0e4000-0e7fffh
85h	7:4	Read Enable	Cache Enable	Write Enable	Reserved	0e8000-0ebfffh

	3:0	Read Enable	Cache Enable	Write Enable	Reserved	0ec000-0effffh
86h	7:4	Read Enable	Cache Enable	Write Enable	Reserved	0f0000-0fffffh

SMRAM Area Re-mapping

The SMRAM area is defined in E0000h to E7FFFh. This area can be re-mapped to A0000h to A7FFFh or B0000h to B7FFFh.

Others

It is supported to assert the RAMW# at the end of each memory read cycle when EDO DRAM is accessed. When the power saving mode is enabled, the RAMW# pulse will be 1.5 CPU clock at least to reduce the power consumption.

The DRAM always-page-miss mode is also supported. Once it is programmed, the DRAM cycle will be a page-start cycle.

The CAS current can be programmed as 8 mA or 4 mA by register 5Dh bit 3 and 4.

2.2.4 PCI Arbiter

The SiS5511 contains a high performance hidden arbitration scheme that allows efficient bus sharing among five PCI Masters and the CPU. Note that one PCI master is reserved for the PSIO chip.

The SiS5511 employs the priority rotation scheme that is done at two different layers. The first layer is shared between PSIO and four PCI Masters as a group. The second layer consists of four PCI masters with equal priority. Arbitration is done at both layers. The winner of arbitration among the four PCI masters arbitrates the PCI bus against PSIO. Fair rotation scheme applies only at layer level. The arbitration scheme assures that ISA master or DMA channels (represented by PSIO) can access the bus with short bus latency required by the traditional ISA masters or DMA devices. This implementation together with PCI Programmable Bursting Address Counter guarantees ISA device will not be starved during PCI master long bursting cycle. For example, when the maximum bursting length is 512 bytes, the maximum arbitration latency for PSIO, and PCI master is about 12us, and 40us respectively. The following two figures detail the rotation arbitration structure and its corresponding timing diagram.

Rotation Arbitration Scheme:

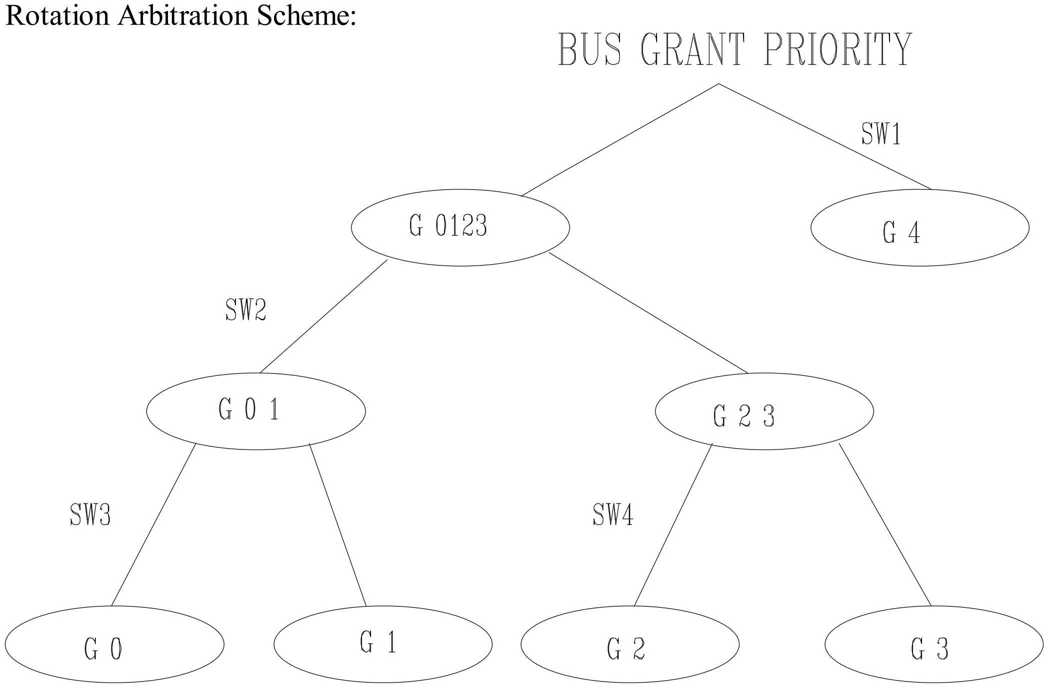


Figure 2.4

Notation:
SW1:is the switch for path from node G4 or G0123 to BUS GRANT PRIORITY
SW2:is the switch for path from node G01 or G23 to node G0123
SW3:is the switch for path from node G0 or G1 to node G01
SW4:is the switch for path from node G2 or G3 to node G23
G01,G23,G0123:are intermediate nodes
G4:is the bus request from PSIO
G0,G1,G2,G3:are the bus requests from PCI device 0, device 1, device 2, device 3 respectively.

Initial Path Parking:
SW1: BUS GRANT PRIORITY-G4
SW2: G0123-G01
SW3: G01-G0
SW4: G23-G2

Rule of Rotating Priority for Bus Arbitration:

- BUS GRANT PRIORITY will choose a path whenever it encounters an optional path.
- PCI bus will be granted as Daisy Chain
- Path switches will be toggled from BUS GRANT PRIORITY to any request node(G4,G0,G1,G2,G3) if any of them have been utilized

Example:
Initial Priority:G4,G01,G0,G2

1.PSIO(G4) Request Bus
PHLDA# is asserted
SW1 is toggled to G0123 (since it has been utilized)
Priority change to G0,G1,G2,G3,G4

2.PSIO,REQ3,REQ2,REQ1,REQ0 are requesting bus
GNT0# is asserted
SW1, SW2 and SW3 are toggled to G4, G23 and G1 respectively (since they have been utilized)
Priority change to G4,G2,G3,G1,G0

3.REQ3,REQ2,REQ1,REQ0 are active
GNT2# is asserted
SW2 and SW4 are toggled to G01 and G3 respectively(since they have been utilized)
Priority change to G4,G1,G0,G3,G2

4.REQ3,REQ2,REQ1,REQ0 are active
GNT1# is asserted
SW2 and SW3 are toggled to G23 and G0 respectively(since they have been utilized)
Priority change to G4,G3,G2,G0,G1

5.REQ3,REQ2,REQ1,REQ0 are active
GNT3# is asserted
SW2 and SW4 are toggled to G01 and G2 respectively(since they have been utilized)
Priority change to G4,G0,G1,G2,G3

6. During 3-5 if there is a request comes from PSIO, the Arbiter will grant bus to PSIO

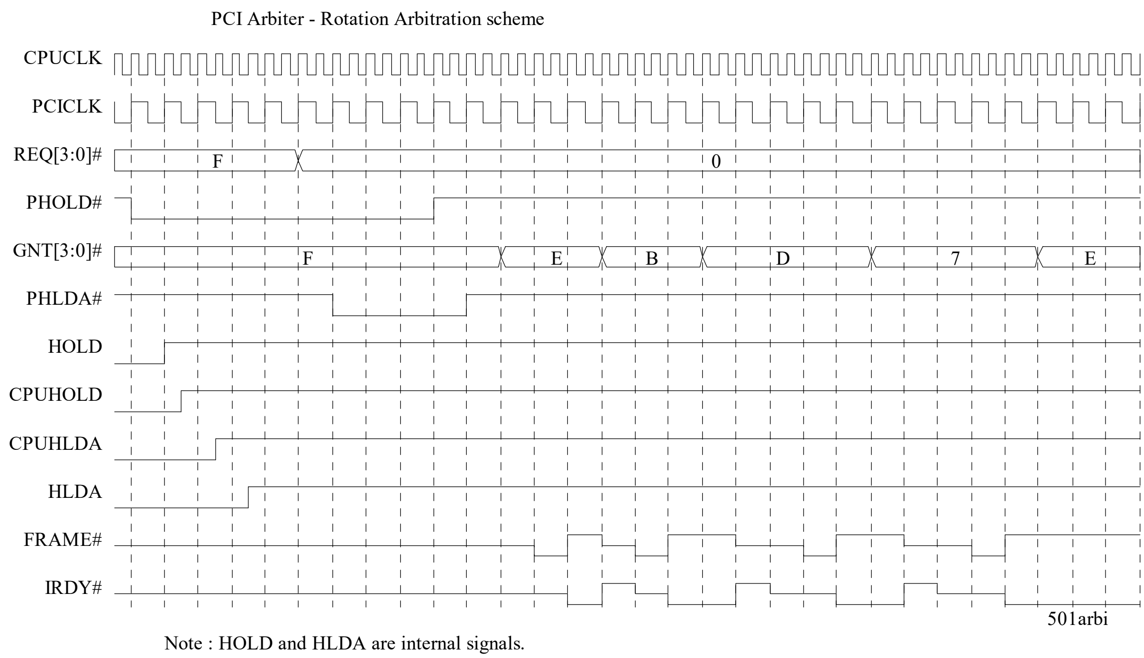


Figure 2.5

A PCI master can burst so long as the PCI target can source/sink the data, and no other agent requests the bus. However, PCI specifies two mechanisms that cap a master's tenure in the presence of other requests, so that predictable bus acquisition latency can be achieved. One is the Master Latency Timer(LT) that is not implemented into the PCMC, the other is the Target Initiated Termination. In the SiS5511, a programmable Bursting Address Counter(PBAC) is implemented to disconnect the PCI master during the long bursting cycle. In this way, high throughput is maintained, and the bus latency is still kept reasonably small. Note that the bursting length is naturally applied to PCI master to local memory accessing. When PCI master access non-local memory target, both the master and target should have the responsibility of maintaining reasonable latency.

The PCI arbiter asserts only one GNT# at any time. The 5511 has also implemented a time-out counter to prevent faulty device hugging the bus. If the PCI bus is granted to a PCI device and the bus is currently idle, 16 PCI clocks is the limitation that device should assert FRAME# during the period of time. If time-out occurs, the arbiter will mask request line, therefore deasserts GNT#. When this happens, all PCI devices start arbitration again. Note that PSIO is free to this constraint.

The 5511 will release the host bus to CPU when PCI master is not targeting to main memory. The arbiter will keep the GNT# to that PCI master until the PCI bus is idle even when other PCI master has asserted REQ# to 5511.

2.2.5 PCI Bridge

PCI Master Controller

The PCI Master Controller forwards the CPU cycles not targeting the local memory to the PCI bus. In the case of a 64-bit CPU request or a misaligned 32-bit CPU request, the PCMC assumes the read assembly and write disassembly control. A 4 level posted write buffer(CTPFF) is implemented to improve the CPU to PCI memory write performance. Except for on-board memory write cycles, any cycles forwarded to the PCI bus will be suspended until the CTPFF is empty. For PCI bus memory write cycles, the CPU data are pushed into the CTPFF if it is not full. The push rate for a DW is 3 CPUCLKs.The pushed data are, at later time, written to the PCI bus. If the consecutive written data are in DW incremental sequence, they will be transferred to the PCI bus in a burst manner.

The PCMC provides a mechanism for converting standard I/O cycles on the CPU bus to Configuration cycles on the PCI bus. Configuration Mechanism#1 in PCI Specification 2.0 page 61 is used to do the cycle conversion.

The PCMC always intercepts the first interrupt acknowledge cycle from CPU bus, and forwards the second interrupt acknowledge cycle onto the PCI bus.

The PCMC supports one level prefetch buffer for IDE data port to enhance the data transfer rate of CPU read the IDE. Usually, it takes 12~14 CPUCLKs for CPU read IDE cycle. If the prefetch function is turned on, it only needs 7 CPUCLKs for the consecutive IDE read cycle, about 2X faster than the former.

The general timing required for CPU read from/write to PCI bus is shown in the following table.

Table 2-14

CPU forwards to PCI cycle	CPUCLK=50/60/66MHz

CPU read	12~14
CPU write (nonposted)	14~16
CPU read prefetch buffer	7
CPU posted write	3

PCI Slave Controller

The SiS5511 operates as a slave on the PCI bus whenever a PCI master requests an access to the SiS5511 resource such as Cache, DRAM and the SiS5511 internal registers. Note that the internal registers can only be accessed by the SiS5511 itself when in CPU cycle. In the SiS551X PCI/ISA system, the CPU is placed in HOLD state before granting the PCI bus to a PCI master. The following figure shows the behavior of CPUHOLD/CPUHLDA in response to PCI masters requests. Only linear ordered PCI cycles are supported by the PCMC PCI slave interface.

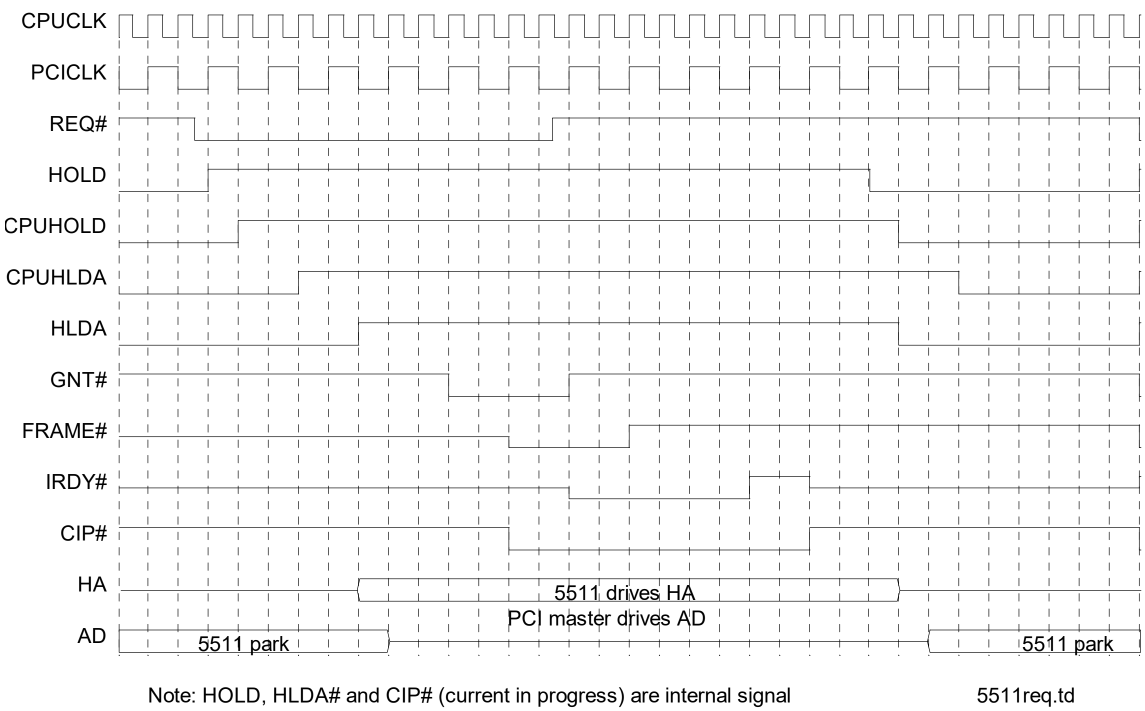


Figure 2.6

A PCI master to the local memory access is not conducted until the snoop cycle has completed. The snoop cycle is used to inquire the first level cache to maintain coherency between first level and second level caches and main memory. Snoop cycles are performed by driving the PCI master address onto the CPU bus and asserting EADS#. Depending on the status of HITM# two clocks after the assertion of EADS#, PCMC conducts the PCI master cycles as table 14 outlines.

Table 2-15

PCI Master Read Cycle		
L1	L2	Data Transfer
Miss (or Unmodified)	Miss	Data transfer from DRAM to PCI
Miss (or Unmodified)	Hit (Dirty or !Dirty)(*1)	Data transfer from L2 to PCI
HitM	Miss	Data is first written back from L1 to DRAM. Then, PCI master gets data from DRAM.(*3)
HitM	Hit (Dirty or !Dirty)(*1)	Data is first written back from L1 to L2. Then, PCI master gets data from L2. The line is marked dirty in the L2.(*3)
PCI Master write Cycle		
L1	L2	Data Transfer
Miss (or Unmodified)	Miss	Data transfer from PCI to DRAM
Miss (or Unmodified)	Hit (Dirty or !Dirty)(*2)	Data transfer from PCI to DRAM and L2. The Dirty bit is not changed.
HitM	Miss	Data is first written back from L1 to DRAM. Then, PCI master writes data to DRAM.(*3)
HitM	Hit (Dirty or !Dirty)(*2)	Data is first written back from L1 to L2. Then, PCI master writes data to L2 and DRAM. The Line is marked dirty in the L2. (*3)

NOTE:
(*1) For burst or pipeline SRAM, the rule is changed as it is described below. If L2 is in WT mode, data transfer is always from DRAM to PCI side. If L2 is in WB mode, data transfer is from DRAM to PCI side if the line is not dirty. If the line is dirty, data transfer is from L2 to PCI side, and PCI transfer is disconnected after the completion of reading this line.

(*2) The rule is changed when burst or pipeline SRAM is used. No matter that the line may be dirty or not, data transfer conducts from PCI to L2 side, and PCI transfer is disconnected after the completion of writing this line.

(*3) This case is only applied to the initial line(line 0). The PCI transfer will be disconnected after the completion of line n if line n+1 is a Modified one in L1, where n>=0. The snooping write back cycle will be deferred until line n is completely transferred.

In the SiS551X, the INV signal of the CPU should be connected to W/R# that is driven by the 5511 in the PCI master cycle. In this way, the 5511 can invalidate the line that is currently inquired via the assertion of EADS# in the PCI master write cycles.

The PCMC slave interface supports PCI burst transfers, the bursting length can be 256 bytes, 512 bytes, 1K bytes, 2K bytes, or 4K bytes. A burst transfer will be disconnected (retry) if the transfer goes across the bursting length. In this way, at most 128 cache lines can be uninterruptedly transferred if they are in I, S, or E state in the L1 cache. Another reason for the constraint is that page miss may occur only once during the entire bursting transaction since the maximum bursting length is always within the page size in any of the used DRAM .

There is a 4QW deep FIFO to prefetch data when PCI master reads from the local memory. To achieve the utmost data transfer speed, the 5511 implements an advanced prefetch algorithm and snoop ahead function. It causes the PCI burst transfer performed in the pace of X-1-1-1.... 5511 always prefetches one or two QW from L2/DRAM in advance to the asserting of TRDY#. This can be programmed by writing bit 2 of configuration register 5Bh. The snoop ahead mechanism ensures the acquiring of the hit modify status of the next prefetching line(line n+1) before the prefetching of line n is completed. If n+1 is not a Modified line in L1, prefetching of n+1 can be conducted right after the completion of prefetching line l. In such a case, 5511 keeps piping data into the FIFO in L2/DRAM side, and it also keeps piping data out of the FIFO in the PCI side in 0 wait state. If n+1 is a Modified line in L1, 5512 will issue STOP# to disconnect the burst transfer after line n being consumed. This function also performs on PCI master write cycle. The PCI master writes are buffered in the 4 QW deep PCI to memory posted write buffer(PTHFF). The PCMC always posted an aligned QW PCI write data into the write buffer and then retires it into the DRAM array or the L2 cache. The PCI write performance is X-1-1-1.

The PCI bus data transfer rate can be calculated from the following formula.

DATA TRANSFER RATE = NB/{ X + (W + 1) * [(NB / 4) -1] } * (1 / f)

where

- NB: Total number of bytes Transferred or Bursting Length which is defined in bit 6-4 of configuration register 5Bh.
- X: number of PCI clocks for the first data transfer or leadoff cycle time.
- W: number of wait state for PCI burst transfer
- F: frequency of PCI clock

Since 5511 PCI bridge is designed as asynchronous to CPU clock, the PCI clock is always running at 33MHz to gain the fast transfer rate.

The leadoff cycle is in general determined by: 1) the relative clock phase between CPUCLK and PCICLK, and 2) L1 cache policy. Specifically, in the PCI master read cycle, the leadoff cycle is determined by the logic of bit 2 of register 5Bh. Morever, whether the initial line hits L2 or whether it is a page hit or miss cycle also affects the leadoff cycle time. It is estimated that the leadoff cycle is 4 to 5 PCICLKs and 6 to 10 PCICLKs for PCI master write and read cycle, respectively. If the initial line hits a modified line in L1, ten more PCICLKs is required for the leadoff cycle. The following table illustrates the PCI Master performance in different Bursting length when the leadoff cycle is 5 and 7 for write and read, respectively.

Table 2-16

Data Transfer Rate		
PCI master cycle	7-1-1-1... read	5-1-1-1... write
Bursting length		
512 bytes	127MB/s	129MB/s
1K bytes	130MB/s	131MB/s
2K bytes	131MB/s	132MB/s
4K bytes	132MB/s	133MB/s

An important factor in the sustaining of 0 wait PCI transferring is the prefetching and retiring rate that the system controller can perform. The following table outlines the rates that 5511 can keep. The rate is numbered in terms of CPUCLK per Qw. The prefetching rate from FP can be 2-2-2-...if DRAM system is populated in the interleaved manner. For 32-bit DRAM organization, it takes twice the parameters cited below.

Table 2-17

	EDO	FP	ASRAM	PBSRAM
Prefetching Rate	2/3	2/3	2	2
Retiring Rate	2/3	3	2	2

Concurrent refresh will still be performed when CPU is put into Hold state. If the DRAM is idle, refresh can be conducted at any time. If refresh request occurs at the same time that a PCI master wants to access DRAM, an arbitration scheme is employed to resolve the conflict. The refresh request may thus get service while the PCI master accessing is suspended until refresh cycle is completed. Although refresh may win the DRAM bus, at most one refresh cycle may be conducted for each individual PCI transaction, i.e. for each FRAME# initiating. On the other hand, refresh may be also deferred until the DRAM is idle. In SiS551x system, the refresh may be postponed for no more than 33 us in the worst case when a PCI master is reading the whole 128 lines through one burst transaction.

2.2.6 Green PC Function

The following paragraphs are the PMU (Power Management Unit) features description:

Power States

The PMU provides different power management states, which are described in the following sections.

(i) Monitor Standby State

The Monitor will be blanked and the external devices are turned off through SMOUT when the Monitor standby timer expires.

Monitor Standby monitors the following events:

IRQ 1-15
HOLD
NMI

Each IRQ has two sets of mask bits, one for wake up mask, and the other for standby mask. The HOLD includes the PCI local masters and the ISA master request. Each event is maskable. If no event happens during the monitored period and the timer expires, an SMI is generated and the monitor enters the standby state.

Once the Monitor is in the standby state, any event from IRQ1-15, NMI or HOLD will cause an SMI which brings the Monitor back to the normal state.

The time slot of the Monitor standby timer is programmable to 6.6sec, 0.84sec, 13.3ms, 1.6ms.

(ii) System Standby State

If the system standby timer expires, an SMI is generated for the system to enter the system standby state. The following events happen:

STPCLK# is asserted to stop the CPU clock
The hard disk drives spindle motors can be turned off
The serial, parallel ports or the programmable I/O port can be turned off

Once the STPCLK# is asserted, any events from IRQ1-15, NMI, HOLD, INIT will cause the STPCLK# be de-asserted. If any of the Hard disk motors, serial, parallel or programmable I/O ports were turned off, they will be back to the normal state only when they are accessed.

System Standby monitored events (each event is maskable)
Programmable I/O ports (one is a 10-bit I/O port, another is a16-bit I/O port)
IRQ 1-15 (each has 2 sets of mask bits as for Monitor Standby State)
HOLD
NMI
Hard Disk ports (1F0-1F7h, 3F6-3F7h, 170-17Fh, 320-32Fh)
Serial ports (2F8-2FFh, 3F8-3FFh, 2E8-2EFh, 3E8-3EFh)
Parallel ports (278-27Fh, 378-37Fh, 3BC-3BEh)
A0000-AFFFFh or B0000-BFFFFh Address trap (Video RAM)
C0000-C7FFFh Address trap (Video BIOS)
3Bx-3Dxh (Video I/O port)

The time slot of the System standby timer is programmable to 9 sec, 1.1 sec, 70ms, and 8.85ms.

(iii) Throttling state

In throttling state, STPCLK# is asserted and de-asserted periodically. This function is maskable. The throttling timer (Registers 61h and 62h) is programmable and the time slot is 35us.

Break Switch SMI

Whenever the break switch is pressed, it caused an SMI to enter or leave power saving state. The signal from the break switch is a level trigger signal which lasts for more than 3 CPU clocks.

Software SMI

If the software SMI enable bit is set and a '1' is written to bit 1 of Register 60h, an SMI# is generated and the software SMI service routine is invoked. The bit 1 of Register 60h should be cleared at the end of the SMI handler.

Shadow Register

In order to support "suspend to HDD" function, all necessary shadow registers are implemented into 5513. For more detailed information, please refer to "5513 Register Description".

2.3 Register Description

There are three types of registers in the PCMC, I/O mapped registers, PCI configuration space registers, and SLiC memory mapped registers.

2.3.1 I/O Mapped Registers

The 5511 uses PCI configuration space access mechanism #1. This mechanism defines two registers, CONFIG_ADDRESS(CF8h) register and CONFIG_DATA(CFCh) register. Both CONFIG_ADDRESS and CONFIG_DATA are read/write registers, and the length is DWORD. The mechanism is to write a value into CONFIG_ADDRESS first, then read or write to CONFIG_DATA. The write to CONFIG_ADDRESS specifies the PCI bus, device on that bus, and the configuration register in that device being accessed. The read or write to CONFIG_DATA will cause the host bridge to translate the CONFIG_ADDRESS value to the requested configuration cycle.

The definition of CONFIG_ADDRESS register is described below:

Register 0CF8h CONFIG_ADDRESS Register

31	30 24	23 16	15 11	10 8	7 2	1	0
	Reserved	Bus Number	Device Number	Function Number	Register Number	0	0

Enable bit ('1' = enabled, '0' = disabled)

Bit 31 **is an enable flag for determining if the accesses to CONFIG_DATA should be translated to configuration cycles on the PCI bus.**

Bits 30:24 Reserved, read only, and must return 0's when read.

Bits 23:16 Choose a specific PCI bus in the system.

Bits 15:11 Choose a specific device on the bus.

Bits 10:8 Choose a specific function in a device.

Bits 7:2 Choose a DWORD in the device's configuration space.

Bits 1:0 Read only and must return 0's when read.

A full Dword I/O write to address 0CF8h, the host bridge will load the data into CONFIG_ADDRESS register. Also, a full DWord I/O read to 0CF8h, the host bridge gets the data from CONFIG_ADDRESS register. Any non-Dword writes or reads to 0CF8h are treated as normal PCI I/O cycles. When the host bridge of 5511 sees an I/O access that falls inside the Dword beginning at CONFIG_DATA address, it checks the enable bit of the CONFIG_ADDRESS register. If bit 31 of CONFIG_ADDRESS register is 1, the I/O cycle is translated into a configuration cycle.

There are two types of configuration cycle determined by bus number. If the Bus Number is zero, the configuration cycle will be Type 0. If the Bus Number is non-zero, the configuration cycle will be Type 1.

For type 0 configuration cycle, AD[1:0] is driven to "00" during the address phase of the cycle. The host bridge decodes the device number of CONFIG_ADDRESS to assert only one "1" on the AD[31:11] and copies bits[10:2] of CONFIG_ADDRESS to AD[10:2] directly. For instance, when accessing the configuration registers of 5511, because 5511 is considered device 0 on bus 0, AD11 will be high, and bits[10:2] of CONFIG_ADDRESS are copied to AD[10:2] directly. Never use AD11 as the IDSEL line for any other PCI target device since it is reserved for PCMC. The 5511 responds to configuration by asserting DEVSEL#.

For type 1 configuration cycle, AD[1:0] is driven to "01" and bits[31:2] of CONFIG_ADDRESS are copied to AD[31:2] directly during the address phase of the cycle.

The byte-enables for the data phase of both types 0 and type 1 configuration cycles are copied from the HBE[7:4]# directly.

The following programming sequences is an example of writing register 51h in PCMC and of reading register 5Ch, 5Dh, 5Eh and 5Fh in PCMC.

```
write 51h:
mov         eax, 80000050h
out         0cf8h, eax
mov         al, DATA
out         0cfdh, al

read 5Ch, 5Dh, 5Eh and 5Fh:
MOV       EAX, 8000005Ch
OUT       0CF8h, EAX
IN        0CFCh
```

Register 0CF9h   Turbo and Reset Control Register .

Bits 7:5 Reserved

Bit 4 INIT Enable

When this bit is set to 1 ,the PCMC drives INIT during software reset. When this bit is cleared to 0, the PCMC drives CPURST during software reset, and INIT is inactive.

Bit 3 BIST

When this bit is set to 1 and bit 4 as well as bit 1 are enabled, a subsequent initiation of the CPU hard reset through bit 2 of this register enables the Built In

Self Test(BIST) mode of the CPU. The PCMC also drives the INIT during the hard reset.

Bit 2 Reset CPU.

There are two types of resets to the CPU: a hard reset using the CPURST signal and a soft reset using the INIT signal. If bit 1 of this register is set to 1 and bit 2 transitions from 0 to 1, the PCMC initiates a hard reset. A hard reset through this register thus requires two write operations to this register: the first write operation writes a 1 to bit 1 and a 0 to bit 2. The second write operation writes a 1 to bit 1 and a 1 to bit 2. When bit 1 of this register is 0 and bit 2 transitions from 0 to 1, the PCMC initiates a soft reset. The sequence to initiate a soft reset through this register is identical to that of a hard reset except a 0 is written to bit 1 in the first write operation.

Bit 1 Enable System Hard Reset.

When this bit is set to 1 and bit 2 transitions from 0 to 1, the PCMC initiates a hard reset to the CPU . When this bit is 0 and bit 2 transitions from 0 to 1, the PCMC initiates a soft reset to the CPU.

Bit 0 Reserved

2.3.2 PCI Configuration Space Mapped Registers

Register 00h Vendor ID - low byte

Bits 7:0 39h

Register 01h Vendor ID - high byte

Bits 7:0 10h

Register 02h Device ID - low byte

Bits 7:0 11h

Register 03h Device ID - high byte

Bits 7:0 55h

Register 04h Command - low byte

Bits 7:0 07h

Register 05h Command - high byte

Bits 7:0 00h

Register 06h Status - Low Byte

Bits 7:0 00h

Register 07h Status - High Byte (default = 02h)

Bit 7 **Detected Parity Error.**

This bit is always 0 since the PCMC does not support parity checking on the PCI bus.

Bit 6 **Reserved**

Bit 5 **Received Master Abort.**

This bit is set by the PCMC whenever it terminates a transaction with master abort. This bit is cleared by writing a 1 to it.

Bit 4 **Received Target Abort.**

This bit is set by the PCMC whenever it terminates a transaction with target abort. This bit is cleared by writing a 1 to it.

Bit 3 **Signaled Target Abort.**

This bit is always 0 since the PCMC will not terminate a transaction with target abort.

Bits 2:1 **DEVSEL# Timing DEVT.**

The two bits define the timing to assert DEVSEL#. The PCMC asserts the DEVSEL# signal within three clocks after the assertion of FRAME#. The default value is DEVT=10. In fact, the PCMC always asserts DEVSEL# in medium timing except in CPU writes to I/O port 64h or 60h.

Bit 0 **Reserved**

Register 08h Revision Identification.

Bits 7:0 **00h**

Register 09h~0Bh Class ID

Bits 23:0 **060000h**

Register 0Eh Header Type

Bits 7:0 **00 (Read Only)**

Register 50h (default = 00h)

Bit 7 **L2 Cache Exist or not**

0: There is no L2 cache

1: L2 cache exists

When this bit is set to 0, no cache cycle occurs. This bit is programmed by BIOS before it processes the cache auto-sizing function, and is disabled after the function has been finished.

Bit 6 L2 Cache Enable

0: Disable

1: Enable

When this bit is disabled, all the cache cycles will be cache miss cycles. We can not read data from cache, but the data in cache can still be updated whenever update cycles occur. This makes the data in the L2 cache is always coherent with the data in DRAM. Then, whenever we enable L2 cache, we can get the correct data from cache immediately. This bit is programmed by BIOS when L2 is auto-detected and initialized.

Bits 5:4 SRAM Type

00 : Asynchronous SRAM

01 : Burst SRAM

10 : Pipeline Burst SRAM

11 : Reserved

Bit 3 L2 Cache WT/WB Policy

0 : Write Through Mode

1 : Write Back Mode

Bits 2:1 L2 Cache Size

00: 64 K Byte

01: 256 K Byte

10: 512 K Byte

11: 1M Byte

Bit 0 CPU L1 Cache Write Back Mode Enable

0: Disable

1: Enable

Register 51h (default = 00h)

Bits 7:6 Asynchronous SRAM Leadoff Timing

	Read Cycle	**Write Cycle**	
00	4	4	CPUCLK
01	3	3	CPUCLK
10	3	4	CPUCLK
11	Reserved		

Bit 5 **Asynchronous SRAM burst Timing**

0: 2 CPUCLK

1: 3 CPUCLK

Bit 4 **Synchronous SRAM Leadoff Timing**

0: 3 CPUCLK

1: 4 CPUCLK

Bit 3 **Cache burst Addressing Support**

0: Toggle mode

1: Linear mode

Bit 2 **Cache Tag size Selection**

0: 7 bits

1: 8 bits

Bit 1 **Cache Sizing Enable**

0: Normal Operation

1: Always Cache hit regardless the input Tag address

By setting this bit, BIOS can implement L2 cache auto-detection and sizing operation.

Bit 0 **Reserved**

Register 52h (default = 20h)

Bit 7 **Reserved**

Bit 6 **Reserved**

This bit should be programmed to 1.

Bit 5 **FP DRAM CAS Precharge Time**

0: 2 CPUCLK

1: 1 CPUCLK

Bits 4:3 **EDO Cycle CAS Pulse Width, when non-interleave**

	Read	Write	
00	2	2	CPUCLK
01	1	1	CPUCLK
10	1	2	CPUCLK
11	Reserved		

Bit 2 **EDO CAS Precharge Time**

0: 1 CPUCLK

1: 2 CPUCLK

Bit 1 MDLE Timing when EDO DRAM Is Read

0: 1 CPUCLK delay from CAS pulse

1: 1.5 CPUCLK delay from CAS pulse

Bit 0 BRDY# Timing when EDO DRAM Is Read

0: 1 CPUCLK delay from CAS pulse

1: 2 CPUCLK delay from CAS pulse

Register 53h (default = 00h)

Bits 7:6 DRAM RAS to CAS Delay Timing

00: 4 CPUCLK

01: 3 CPUCLK

10: 2 CPUCLK

11: Reserved

Bits 5:4 DRAM RAS Precharge Timing for FP DRAM

00: 5 CPUCLK

01: 4 CPUCLK

10: 3 CPUCLK

11: Reserved

Bits 3:2 RAS Active When Refresh

00: 6 CPUCLK

01: 5 CPUCLK

10: 4 CPUCLK

11: Reserved

Bit 1 RAS Precharge Timing for EDO DRAM

0: 3 CPUCLK

1: 4 CPUCLK

Bit 0 CAS Output Delay from push HD into CTMFF in DRAM Posted-Write Cycle

0: 1 CPUCLK

1: 2 CPUCLK

Register 54h (default = 00h)

Bits 7:6 DRAM Bank-Interleave Mode

00: Non-interleave

01: Bank 0 and Bank 1 only

10: Bank 2 and Bank 3 only

11: Bank 0, 1 and Bank 2, 3

Bit 5 **Always Page Miss Mode in DRAM Read Cycle**

0: Normal mode

1: Always page miss

Bit 4 **RAMW# Power Saving Mode When EDO Bank Is Being Accessed**

0: Normal mode

1: Power saving mode

Bit 3 **NA# Disable**

0: Enable

1: Disable

Bit 2 **EDO Test Mode**

0: Normal Mode

1: Test Mode

When set, 5511 will delay the assertion of BRDY# by 15us after the negation of CAS#. The EDO test procedure is summerized below:

1. Enable this bit.

2. Write data to DRAM and then read it.

3. If the read data is the same as the write data, EDO type DRAM is used.

Bit 1 **Pipelined Burst SRAM / Burst SRAM Test Mode**

0: Normal Mode

1: Test Mode

Like EDO test mode, BIOS writes a data into L2 cache, and then reads the data from cache at the same address. If BIOS can read the right data from cache by the end of T2, Burst SRAMs are detected, otherwise L2 cache is not Burst SRAM type. The test-mode bit is set by BIOS before L2 cache auto-detection process. During the test mode, all the memory cycles are treated as L2 hit, and BRDY# is always asserted at the clock immediately after the asserted ADS#.

Bit 0 **Reserved**

This bit should be programmed to 1.

Register 55h (default = 00h)

Bit 7 **Slow Refresh Enable (1:4)**

0: Disable

1: Enable

Bit 6 **Deturbo Enable**

0: Disable

1: Enable

Bit 5 **FLUSH# & INTR1 Selection Bit**

0: FLUSH#

1: INTR1

Bit 4 **CPUID & Turbo Selection Bit**

0: CPUID

1: Turbo

Bit 3 **Selection of Current Rating of RAS[3:0]#**

0: 8 mA

1: 12 mA

Bit 2 **Reserved**

This bit should always be programmed as 1 to support more than 128M Bytes.

Bit 1 **BEDO write pulse delay 1T after read BEDO**

1: Enable

0: Disable

Bit 0 **BEDO MDLE 0.5 T pulse width generated type**

1 for falling edge of CPUCLK

0 for rising edge of CPUCLK

Register 56h (default = 00h)

Bit 7 **Allocation of Non-Cacheable Area I**

0: Local DRAM

1: PCI Bus. The local DRAM is disabled.

Bit 6 **Non-Cacheable Area I Enable**

0: Disable

1: Enable

Bits 5:3 **Size of Non-Cacheable Area I (within 128 MBytes)**

000	64KB	100	1MB

001	128KB	101	2MB
010	256KB	110	4MB
011	512KB	111	8MB

Bits 2:0 A26~A24 of Non-Cacheable Area I (within 128 MBytes)

Register 57h (default = 00h)

Bits 7:0 A23~A16 of Non-Cacheable Area I (within 128 MBytes)

Register 58h (default = 00h)

Bit 7 Allocation of Non-Cacheable Area II

0: Local DRAM

1: PCI Bus. The local DRAM is disabled.

Bit 6 Non-Cacheable Area II Enable

0: Disable

1: Enable

Bits 5:3 Size of Non-Cacheable Area II

000	64KB	100	1MB
001	128KB	101	2MB
010	256KB	110	4MB
011	512KB	111	8MB

Bits 2:0 A26~A24 of Non-Cacheable Area II (within 128 MBytes)

Register 59h (default = 00h)

Bits 7:0 A23~A16 of Non-Cacheable Area II (within 128 MBytes)

Register 5Ah (default = 00h)

Bit 7 Fast Gate A20 Emulation Enable

0 : Disable

1 : Enable

The sequence to generate A20M# is: write D1h to I/O port 64h followed by I/O write to port 60h with data 00h. When this bit is enabled, the SiS5511 responds the cycle by asserting DEVSEL# in slowest timing. Otherwise, the cycle is subtractively decoded by SiS 5513, and then is passed to 8042 on the ISA bus.

Bit 6 Fast Reset Emulation Enable

0 : Disable

1 : Enable

The Fast reset command is I/O write to port 64h with data 1111XXX0b.

After the command is issued, the assertion of INIT or CPURST is delayed by 2us or 6us which can be programmed in bit 5, and is held for 25 CPUCLK.

Bit 5 Fast Reset Latency Control

0 : 2us

1 : 6us

Bit 4 Enable IDE Prefetching Function

0: Disable

1: Enable

Bit 3 CPU-to-PCI Post Write Rate Control

0: 4 CPUCLK

1: 3 CPUCLK (recommended)

Bit 2 Latency from the Disarming of "Full" to the Assertion of BRDY# for the Pending CPU to PCI Write Cycle

0: 1 CPUCLK (recommended)

1: 2 CPUCLK

Bit 1 CPU-to-PCI Burst Memory Write Enable

0: Disable

1: Enable

Bit 0 CPU-to-PCI Post Memory Write Enable

0: Disable

1: Enable

Register 5Bh (default = 00h)

Bit 7 Enable/Disable DRAM refresh cycle in PCI master cycles

0: Disable

1: Enable

Bits 6:4 Maximum burstable address range in PCI master accessing main memory.

When 32-bit DRAM organization is employed with 256K or 512K type DRAM, maximum burstable range reduces to 2KB only because the physical page size is 2KB in this situation. Thus, never program these bits to 100b in 32 bit DRAM organization.

000: 256B

001: 512B

010: 1KB

011: 2KB

100: 4KB

others: reserved

Bit 3 **to select RAS precharge time 2/3T for BEDO bank**

1: to select 3T

0: to select 2T

Bit 2 **TRDY# assertion timing in PCI master read cycle**

0: Assert TRDY# after prefetching two Qws

1: Assert TRDY# after prefetching one Qw

Bit 1 **Enable/Disable advanced snoop in PCI master write cycle**

0:Disable

1:Enable

Bit 0 **Enable/Disable advanced snoop in PCI master read cycle**

0:Disable

1:Enable

Register 5Ch Default=00h

Bit 7 **Enable/Disable CPU to L2/DRAM and PCI peer-to-peer concurrency mode**

0: Disable

1: Enable

Bit 6 **KWE# Synchronization Clock in PCI master write Asyn. SRAM**

0: ACLK(Recommended)

1: CPUCLK

Bit 5 **Reserved**

This bit should always be programmed to logic 1.

Bit 4 **Reserved**

This bit should always be programmed to logic 0.

Bit 3 **Pshmd of HCR[3:0] timing control in PCI master reading EDO DRAM.**

In reality, this bit defines the timing of MD being pushed into CTPFF wrt. the assertion of CAS#, in PCI master reading EDO cycle. Note that 5512 buffers the prefetching data on MD bus into the rear element of CTPFF on the CPUCLK rising edge that Pshmd is sampled on the HCR[3:0].

0: 1 CPUCLK delay from the assertion of CAS#(recommended in 50MHz)

1: 2 CPUCLK delay from the assertion of CAS#(recommended in 60/66MHz)

Bit 2 Pshmd of HCR[3:0] timing control in PCI master reading FP DRAM

0: 1 CPUCLK delay from the assertion of CAS#(recommended in 50MHz)

1: 2 CPUCLK delay from the assertion of CAS#(recommended in 60/66MHz)

Bit 1 Retiring rate from PTHFF to EDO in PCI master write cycle

0: 3 CPUCLK(Recommended)

1: 2 CPUCLK

Bit 0 Prefetching rate from EDO in PCI master read cycle

0: 3 CPUCLK

1: 2 CPUCLK (Recommended)

Register 5Dh (default = 00h)

Bit 7 Selection of AD[31:0] Current Rating

0: 50mA/2.2V

1: 95mA/2.2V

Bit 6 Selection of Current Rating of FRAME#, IRDY#, TRDY#, DEVSEL#, STOP#, and C/BE[3:0]#

0: 50mA/2.2V

1: 95mA/2.2V

Bit 5 Selection of Current Rating of GNT[3:0]#, PAR

0: 50mA/2.2V

1: 95mA/2.2V

Bit 4 Selection of CASE[7:0]# Buffer Strength

This bit is recommended to be programmed to 1, when 3.3 V DRAM is employed, so that 5511 can provide higher buffer strength.

Bit 3 Selection of Current Rating of CASE[7:0]#

0: 4 mA

1: 8 mA

Bit 2 Selection of CASO[7:0]# Buffer Strength

This bit is recommended to be programmed to 1, when 3.3 V DRAM is employed, so that 5511 can provide higher buffer strength.

Bit 1 Selection of Current Rating of CASO[7:0]#

0: 4 mA

1: 8 mA

Bit 0 Selection of RAS[3:0]# Buffer Strength

This bit is recommended to be programmed to 1, when 3.3 V DRAM is employed, so that 5511 can provide higher buffer strength.

Register 5Eh (default = 00h)

This register mainly defines the enable bits for the events monitored by System Standby timer. If any monitored event occurs during the programmed time, the System standby timer will be reloaded and starts to count down again.

Bit 7 Programmable 10 bit I/O Port Enable

When set, any I/O access to the address will cause the timer be reloaded. The address is defined in Registers 66h and 67h.

Bit 6 Programmable 16 bit I/O Port Enable

When set, any I/O access to the address will cause the timer be reloaded. The address is defined in Registers 6Dh and 6Eh.

Bit 5 Hard Disk Port Enable

When set, any I/O access to the Hard Disk ports (1F0-1F7h or 3F6h) will cause the timer be reloaded.

Bit 4 Serial Port Enable

When set, any I/O access to the Serial Ports (2F8-2FFh, 3F8-3FFh, 2E8-2EFh or 3E8-3EFh) will cause the timer be reloaded.

Bit 3 Parallel Port Enable

When set, any I/O access to the Parallel ports (278-27Fh, 378-37Fh or 3BC-3BEh) will cause the timer be reloaded.

Bit 2 Hold Enable

When set, any event from the ISA master or the PCI Local Master will cause the timer be reloaded.

Bit 1 IRQ1~15, NMI

When set, any event from the IRQ1-15 or NMI will cause the timer be reloaded.

Bit 0 write pulse width generated way of L2 Asynchonous SRAM at updated cyle.

1: By delayed 2 ns's ACLK

0: By CPU's clock

Register 5Fh (default = 00h)

Bits 7:6 **Define the events monitored by the Monitor standby timer**

Bits 5:0 **Define the events to break the Monitor and System standby state.**

Bit 7 **IRQ 1-15, NMI**

When set, any event from the IRQ1-15 or NMI will cause the Monitor standby timer be reloaded.

Bit 6 **HOLD**

When set, any event from the ISA master or the PCI local master will cause the Monitor standby timer be reloaded.

Bit 5 **IRQ 1-15, NMI**

When enabled, any event from the IRQ1-15 or NMI will bring the Monitor back to the Normal state from the Standby state.

Bit 4 **HOLD**

When enabled, any event from the ISA master or the PCI local master will bring the Monitor back to the Normal state from the Standby state.

Bit 3 **IRQ 1-15, NMI**

When enabled, any event from the IRQ1-15 or NMI will de-assert the STPCLK#.

Bit 2 **HOLD**

When enabled, any event from the ISA master or the PCI local master will de-assert the STPCLK#.

Bit 1 **INIT**

When enabled, an event from the INIT will de-assert the STPCLK#.

Bit 0 **Reserved (must be '0')**

Register 60h (default = 00h)

Bit 7 **Reserved. It should be written with 0.**

Bit 6 **Reserved. It should be written with 0.**

Bit 5 **STPCLK# Enable**

When set, writing a '1' to bit 3 of Register 60h will cause the STPCLK# to become active. This bit can be cleared.

Bit 4 **Throttling Enable**

When set, writing a '1' to bit 3 of Register 60h will cause the STPCLK# throttling state to become active. The throttling function can be disabled by clearing this bit.

Bit 3 **STPCLK# Control**

When this bit is set, the STPCLK# will be asserted or the Throttling function will be enabled depending on bits 5 and 4. If both bits 5 and 4 are enabled, the system will do the throttling function.

Bit 2 Break SW., Keyboard reset selection

0: KBRST #

1: BREAK#

The Break SW. disable function can be done by programming register 68 bit 1 to "0".

Bit 1 APM SMI

When Register 68h bit 0 is enabled, and a '1' is written to this bit, an SMI is generated. It is used by the software controlled SMI function like APM. This bit should be cleared at the end of the SMI handler.

Bit 0 Reserved.

Register 61 STPCLK# Assertion Timer (default = FFh)

Bits 7:0 This register defines the period of the STPCLK# assertion time.

Bits[7:0] define the period of the STPCLK# assertion time when the STPCLK# enable bit is set. The timer will not start to count until the Stop Grant Special Cycle is received. The timer slot is 35 us.

Register 62 STPCLK# De-assertion Timer (default = FFh)

Bits 7:0 This register defines the period of the STPCLK# de-assertion time.

Bits[7:0] define the period of the STPCLK# de-assertion time when the STPCLK# enable bit is set. The timer starts to count when the STPCLK# assertion timer expires. The timer slot is 35us.

When these two registers are read, the current values are returned.

Register 63h System Standby Timer (default = FFh)

Bits 7:0 The register defines the duration of the System Standby Timer.

When the System Standby Timer expires, the system enters System Standby State. If any non-masked event occurs before the timer expires, the timer is reloaded with programmed number and the timer starts counting down again.

Register 64h (default = 00h)

Bit 7 M1 SMAC access

It must be set whenever the M1 CCR1 bit 2 is set and cleared if CCR1 bit 3 is cleared.

Bit 6 M1 MMAC access

If set, access to address within SMM space is conducted to main memory instead of SMM area. It must be set whenever the M1 CCR1 bit 3 is set and cleared if CCR1 bit 3 is cleared.

In the M1's specification, the SMIACT will be de-asserted when MMAC is set and re-asserted after it is cleared. This allows the SMI service routine to access normal memory area instead of SMM memory area.

Bit 5 **M1 CPU**

It should be set if the current CPU is M1.

Bit 4 **Toggle Mode Enable**

0: Break SW. without toggle mode

1: Break SW. with toggle mode

Bit 3 **Flush Function Block Mode**

It is suggested to block the FLUSH (Deturbo Mode) when the STPCLK is asserted.

0: Un-block

1: Block

Bit 2 **SMOUT Control**

When this bit is set to "1", the SMOUT is asserted low.

Bit 1 **WAKEUP0, ALT Selection**

1: WAKEUP0

0: ALT

Bit 0 **SMOUT, RAS2# Selection**

1: SMOUT

0: RAS2#

Register 65h (default = 00h)

Bits 7:6 **SMRAM Area Selection**

	Logic Address	**Physical address**
00	E0000h ~ E7FFFh	E0000h ~ E7FFFh
01	E0000h ~ E7FFFh	A0000h ~ A7FFFh
10	E0000h ~ E7FFFh	B0000h ~ B7FFFh
11	Reserved	

Bit 5 **Reserved**

Bit 4 **SMRAM Access Control**

1: When set, the SMRAM area can be used. This bit can be set whenever it is necessary to access the SMRAM area. It is cleared after the access is finished.

0: The SMRAM area can only be accessed during the SMI handler.

Bits 3:0 Reserved

Register 66h (default = 00h)

Bit 7 Reserved

Bits 6:5 Define the time slot of the Monitor Standby timer

00 : 6.6 seconds

01 : 0.84 seconds

10 : 13.3 milli-seconds

11 : 1.6 milli-seconds

Bits 4:2 Programmable 10-bit I/O port address mask bits

000 : No mask

001 : A0 masked

010 : A1-A0 masked

011 : A2-A0 masked

100 : A3-A0 masked

101 : A4-A0 masked

110 : A5-A0 masked

111 : A6-A0 masked

Bits 1:0 Programmable 10-bit I/O port address bits A1, A0.

Bits 1:0 correspond to the address bits A1 and A0.

Register 67h (default = FFh)

Bits 7:0

Bits 7:0 define the programmable 10-bit I/O port address bits A[9:2].

Register 68h (default = 00h)

This register defines the enable status of the devices in SMM. The bits 6:2 are set when the devices are in standby state and cleared when the respective devices are in normal state.

Bit 7 System Standby SMI enable

When no non-masked event occurs during the programmed duration of the system standby timer, the timer expires. If this bit is enabled, the SMI# is generated and the system enters the System Standby state.

Bit 6 Programmable 10-bit I/O port wake up SMI enable

When set, any I/O access to this port will be monitored to generate the SMI# to wake up this I/O port from the standby state to the Normal state. This bit is enabled only when the I/O port is in the Standby state.

Bit 5 Programmable 16-bit I/O port wake up SMI enable

When set, any I/O access to this port will be monitored to generate the SMI# to wake up this I/O port from the standby state to the Normal state. This bit is enabled only when the I/O port is in the Standby state.

Bit 4 Serial ports wake up SMI enable

When set, any I/O access to the serial ports will be monitored to generate the SMI# to wake up the serial ports from the standby state to the Normal state. This bit is enabled only when the serial ports are in the Standby state.

Bit 3 Parallel ports wake up SMI enable

When set, any I/O access to the parallel ports will be monitored to generate the SMI# to wake up the parallel ports from the standby state to the Normal state. This bit is enabled only when the parallel ports are in the Standby state.

Bit 2 Hard Disk port SMI enable

When set, any I/O access to the hard disk port will be monitored to generate the SMI# to wake up the hard disk from the standby state to the Normal state. This bit is enabled only when the hard disk port is in the Standby state.

Bit 1 Break Switch SMI enable

When set, the break switch can be pressed to generate the SMI# for the system to enter the Standby state.

Bit 0 Software SMI enable

When set, an I/O write to register 60h bit 1 will generate an SMI.

Register 69h (default = 00h)

This register defines the SMI request status. If the respective SMI enable bit is set, each specific event will cause the respective bit to be set. The asserted bit should be cleared at the end of the SMI handler.

Bit 7 System Standby SMI request

This bit is set when the system standby timer expires.

Bit 6 Programmable 10-bit I/O port wake up request

This bit is set when there is an I/O access to the port.

Bit 5 Programmable 16-bit I/O port wake up request

This bit is set when there is an I/O access to the port.

Bit 4 Serial ports wake up request

This bit is set when the serial ports are accessed.

Bit 3 Parallel ports wake up request

This bit is set when the parallel ports are accessed.

Bit 2 Hard Disk port wake up request

This bit is set when the hard disk port is accessed.

Bit 1 Break Switch SMI request

This bit is set when the break switch is pressed.

Bit 0 Software SMI request

This bit is set when an I/O write to the bit 1 of register 60h.

Register 6Ah (default = 00h)

Bit 7 Monitor Standby SMI enable

0 : Disable

1 : Enable

When there is no access from the IRQ1-15, HOLD and NMI during the programmed time of the Monitor Standby Timer, the timer expires. If this bit is set, an SMI is generated to bring the Monitor to the standby state.

Bit 6 Monitor Standby SMI request

This bit is set when the Monitor Standby Timer expires. This bit should be cleared at the end of the SMI handler.

Bit 5 Monitor wake up SMI enable

When set, any event from the IRQ1-15, HOLD or NMI will be monitored to generate the SMI# to wake up the monitor from the standby state to the normal state.

Bit 4 Monitor wake up request

This bit is set when there is an event from the IRQ1-15, HOLD or NMI, and the Monitor is in the standby state.

Bit 3 Throttling wake up SMI request

This bit is set when there is any unmasked event from the NMI, INIT, IRQ1-15, or HOLD when the system is in the throttling state.

Bit 2 Throttling wake up SMI enable

When set, any unmasked event from the NMI, INIT, IRQ1-15, or HOLD will cause an SMI to be generated to bring the system back to the Normal state from the throttling state.

Bit 1 System wake up SMI enable

When set, any unmasked event from the NMI, INIT, IRQ1-15, or HOLD will cause an SMI to be generated to bring the system back to the Normal state from the standby state.

Bit 0 System wake up SMI request

This bit is set when there is any unmasked event from the NMI, INIT, IRQ1-15, or HOLD when the system is in the standby state.

Register 6Bh Monitor Standby timer - Low byte (default = FFh)

Bits 7:0 Bits 7:0 define the low byte of the Monitor standby timer.

It is a count-down timer and the time slot is programmable for 6.6s, 0.84s, 13.3 ms or 1.6ms. The value programmed to this register is loaded when the timer is enabled and the timer starts counting down. The timer is reloaded when an event from the IRQ1-15, HOLD or NMI occurs before the timer expires. When this register is read, the current value is returned.

Register 6Ch Monitor Standby timer - High byte (default = FFh)

Bits 7:0 Bits 7:0 define the high byte of the Monitor standby timer.

Register 6Dh Programmable 16-bit I/O port - Low byte (default = FFh)

Bits 7:0 Bits 7:0 define the low byte of the Programmable 16-bit I/O port.

Register 6Eh Programmable 16-bit I/O port - High byte (default = FFh)

Bits 7:0 Bits 7:0 define the high byte of the Programmable 16-bit I/O port.

Register 6Fh (default = 00h)

This register except bit 7 mainly defines the events monitored by the System Standby timer. If any unmasked event occurs before the timer expires, the System Standby Timer will be reloaded and the timer starts to count down again.

Bit 7 Reserved

Bit 6 Reserved

Bit 5 A0000h - AFFFFh or B0000 - BFFFFh Address trap

When set, any memory access to the address range will cause the timer to be reloaded.

Bit 4 C0000h - C7FFFh Address trap

When set, any memory access to the address range will cause the timer to be reloaded.

Bit 3 3B0-3BFh, 3C0-3CFh, 3D0-3DFh Address trap

When set, any I/O access to the I/O addresses will cause the timer to be reloaded.

Bit 2 Secondary Drive port

When set, any I/O access to the secondary drive port (170-17Fh, 320-32Fh, 3F7h) will reload the system standby timer.

Bits 1:0 System Standby Timer Slot

11 : 8.85 milli seconds

10 : 70 milli seconds

01 : 1.1 seconds

00 : 9 seconds

Register 70h DRAM Bank Register 0-0(*) (default = 04h)

NOTE: * means DRAM Bank Register x-y, where x=0, 1, 2, or 3 stand for bank x, and y=0 or 1 stand for even side or odd side, respectively.

Bits 7:0 DRAM Boundary Address HA[28:21]

00h: 0Mbyte

01h: 2Mbyte

02h: 4Mbyte

04h: 8Mbyte

....

....

Register 71h DRAM Bank Register 0-0 (default = 00h)

Bits 7:2 Define the characteristics of Bank 0

Bit 7 Half populated Bank for the first BANK 0

0: Full Populated Bank (64 Bits Data)

1: Half Populated Bank (32 Bits Data)

Bit 6 FP DRAM / EDO DRAM setting

0: FP DRAM

1: EDO DRAM

Bit 5 Double side / Single side DRAM

0: single side DRAM

1: double side DRAM

Bits 4:2 DRAM type setting

000: 1M x N Symmetric DRAM

001: 512K x N 4 Asymmetric DRAM

010: 4M x N Symmetric DRAM

011: 2M x N Asymmetric DRAM (11x10)

100: 4M x N Asymmetric DRAM (12x10)

101: 256K x N Symmetric DRAM

110: 16M x N Symmetric DRAM

111: 1MxN Asymmetric DRAM

Bits 1:0 DRAM Boundary Address HA[30:29]

Register 72h DRAM Bank Register 0-1 (default = 04h)

Bits 7:0 DRAM Boundary Address HA[28:21]

00h: 0Mbyte

01h: 2Mbyte

02h: 4Mbyte

04h: 8Mbyte

....

....

Register 73h DRAM Bank register 0-1 (default = 80h)

Bit 7 This bit is set when RAS0 is populated.

Bit 6 Reserved

Bits 5:3 DRAM type setting of BANK 0

010 1MxN asymmetric DRAM (12 X 8)

011 2MxN asymmetric DRAM (12 X 9)

000, 001, 100, 101 reserverd

110 S3 asymmetric

111 S3 symmetric

Bit 2 BEDO exist or not

0: Non-exist

1: Exist

Bits 1:0 DRAM Boundary Address HA[30:29]

Register 74h DRAM Bank Register 1-0 (default = 04h)

Bits 7:0 DRAM Boundary Address HA[28:21]

00h: 0Mbyte

01h: 2Mbyte

02h: 4Mbyte

04h: 8Mbyte

....

....

Register 75h DRAM Bank Register 1-0 (default = 00h)

Bit 7 Half populated Bank for BANK 1

0: Disable

1 Enable

Bit 6 FP DRAM / EDO DRAM setting

0: FP DRAM

1: EDO DRAM

Bit 5 Double side / Single side DRAM

0: single side DRAM

1: double side DRAM

Bits 4:2 DRAM type setting

000: 1M x N Symmetric DRAM

001: 512K x N Asymmetric DRAM

010: 4M x N Symmetric DRAM

011: 2M x N Asymmetric DRAM (11x10)

100: 4M x N Asymmetric DRAM (12x10)

101: 256K x N Symmetric DRAM

110: 16M x N Symmetric DRAM

111: 1MxN Asymmetric DRAM

Bits 1:0 DRAM Boundary Address HA[30:29]

Register 76h DRAM Bank Register 1-1 (default = 04h)

Bits 7:0 DRAM Boundary Address HA[28:21]

00h: 0Mbyte

01h: 2Mbyte

02h: 4Mbyte

04h: 8Mbyte

....

....

Register 77h DRAM bank Register 1-1 (default = 80h)

Bit 7 **This bit is set when RAS1 is populated.**

Bits 6 **Reserved**

Bit 5:3 **DRAM type setting**

010 asymmetric 12 X 8

011 asymmetric 12 X 9

000, 001, 100, 101 reserverd

110 S3 asymmetric

111 S3 symmetric

Bit 2 **BEDO exist or not**

0: to non-exist

1: to exist

Bits 1:0 **DRAM Boundary Address HA[30:29]**

Register 78h DRAM Bank Register 2-0 (default = 04h)

Bits 7:6 **DRAM Boundary Address HA[28:21]**

00h: 0Mbyte

01h: 2Mbyte

02h: 4Mbyte

04h: 8Mbyte

....

....

Bits 5:3 **DRAM type setting**

010 asymmetric 12 X 8

011 asymmetric 12 X 9

000, 001, 100, 101 reserverd

110 S3 asymmetric

111 S3 symmetric

Bit 2 **BEDO exist or not**

1 to exist

0 to non-exist

Bits 1:0 **DRAM Boundary Address HA[28:21]**

00h: 0Mbyte

01h: 2Mbyte

02h: 4Mbyte

04h: 8Mbyte

....

....

Register 79h DRAM Bank Register 2-0 (default = 00h)

Bit 7 **Half populated Bank for BANK 2**

0: Disable

1: Enable

Bit 6 **FP DRAM / EDO DRAM setting**

0: FP DRAM

1: EDO DRAM

Bit 5 **Double side / Single side DRAM**

0: single side DRAM

1: double side DRAM

Bits 4:2 **DRAM type setting**

000: 1M x N Symmetric DRAM

001: 512K x N Asymmetric DRAM

010: 4M x N Symmetric DRAM

011: 2M x N Asymmetric DRAM (11x10)

100: 4M x N Asymmetric DRAM (12x10)

101: 256K x N Symmetric DRAM

110: 16M x N Symmetric DRAM

111: 1MxN Asymmetric DRAM

Bits 1:0 **DRAM Boundary Address HA[30:29]**

Register 7Ah DRAM Bank Register 2-1 (default = 04h)

Bits 7:0 **DRAM Boundary Address HA[28:21]**

00h: 0Mbyte

01h: 2Mbyte

02h: 4Mbyte

04h: 8Mbyte

....

....

Register 7Bh DRAM Bank Register 2-1 (default = 80h)

Bit 7 **This bit is set when RAS2 is populated.**

Bit 6 **Reserved**

Bits 5:3 **DRAM type setting**

010 asymmetric 12 X 8

011 asymmetric 12 X 9

000, 001, 100, 101 reserved

110 S3 asymmetric

111 S3 symmetric

Bit 2 **BEDO exist or not**

0: Non-exist

1: Exist

Bits 1:0 **DRAM Boundary Address HA[30:29]**

Register 7Ch DRAM Bank Register 3-0 (default = 04h)

Bits 7:0 **DRAM Boundary Address HA[28:21]**

00h: 0Mbyte

01h: 2Mbyte

02h: 4Mbyte

04h: 8Mbyte

....

....

Register 7Dh DRAM Bank Register 3-0 (default = 00h)

Bit 7 **Bit 7: Half populated Bank for BANK 3**

0:Ddisable

1:Enable

Bit 6 **FP DRAM / EDO DRAM setting**

0: FP DRAM

1: EDO DRAM

Bit 5 **Double side / Single side DRAM**

0: single side DRAM

1: double side DRAM

Bits 4:2 DRAM type setting

000: 1M x N Symmetric DRAM

001: 512K x N Asymmetric DRAM

010: 4M x N Symmetric DRAM

011: 2M x N Asymmetric DRAM (11x10)

100: 4M x N Asymmetric DRAM (12x10)

101: 256K x N Symmetric DRAM

110: 16M x N Symmetric DRAM

111: 1MxN Asymmetric DRAM

Bits 1:0 DRAM Boundary Address HA[30:29]

Register 7Eh DRAM Bank Register 3-1 (default = 04h)

Bits 7:0 DRAM Boundary Address HA[28:21]

00h: 0Mbyte

01h: 2Mbyte

02h: 4Mbyte

04h: 8Mbyte

....

....

Register 7Fh DRAM Bank Register 3-1 (default = 80h)

Bit 7 This bit is set when RAS3 is populated.

Bit 6 Reserved

Bits 5:3 DRAM type setting

010 asymmetric 12 X 8

011 asymmetric 12 X 9

000, 001, 100, 101 reserved

110 S3 asymmetric

111 S3 symmetric

Bit 2 BEDO exist or not

0: Non-exist

1: Exist

Bits 1:0 DRAM Boundary Address HA[30:29]

(Register 80h to register 86h define the attribute of the Shadow RAM from 640 KByte to 1 MByte. All of the registers 80h to 85h are defined as below, and each register defines the corresponding memory segment's attribute which are listed in the following table.)

Registers 80h~82h (default = 00h)

Bit 7 Read enable

Bit 6 L1/L2 cacheable

Bit 5 Write enable

Bit 4 Reserved

This bit should always be programmed as 1 to support more than 128M Bytes.

Bit 3 Read enable

Bit 2 L1/L2 cacheable

Bit 1 Write enable

Bit 0 Reserved

This bit should always be programmed as 1 to support more than 128M Bytes.

Register	Defined Range	Register	Defined Range
register 80h bits 7:5	0C0000h-0C3FFFh	register 83h bits 7:5	0D8000h-0DBFFFh
register 80h bits 3:1	0C4000h-0C7FFFh	register 83h bits 3:1	0DC000h-0DFFFFh
register 81h bits 7:5	0C8000h-0CBFFFh	register 84h bits 7:5	0E0000h-0E3FFFh
register 81h bits 3:1	0CC000h-0CFFFFh	register 84h bits 3:1	0E4000h-0E7FFFh
register 82h bits 7:5	0D0000h-0D3FFFh	register 85h bits 7:5	0E8000h-0EBFFFh
register 82h bits 3:1	0D4000h-0D7FFFh	register 85h bits 3:1	0EC000h-0EFFFFh

Registers 83h (default = 00h)

Bit 7 Read enable

Bit 6 L1/L2 cacheable

Bit 5 Write enable

Bit 4 Reserved

This bit should always be programmed as 1 to support more than 128M Bytes.

Bit 3 Read enable

Bit 2 L1/L2 cacheable

Bit 1 Write enable

Bit 0 **Pipeline Burst SRAM with back to back 3-1-1-1-1-1....**

0: to disable, but 3-1-1-1-2-1-1-1...

1: to enable

Registers 84h~85h (default = 00h)

Bit 7 **Read enable**

Bit 6 **L1/L2 cacheable**

Bit 5 **Write enable**

Bit 4 **Reserved**

This bit should always be programmed as 1 to support more than 128M Bytes.

Bit 3 **Read enable**

Bit 2 **L1/L2 cacheable**

Bit 1 **Write enable**

Bit 0 **Reserved**

This bit should always be programmed as 1 to support more than 128M Bytes.

Register 86h (default = 00h)

Bits 7:4 define the attributes of BIOS area 0F0000-0FFFFFh

Bit 7 **Read enable**

Bit 6 **L1/L2 cacheable**

Bit 5 **Write enable**

Bit 4 **Reserved**

Bit 3 **Shadow RAM enable for PCI master access**

0: Disable

1: Enable

Bits 2:1 **Reserved**

Bit 0 **DRAM refresh arbited way**

0 for normal refresh

1 for advanced refresh

Register 90~93h - 5512 General Purpose Register Index

Register A4, default 0

Bit 7 **to select internal ACLK output to RAS2#, only for adjust internal ACLK delay**

0: Disable

1: Enable

This bit can't enable at the same time as register 64's bit 0, 2.

Bit 6 **32/64 bit auto Bank enable**

0: to disable, only support Bank 0 for 32 bit DRAM of FP and EDO

1: to enable auto Bank, with any slot's 32 bit DRAM of FP and EDO

Bit 5:1 **Advanced clock (ACLK) delay scale from CPUCLK's rising edge**

00000: to delay 4.5 ns from CPUCLK

00001: to delay 4.9 ns from CPUCLK

00010: to delay 5.5 ns from CPUCLK

~~~~~

00100: to delay 6.5 ns from CPUCLK

~~~~~

01000: to delay 8.5 ns from CPUCLK

~~~~~

10000: to delay 12.5 ns from CPUCLK

~~~~~

Bit 0 **to select internal or external ACLK**

0: to select external ACLK

1: to select internal ACLK

Register A5, default 0

Bit 7 **to select enhanced post write**

0: to disable

1: to enable

Bit 6:3 **Reserved**

Bit 2 **BEDO write cycle time**

1: to 1T cycle time

0: to 2T cycle time

Bit 1 **turn on programming WCBR's BEDO cycle**

1: to enable

0: to disable

This bit is turned on at programming state, but it will turn off after leaving programming state.

Bit 0 **to select DRAM 5-X-X-X-X ...**

0: to disable

1: to enable

Register A6, default 0

Bit 7:5 **Select added extra CASO# and CASE# delay**

000: to delay 0 ns

001: to delay 1 ns

010: to delay 2 ns

011: to delay 3 ns

100: to delay 4 ns

101: to delay 5 ns

110: to delay 6 ns

111: to delay 7 ns

Bit 4 **Reserved**

Bit 3:1 **to select added extra MDLE delay**

000: to delay 0 ns

001: to delay 1 ns

010: to delay 2 ns

011: to delay 3 ns

100: to delay 4 ns

101: to delay 5 ns

110: to delay 6 ns

111: to delay 7 ns

Bit 0 **Reserved**

Register A7, default 0

Bit 7:4 **Reserved**

Bit 3:1 **select KWE# delay of CPUCLK at PCI master cycle**

000: to delay 4 ns

001: to delay 5 ns

010: to delay 6 ns

011: to delay 7 ns

100: to delay 8 ns

101: to delay 9 ns

110: to delay 10 ns

111: to delay 11 ns

Bit 0 **Reserved**

2.3.3 SLiC Memory Mapped Registers

IPI Dispatch Register

(1) Address: XFC00000h. (X depends on the value of global register.)
(2) Write-only register.
(3) The data which writes to this register will be disregarded.

8259 Interrupt Mask Register

(1) Address: XFC00010h.
(2) 1-bit write-only registers (D0).
(3) The initial value: 0 for CPU1 and 1 for CPU2.
(4) Each CPU has its own 8259 interrupt mask register which can be used to mask interrupt request from 8259. The CPU can program its own register only. For example, the CPU1 execute a non-cacheable memory write to set its own 8259 mask register to 1. All the interrupts to CPU 1 will be masked.

Toggle Mode Register

(1) Address: XFC00018h.
(2) 1-bit write-only register (D0).
(3) The initial value is 0.
(4) If its value is 0, all 8259 interrupt requests will be routed to CPU1. If the toggle mode register is set, the interrupt requests will send to CPU1 and CPU2 alternatively. Note that the IPI will not cause toggling.

Not M1 Mode Register

(1) Address: XFC00028h.
(2) 1-bit write-only register (D0).
(3) The initial value is 0.
(4) This register is used to indicate whether the Cyrix M1 CPU is used (value=1) or not (value=0)

Global Register

(1) Address: XFC00030h.
(2) 4-bit write-only register (D3-D0).

(3) The initial value is 1111b.

Multi-processor Enable Register

(1) Address: XFC00038h.
(2) 1-bit write-only register (D0).
(3) The initial value is 0.
(4) If the value is 1, the multi-processor mode is enabled.

2.4 Pin Assignment and Description

2.4.1 Pin Assignment

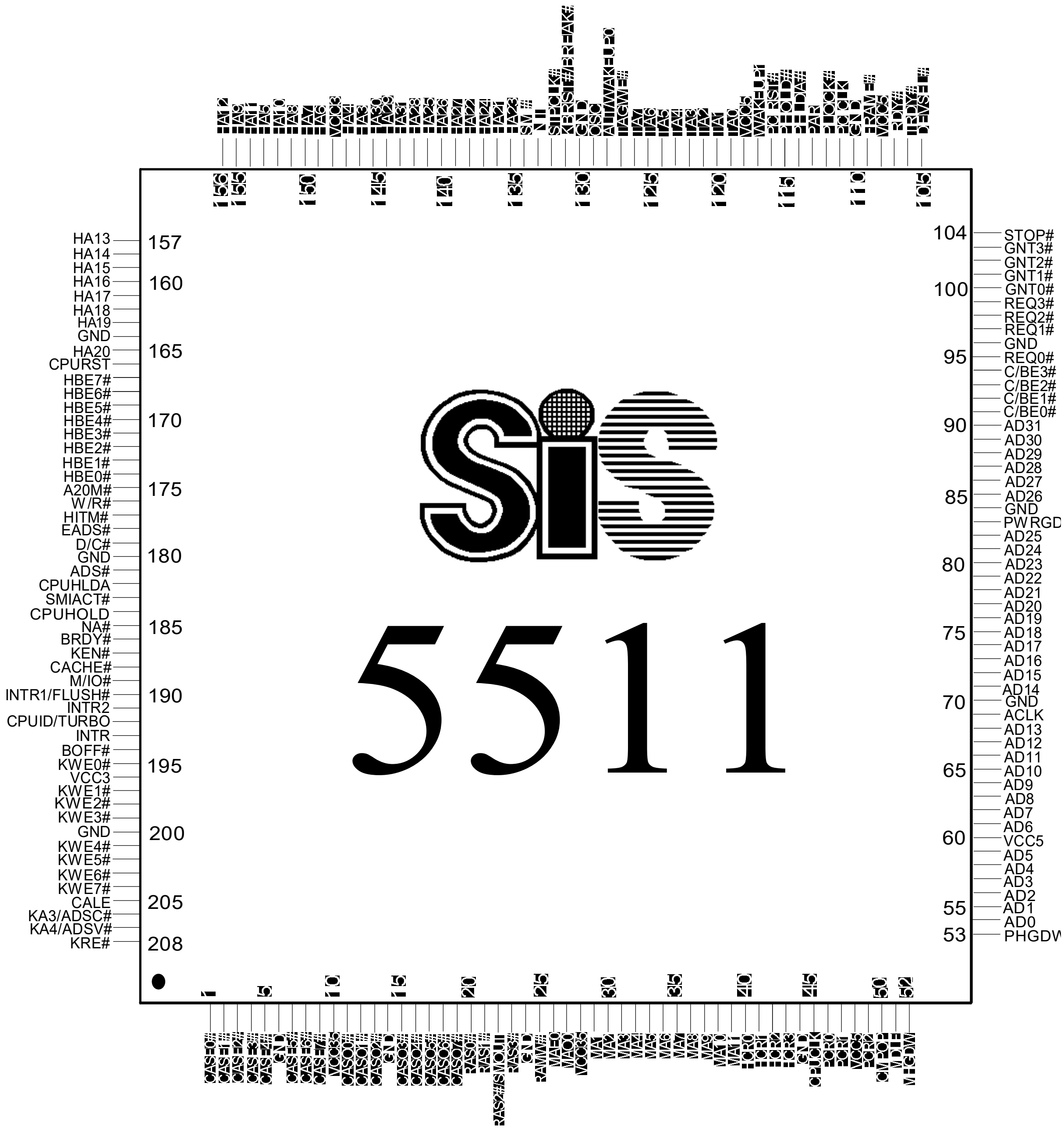

HA13
HA14
HA15
HA16
HA17
HA18
HA19
GND
HA20
CPURST
HBE7#
HBE6#
HBE5#
HBE4#
HBE3#
HBE2#
HBE1#
HBE0#
A20M#
W/R#
HITM#
EADS#
D/C#
GND
ADS#
CPUHLDA
SMIACT#
CPUHOLD
NA#
BRDY#
KEN#
CACHE#
M/IO#
INTR1/FLUSH#
INTR2
CPUID/TURBO
INTR
BOFF#
KWE0#
VCC3
KWE1#
KWE2#
KWE3#
GND
KWE4#
KWE5#
KWE6#
KWE7#
CALE
KA3/ADSC#
KA4/ADSV#
KRE#
157
160
165
170
175
180
185
190
195
200
205
208
104
100
95
90
85
80
75
70
65
60
55
53
STOP#
GNT3#
GNT2#
GNT1#
GNT0#
REQ3#
REQ2#
REQ1#
GND
REQ0#
C/BE3#
C/BE2#
C/BE1#
C/BE0#
AD31
AD30
AD29
AD28
AD27
AD26
GND
PWRGD
AD25
AD24
AD23
AD22
AD21
AD20
AD19
AD18
AD17
AD16
AD15
AD14
GND
ACLK
AD13
AD12
AD11
AD10
AD9
AD8
AD7
AD6
VCC5
AD5
AD4
AD3
AD2
AD1
AD0
PHGDW
5511

Figure 2.7

2.4.2 Pin Listing (# means active low)

Table 2-18

1=CASE0#	3V/5V	48=VCC5	5V	95=REQ0#	5V
2=CASE1#	3V/5V	49=PCR2	5V	96=GND	VSS
3=CASE2#	3V/5V	50=CMPOP	5V	97=REQ1#	5V
4=CASE3#	3V/5V	51=MDLE	5V	98=REQ2#	5V
5=CASE4#	3V/5V	52=MHGDW	5V	99=REQ3#	5V
6=GND	VSS	53=PHGDW	5V	100=GNT0#	5V
7=CASE5#	3V/5V	54=AD0	5V	101=GNT1#	5V
8=CASE6#	3V/5V	55=AD1	5V	102=GNT2#	5V
9=CASE7#	3V/5V	56=AD2	5V	103=GNT3#	5V
10=VCC35	3V/5V	57=AD3	5V	104=STOP#	5V
11=CASO0#	3V/5V	58=AD4	5V	105=DEVSEL#	5V
12=CASO1#	3V/5V	59=AD5	5V	106=TRDY#	5V
13=CASO2#	3V/5V	60=VCC5	5V	107=IRDY#	5V
14=GND	VSS	61=AD6	5V	108=VCC5	5V
15=CASO3#	3V/5V	62=AD7	5V	109=FRAME#	5V
16=CASO4#	3V/5V	63=AD8	5V	110=GND	VSS
17=CASO5#	3V/5V	64=AD9	5V	111=PCICLK	5V
18=CASO6#	3V/5V	65=AD10	5V	112=PLOCK#	5V
19=CASO7#	3V/5V	66=AD11	5V	113=PAR	5V
20=RAS0#	3V/5V	67=AD12	5V	114=PHLDA#	5V
21=RAS1#	3V/5V	68=AD13	5V	115=PHOLD#	5V
22=RAS2#/SMOUT	3V/5V	69=ACLK	5V	116=PCIRST#	5V
23=RAS3#	3V/5V	70=GND	VSS	117=WAKEUP1	5V
24=GND	VSS	71=AD14	5V	118=VCC5	5V
25=RAMW#	3V/5V	72=AD15	5V	119=TA0	5V
26=MAE0	3V/5V	73=AD16	5V	120=TA1	5V
27=MAO0	3V/5V	74=AD17	5V	121=TA2	5V
28=VCC35	3V/5V	75=AD18	5V	122=TA3	5V
29=MA1	3V/5V	76=AD19	5V	123=TA4	5V
30=MA2	3V/5V	77=AD20	5V	124=TA5	5V
31=MA3	3V/5V	78=AD21	5V	125=TA6	5V
32=MA4	3V/5V	79=AD22	5V	126=TA7	5V
33=MA5	3V/5V	80=AD23	5V	127=TAGWE#	5V
34=MA6	3V/5V	81=AD24	5V	128=ALT/WAKEUP0	5V
35=MA7	3V/5V	82=AD25	5V	129=OSC	5V
36=MA8	3V/5V	83=PWRGD	5V	130=GND	VSS
37=MA9	3V/5V	84=GND	VSS	131=KBRST#/BREAK#	5V
38=MA10	3V/5V	85=AD26	5V	132=STPCLK#	3V
39=MA11	3V/5V	86=AD27	5V	133=INIT	3V
40=HCR0	5V	87=AD28	5V	134=SMI#	3V
41=HCR1	5V	88=AD29	5V	135=HA23	3V
42=HCR2	5V	89=AD30	5V	136=HA21	3V
43=HCR3	5V	90=AD31	5V	137=HA24	3V
44=GND	VSS	91=C/BE0#	5V	138=HA22	3V
45=CPUCLK	5V	92=C/BE1#	5V	139=HA27	3V
46=PCR0	5V	93=C/BE2#	5V	140=HA26	3V
47=PCR1	5V	94=C/BE3#	5V	141=HA25	3V

142=HA28	3V	176=W/R#	3V

143=HA31	3V	177=HITM#	3V
144=HA29	3V	178=EADS#	3V
145=HA30	3V	179=D/C#	3V
146=HA3	3V	180=GND	VSS
147=HA4	3V	181=ADS#	3V
148=VCC3	3V	182=CPUHLDA	3V
149=HA6	3V	183=SMIACT#	3V
150=HA7	3V	184=CPUHOLD	3V
151=HA8	3V	185=NA#	3V
152=HA10	3V	186=BRDY#	3V
153=HA5	3V	187=KEN#	3V
154=HA11	3V	188=CACHE#	3V
155=HA9	3V	189=M/IO#	3V
156=HA12	3V	190=INTR1/FLUSH#	3V
157=HA13	3V	191=INTR2	3V
158=HA14	3V	192=CPUID/TURBO	3V
159=HA15	3V	193=INTR	3V
160=HA16	3V	194=BOFF#	3V
161=HA17	3V	195=KWE0#	3V
162=HA18	3V	196=VCC3	3V
163=HA19	3V	197=KWE1#	3V
164=GND	VSS	198=KWE2#	3V
165=HA20	3V	199=KWE3#	3V
166=CPURST	3V	200=GND	VSS
167=HBE7#	3V	201=KWE4#	3V
168=HBE6#	3V	202=KWE5#	3V
169=HBE5#	3V	203=KWE6#	3V
170=HBE4#	3V	204=KWE7#	3V
171=HBE3#	3V	205=CALE	3V
172=HBE2#	3V	206=KA3/ADSC#	3V
173=HBE1#	3V	207=KA4/ADSV#	3V
174=HBE0#	3V	208=KRE#	3V
175=A20M#	3V		

2.4.3 Pin Description

Table 2-19 Host Interface Pin Description

Pin No.	Symbol	Typ	Function
143, 145-144, 142, 139-140, 141, 137, 135, 138, 136, 165, 163-156, 154, 152, 155, 151-149, 153, 147-146	HA[31:3]	I/O	The CPU Address is driven by the CPU during CPU bus cycles. The 5511 forwards it to either the DRAM or the PCI bus depending on the address range. The address bus is driven by the 5511 during bus master cycles.
167-174	HBE[7:0]#	I	CPU Byte Enables indicate which byte lanes on the CPU data bus carry valid data during the current bus cycle. HBE7# indicates that the most significant byte of the data bus is valid while HBE0# indicates that the least significant byte of the data bus is valid.
181	ADS#	I	Address Status is driven by the CPU to indicate the start of a CPU bus cycle.
189	M/IO#	I	Memory I/O definition is an input to indicate an I/O cycle when low, or a memory cycle when high.
176	W/R#	I/O	Write/Read from the CPU indicates whether the current cycle is a write or read access. It is an output during the PCI master cycles.
179	D/C#	I	Data/Code is used to indicate whether the current cycle is a data or code access.
186	BRDY#	O	Burst Ready indicates that data presented are valid during a burst cycle.
184	CPUHOLD	O	CPU Hold Request is used to request the control of the CPU bus. CPUHLDA will be asserted by the CPU after completing the current bus cycle.

182	CPUHLDA	I	CPU Hold Acknowledge comes from the CPU in response to a CPUHOLD request. It is active high and remains driven during bus hold period. The assertion of CPUHLDA indicates that the CPU has relinquished the control of the host bus..
177	HITM#	I	Hit Modified indicates the snoop cycle hits a modified line in the L1 cache of the CPU.
175	A20M#	O	A20 Mask is the fast A20GATE output to the CPU. It remains high during power up and CPU reset period. It forces A20 to go low when active.
187	KEN#	O	The CPU Cache Enable pin is used when the current cycle is cacheable to the L1 cache of the CPU. It is an active low signal asserted by the 5511 during cacheable cycles.
188	CACHE#	I	The Cache pin indicates an L1 internally cacheable read cycle or a burst write-back cycle. If this pin is driven inactive during a read cycle, the CPU will not cache the returned data, regardless of the state of the KEN# pin.
178	EADS#	O	The EADS# is driven to indicate that a valid external address has been driven to the CPU address pins to be used for an inquire cycle.
185	NA#	O	The 5511 always asserts NA# no matter the asynchronous, burst, or pipelined burst SRAMs are used.
194	BOFF#	O	The 5511 asserts BOFF# to stop the current CPU cycle.
166	CPURST	O	Reset CPU is an active high output to reset the CPU.
133	INIT	O	The Initialization output forces the CPU to begin execution in a known state. The CPU state after INIT is the same as the state after CPURST except that the internal caches, model specific registers, and floating point registers retain the values they had prior to INIT.
134	SMI#	O	System Management Interrupt is used to indicate the occurrence of system management events. It is connected directly to the CPU SMI# input.

183	SMIACT#	I	The SMIACT# pin is used as the SMI acknowledgment input from the CPU to indicate that the SMI# is being acknowledged and the processor is operating in System Management Mode(SMM).
132	STPCLK#	O	Stop Clock indicates a stop clock request to the CPU.
193	INTR	I	In Dual Processor mode, this pin is connected to 5513 "INTR".
190	INTR1/FLUSH#	O	In Dual Processor mode, this pin is connected to INTR of CPU1. When this pin is programmed as FLUSH#, it is used to slow down the system in deturbo mode.
191	INTR2	O	In Dual Processor mode, this pin is connected to INTR of CPU2.
192	CPUID/TURBO	I	When this pin is used as CPUID, it is connected to U/O# of P54C processor. Another function of this pin is used to slow down the system by connecting it to ground.

Table 2-20 Cache & DRAM Interface

Pin No.	Symbol	Type	Function
126-119	TA[7:0]	I/O	TAG RAM data bus lines.
206	KA3/ADSC#	O	Cache address 3 for asynchronous SRAM or cache address strobe for burst and pipelined burst SRAM.
207	KA4/ADSV#	O	Cache address 4 for asynchronous SRAM or cache address advance for burst and pipelined burst DRAM.
208	KRE#	O	Cache Read Enable for standard SRAM, or Cache Output Enable for burst and pipelined burst SRAM.
204-201, 199-197, 195	KWE[7:0]#	O	Cache write enable signal.
205	CALE#	O	The CALE controls the external latch between the host address lines and the cache address lines. When high, it allows the CPU address lines to propagate through external latches and onto cache address lines. When low, it is used to latch cache address lines.
127	TAGWE#	O	TAG RAM write enable output.
128	ALT/WAKEUP0	I/O	This pin is used as either dirty bit of cache or WAKEUP0.

22	RAS2#/SMOUT	O	Row address strobe 2 for DRAM Bank 2 or acts as SMOUT to control external device.
23	RAS3#	O	Row address strobe 3 for DRAM bank 3.
21-20	RAS[1:0]#	O	Row address strobe 1-0 for DRAM banks 1-0.
9-7, 5-1	CASE[7:0]#	O	Even column address strobe 7-0 for byte 7-0.
19-15, 13-11	CASO[7:0]#	O	Odd column address strobe 7-0 for byte 7-0.
39-29	MA[11:1]	O	Memory address 11-1 are the row and column addresses for DRAM.
26	MAE0	O	Memory address 0 for even bank.
27	MAO0	O	Memory address 0 for odd bank.
25	RAMW#	O	RAM Write is an active low output signal to enable local DRAM writes.

Table 2-21 PCI Interface

Pin No.	Symbol	Type	Function
94-91	C/BE[3:0]#	I/O	PCI Bus Command and Byte Enables define the PCI command during the address phase of a PCI cycle, and the PCI byte enables during the data phases. C/BE[3:0]# are outputs when the 5511 is a PCI bus master and inputs when it is a PCI slave.
90-85, 82-71, 68-61, 59-54	AD[31:0]	I/O	PCI Address /Data Bus *In address phase:* 1. When the 5511 is a PCI bus master, AD[31:0] are output signals. 2. When the 5511 is a PCI target, AD[31:0] are input signals. *In data phase:* 1. When the 5511 is a target of a memory read/write cycle, AD[31:0] are floating. 2. When the 5511 is a target of a configuration or an I/O cycle, AD[31:0] are output signals in a read cycle, and input signals in a write cycle.
109	FRAME#	I/O	FRAME# is an output when the 5511 is a PCI bus master. The 5511 drives FRAME# to indicate the beginning and duration of an access. When the 5511 is a PCI slave, FRAME# is an input signal.

107	IRDY#	I/O	IRDY# is an output when the 5511 is a PCI bus master. The assertion of IRDY# indicates the current PCI bus master's ability to complete the current data phase of the transaction. For a read cycle, IRDY# indicates that the PCI bus master is prepared to accept the read data on the following rising edge of the PCI clock. For a write cycle, IRDY# indicates that the bus master has driven valid data on the PCI bus. When the 5511 is a PCI slave, IRDY# is an input.
106	TRDY#	I/O	TRDY# is an output when the 5511 is a PCI slave. The assertion of TRDY# indicates the target agent's ability to complete the current data phase of the transaction. For a read cycle, TRDY# indicates that the target has driven valid data onto the PCI bus. For a write cycle, TRDY# indicates that the target is prepared to accept data from the PCI bus. When the 5511 is a PCI master, it is an input.
105	DEVSEL#	I/O	The 5511 drives DEVSEL# based on the DRAM address range being accessed by a PCI bus master or if the current configuration cycle is to the 5511. As an input it indicates if any device has responded to current PCI bus cycle initiated by the 5511.
104	STOP#	I/O	STOP# indicates that the bus master must start terminating its current PCI bus cycle at the next clock edge and release control of the PCI bus. STOP# is used for disconnect, retry, and target-abort sequences on the PCI bus.
113	PAR	O	Parity is an even parity generated across AD[31:0] and C/BE[3:0]#.
99-97, 95	REQ[3:0]#	I	PCI Bus Request is used to indicate to the PCI bus arbiter that an agent requires use of the PCI bus.
103-100	GNT[3:0]#	O	PCI Bus Grant indicates to an agent that access to the PCI bus has been granted.
112	PLOCK#	I	PCI Lock indicates an exclusive bus operation that may require multiple transactions to complete. When PLOCK# is sampled asserted at the beginning of a PCI cycle, the 5511 considers itself a locked resource and remains in the locked state until PLOCK# is sampled negated on a new PCI cycle.

111	PCICLK	I	The PCICLK input provides the fundamental timing and the internal operating frequency for the 5511. It runs at the same frequency and skew of the PCI local bus.
116	PCIRST#	O	The PCI Reset forces the PCI devices to a known state.

Table 2-22 Data Buffer Control Interface

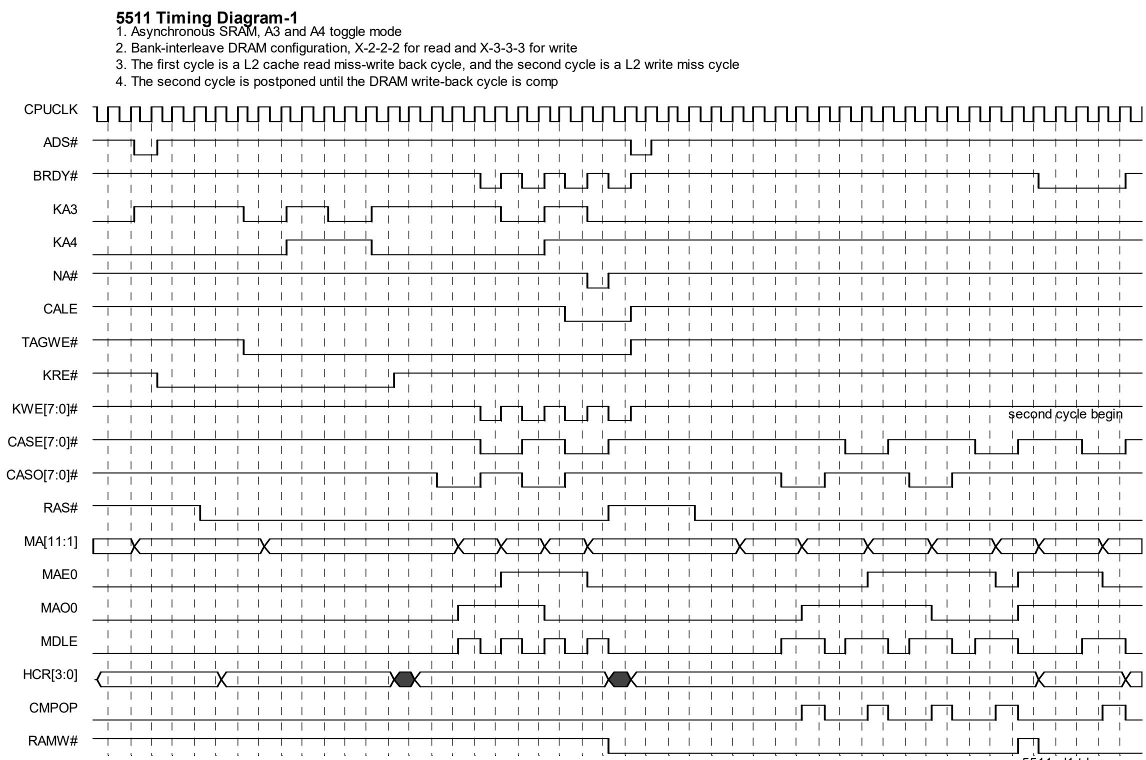
Pin No.	Symbol	Type	Function
43-40	HCR[3:0]	O	Host Data Bus Controls. These signals are driven by the 5511 and are used to control the 5512 HD[63:0] bus and 5512 internal FIFO.
49, 47-46	PCR[2:0]	O	PCI Data Bus Controls. These signals are driven by the 5511 and are used to control the 5512 AD[31:0] bus and 5512 internal FIFO.
51	MDLE	O	Memory Data Read Latch Enable.
50	CMPOP	O	When this signal is sampled active on CPUCLK rising edge, the rear pointer of the CTMFF is forwarded.
52	MHGDW	O	Memory high double word indicator. When high, the high DW of the rear element of the CTMFF or PTHFF is driven onto the low DW of the MD bus.
53	PHGDW	O	PCI high double word indicator.

Table 2-23 Misell

Pin No.	Symbol	Type	Function
129	OSC	I	OSC is the time base of refresh counter. It is 14.318MHz and is generated by an external oscillator.
83	PWRGD	I	Power Good is a power on reset and push button reset input.
115	PHOLD#	I	SIO Request from the 5513 to request the PCI bus.
114	PHLDA#	O	SIO Grant. When asserted, PHLDA# indicates that the PCI arbiter has granted use of the bus to the 5513.
117	WAKEUP1	I	When this input is activated, the 5511 will reload the system standby timer. If it is inactive and the system standby timer expires, the system will enter system standby state. During the system standby state, if this input becomes active, the system will wake up from standby state and return back to normal state.

131	KBRST# / BREAK#	I	When the break switch enable bit is set, the KBRST# will be disabled. A signal from the break switch will cause the system enters the standby state. The pulse width of the BREAK# must greater than 4 CPUCLK.
45	CPUCLK	I	CPU clock input runs at the frequency and skew equal to those of the CPU clock.
69	ACLK	I	Advanced CPU clock should lead the CPUCLK by 4 to 7 ns to provide the clock for the 5511 internal cache control logic.
60, 48, 108, 118	VCC5		+5V DC Power
196, 148	VCC3		+3V DC Power
10, 28	VCC35		Power Signals for DRAM interface. Connected to +5V DC Power for 5V DRAM, while connected to +3V DC power for 3V DRAM.
6, 14, 24, 44, 180, 200, 164, 130, 110, 96, 84, 70	GND		Ground

2.4.4 Timing Diagram



8Figure 2.9

5511 Timing Diagram-2

1. Cache configuration: asynchronous SRAM, A3 & A4 toggle mode
2. DRAM configuration: 32 bits EDO DRAM, speed setting 6-2-2-2
3. The first cycle is a L2 cache read miss-write back cycle, the second cycle is a L2 write hit cycle

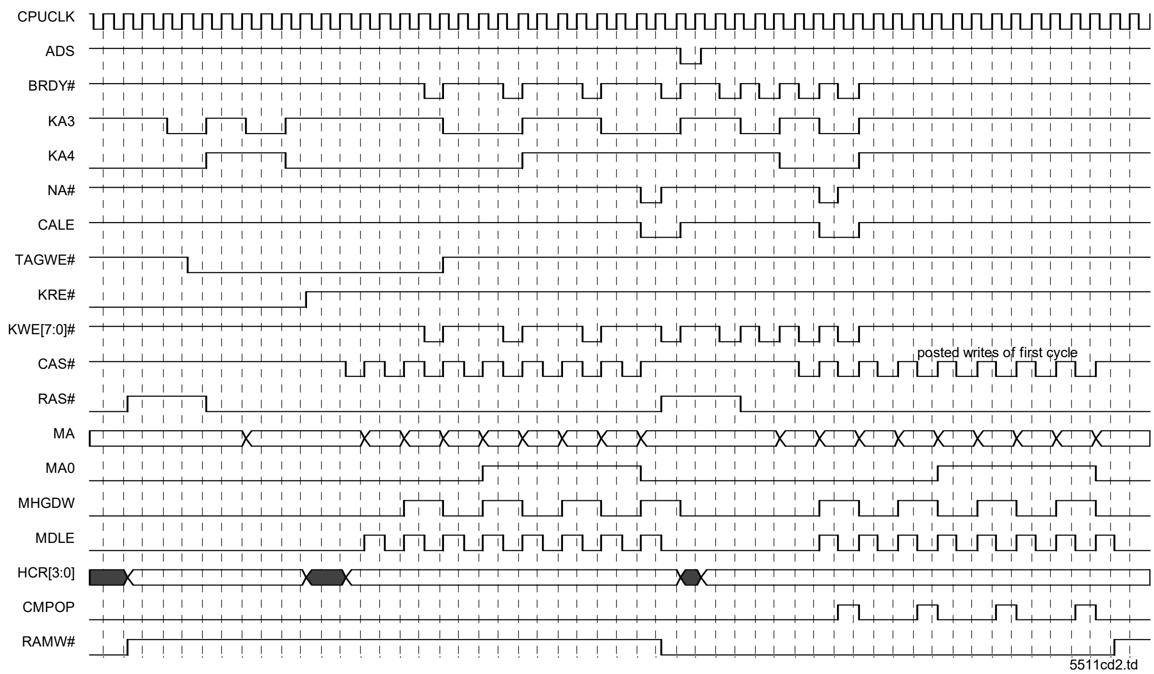


Figure 2.10

5511 Timing Diagram-3

1. Configuration: asynchronous SRAM, 32 bits EDO DRAM with X-2-2-2 foe read and X-3-3-3 for write
2. First cycle is L2 cache read miss update cycle(no write back)
3. Second cycle is DRAM posted write cycle
4. Third cycle is L2 cache read hit cycle
5. Fourth cycle is DRAM posted write cycle

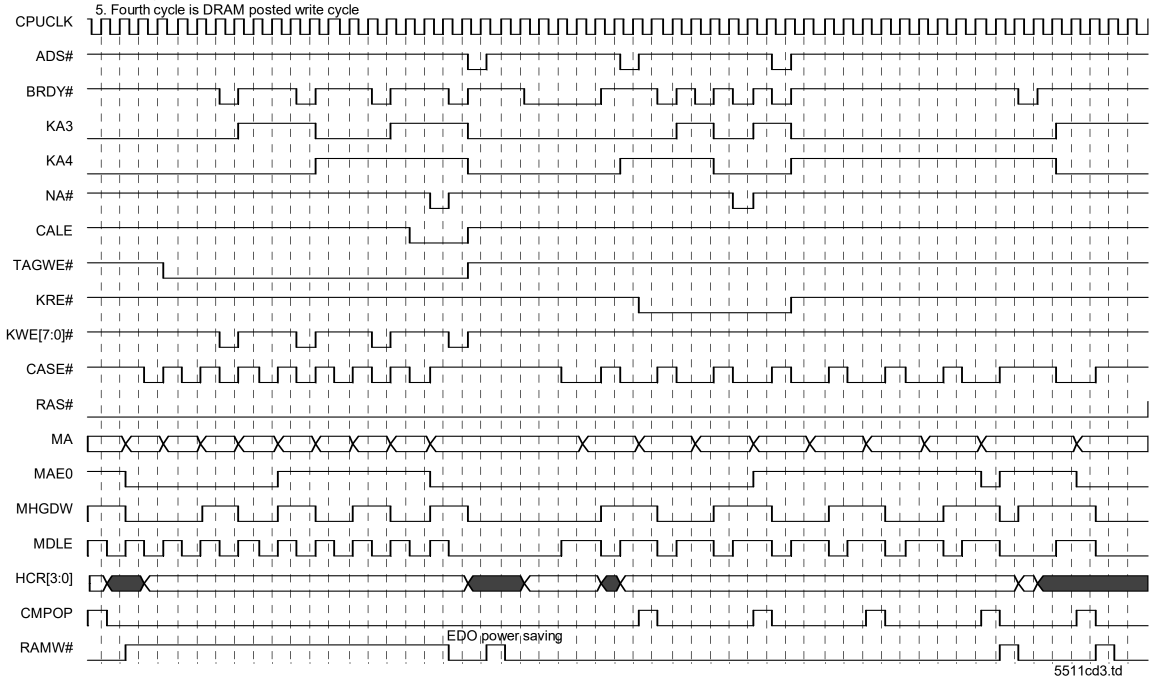


Figure 2.11

5511 Timing Diagram-4

1. Configuration: pipelined burst SRAM with 3-1-1-1 speed setting, EDO DRAM non-interleave mode with
2. L2 cache read miss-write back cycle

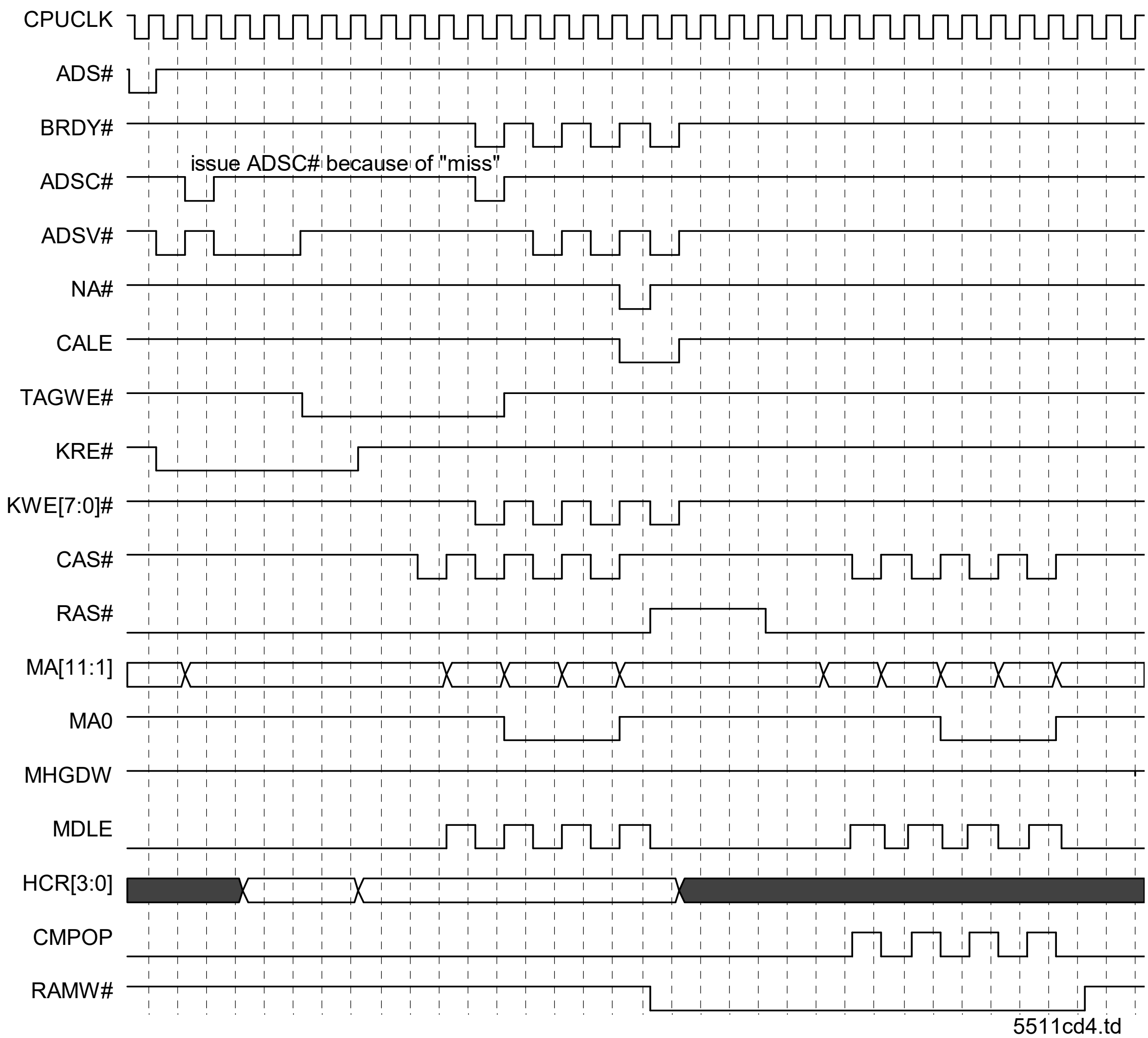


Figure 2.12

5511 Timing Diagram-5

1. Configuration: Burst SRAM,speed setting 3-1-1-1. EDO DRAM non-interleave mode,speed setting 6-2-2-2.
2. The first cycle is a L2 cache read miss-write back cycle
3. The second cycle is a burst write DRAM cycle (L2 cache write miss)

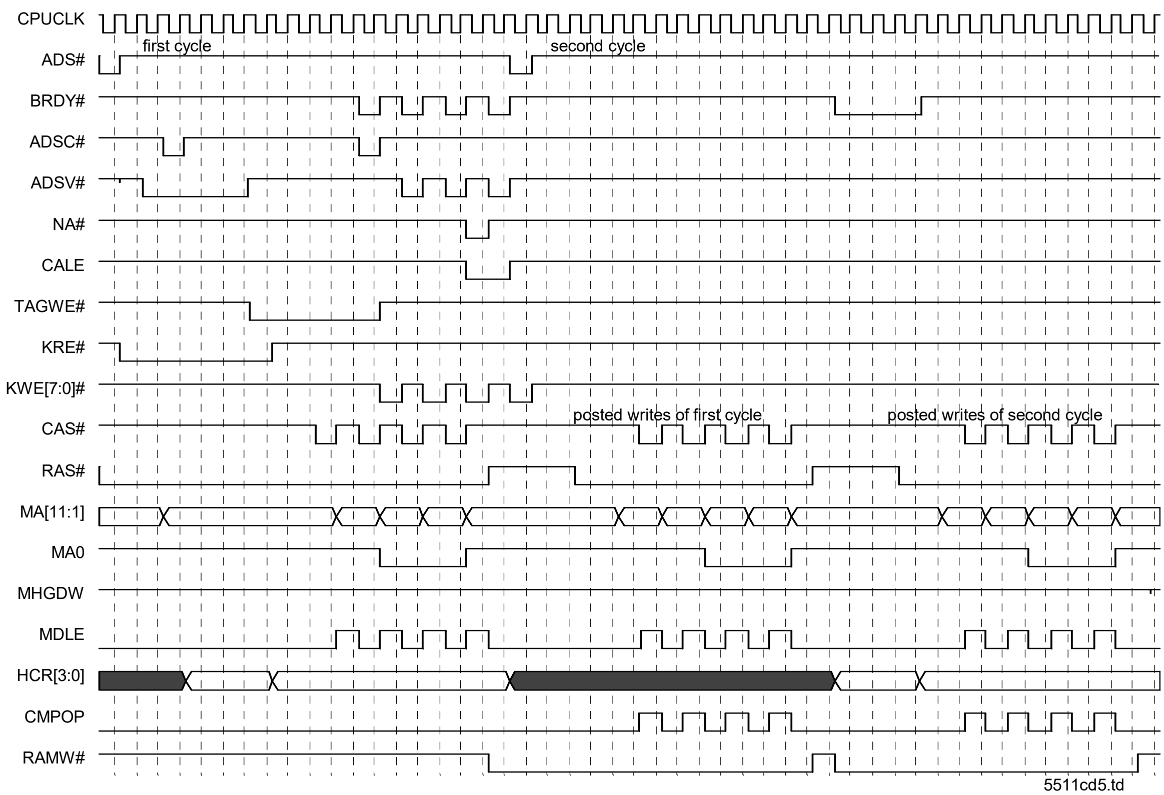


Figure 2.13

5511 Timing Diagram-6

1. Configuration: pipelined burst SRAM with speed setting 3-1-1-1, FP DRAM with speed setting R/W 6-3-3-3
2. First cycle is L2 cache read miss-write back cycle
3. Second cycle is L2 cache read hit cycle

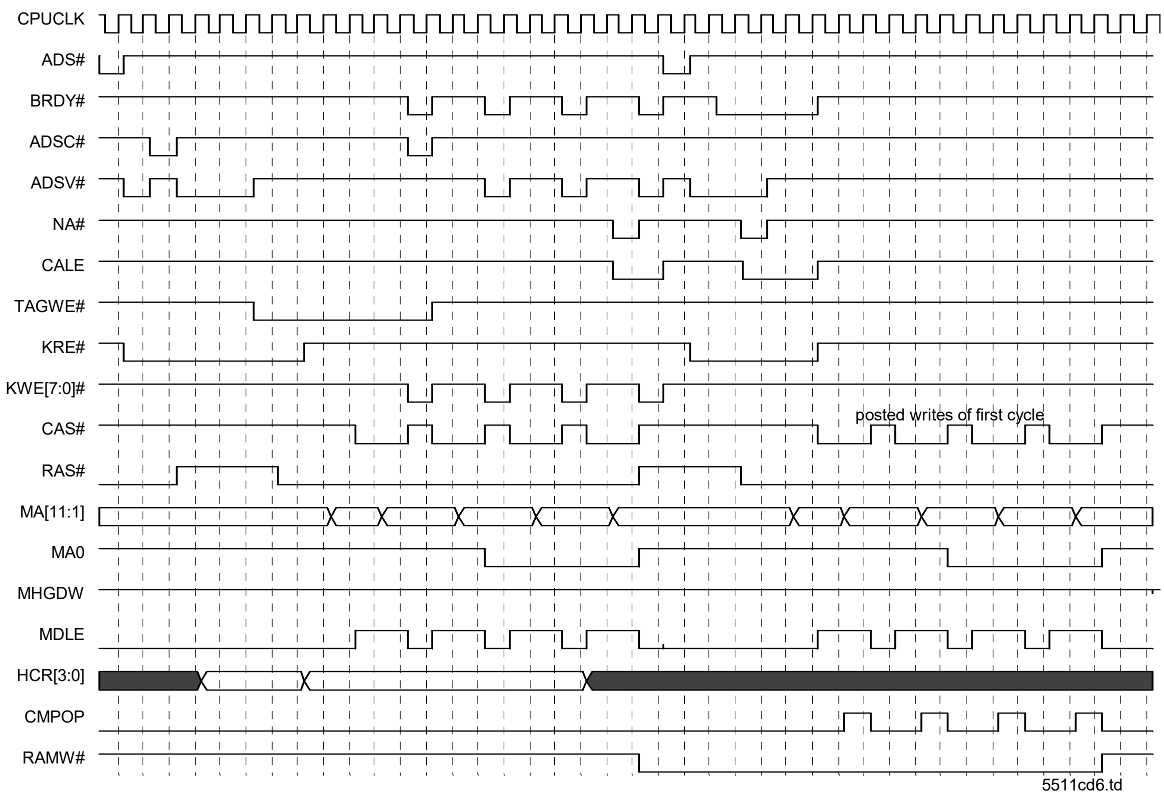


Figure 2.14

5511 Timing Diagram-7

1. Configuration: pipelined burst SRAM with speed setting 3-1-1-1, EDO DRAM with speed setting 6-2-2-2
2. First cycle is L2 cache read miss-write back cycle
3. Second cycle is L2 cache read hit cycle
4. Third cycle is L2 cache write hit cycle

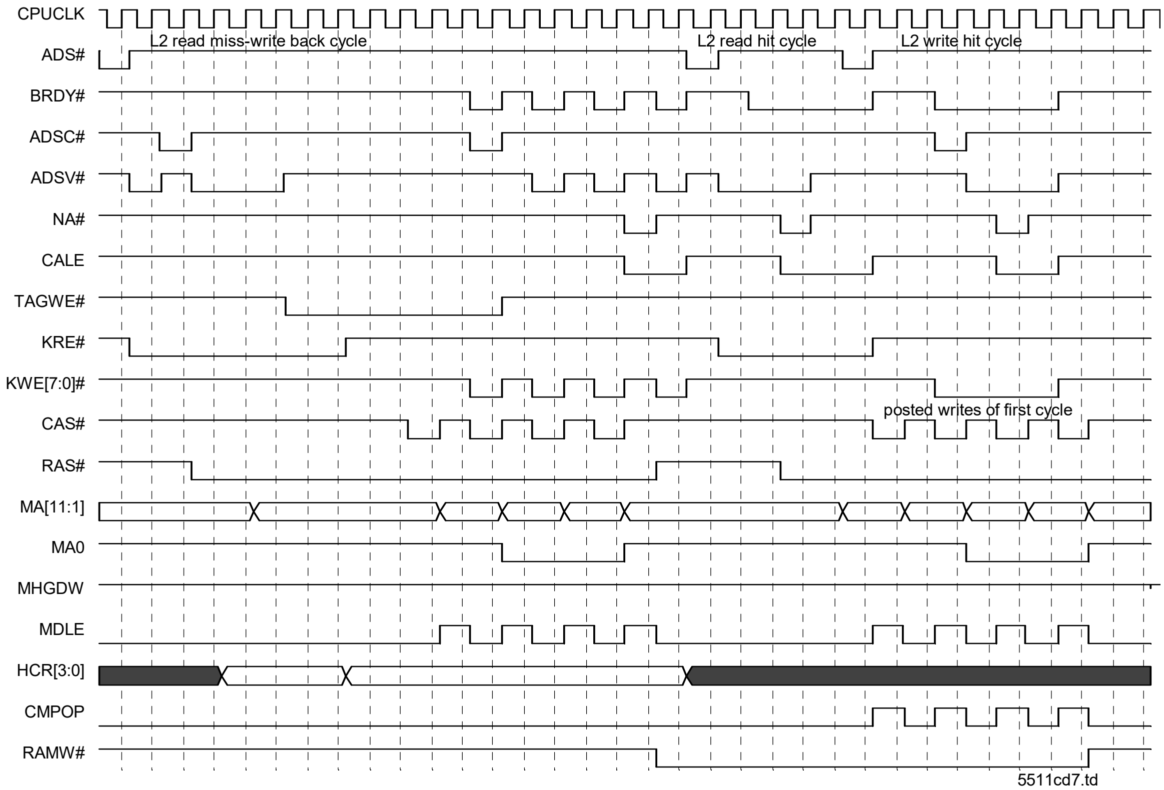


Figure 2.15

5511 Timing Diagram-8

1. Configuration: pipelined burst SRAM with speed setting 3-1-1-1, FP DRAM interleaved mode with speed setting 6-2-2-2
2. First cycle is L2 read miss-write back cycle
3. Second cycle is L2 cache read hit cycle

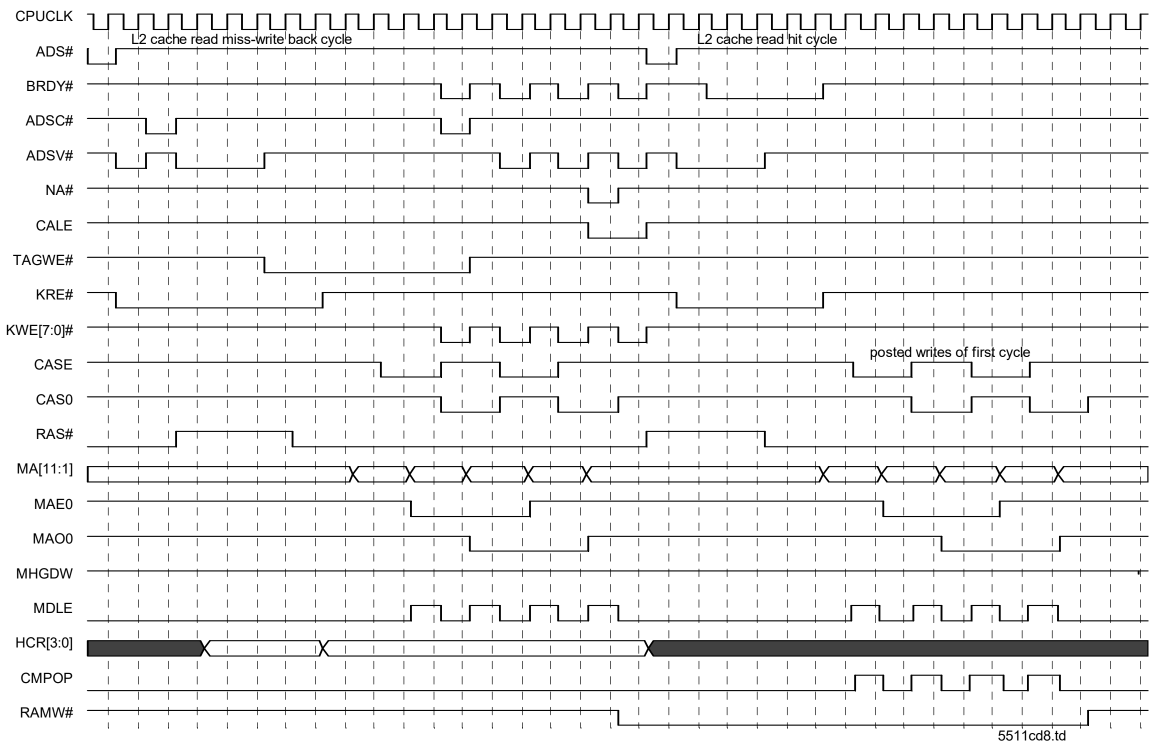


Figure 2.8

1. Asynchronous SRAM, A3 and A4 toggle mode
2. DRAM configuration, 5-2-2-2 for read and 5-2-2-2 for write
3. The first cycle is a L2 cache read miss-write back cycle, and the second cycle is a L2 read miss cycle and DRAM page hit.
4. The second cycle is postponed until DRAM write-back cycle is completed.

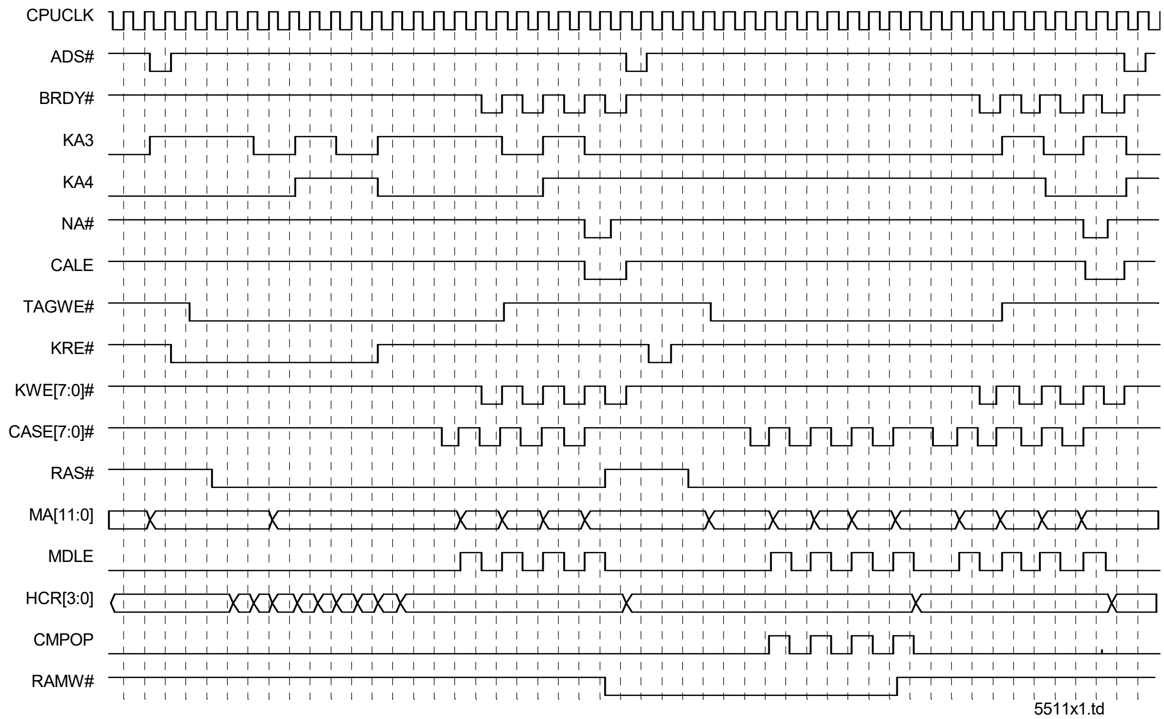


Figure 2.?

1. Asynchronous SRAM, A3 and A4 toggle mode
2. DRAM configuration, 5-2-2-2 for read and 5-2-2-2 for write
3. The first cycle is a L2 cache read miss updated cycle, and the second cycle is a L2 read miss cycle and DRAM page hit.

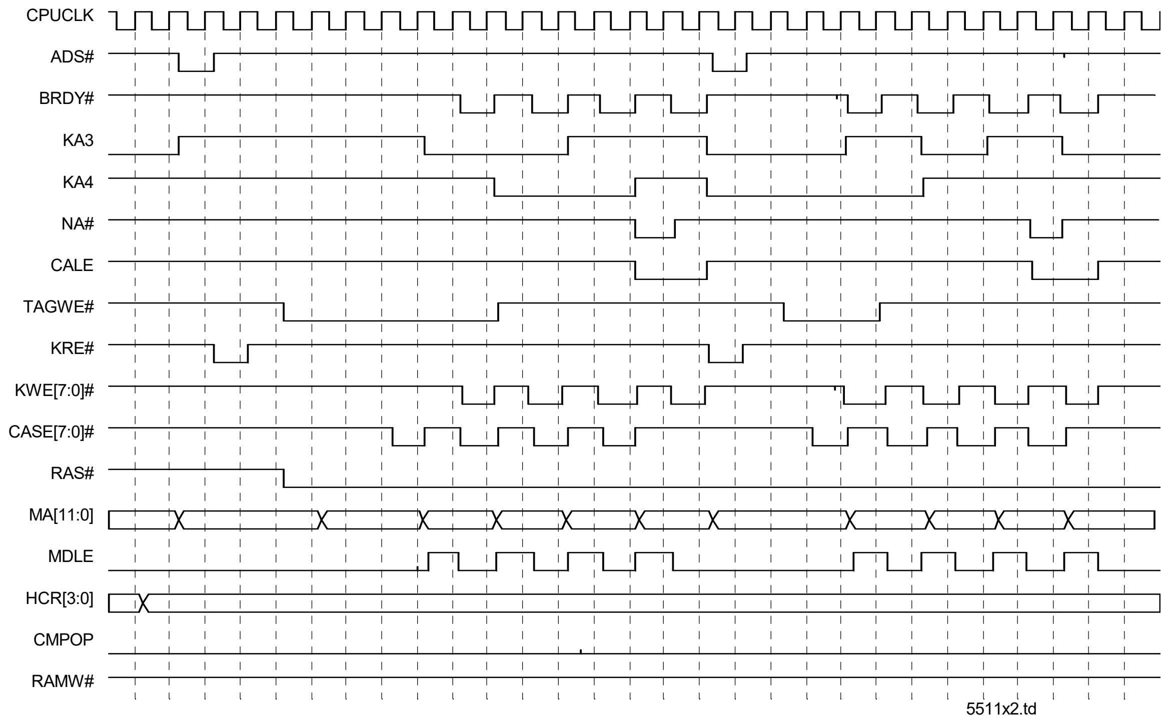


Figure 2.?

CPU Read PCI Cycle (Normal Termination)

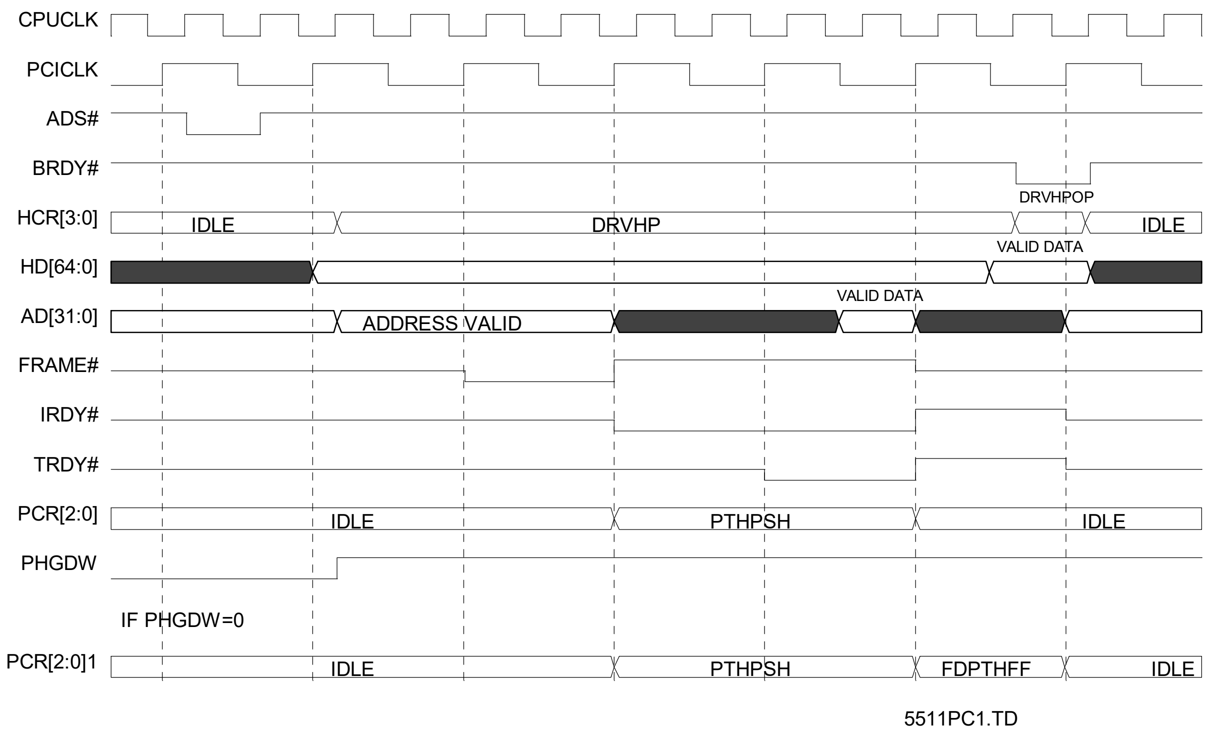


Figure 2.16

CPU Nonpost Write PCI Cycle (Normal Termination)

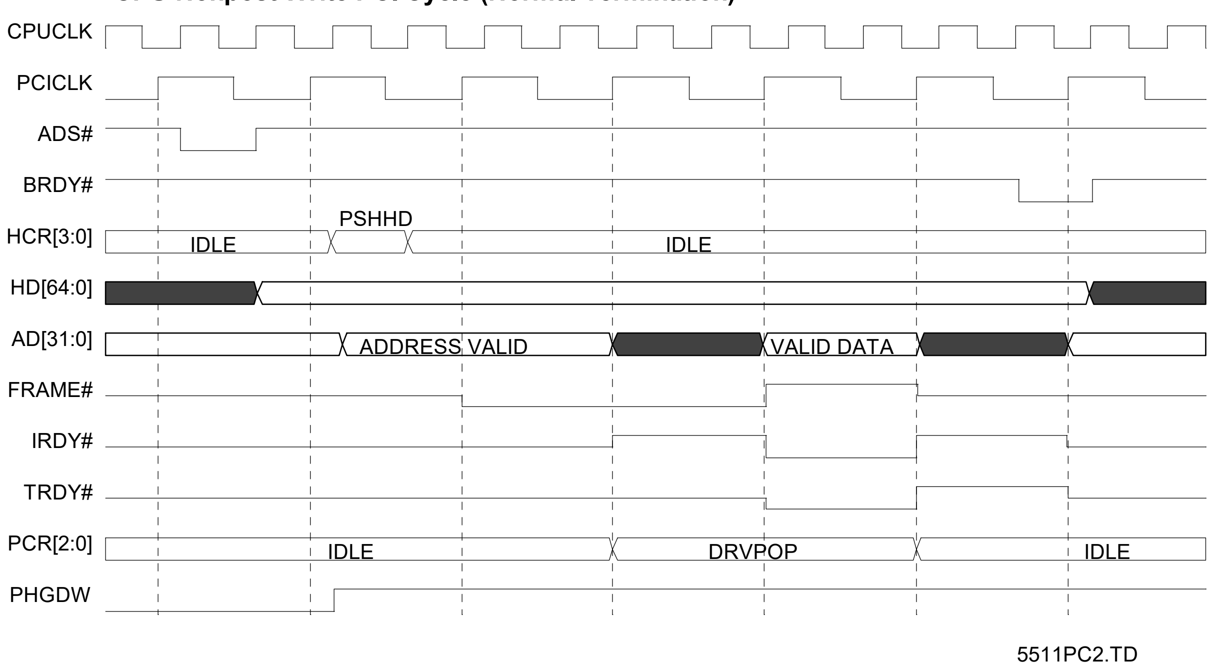


Figure 2.17

CPU Read PCI Cycle (Master Abort)

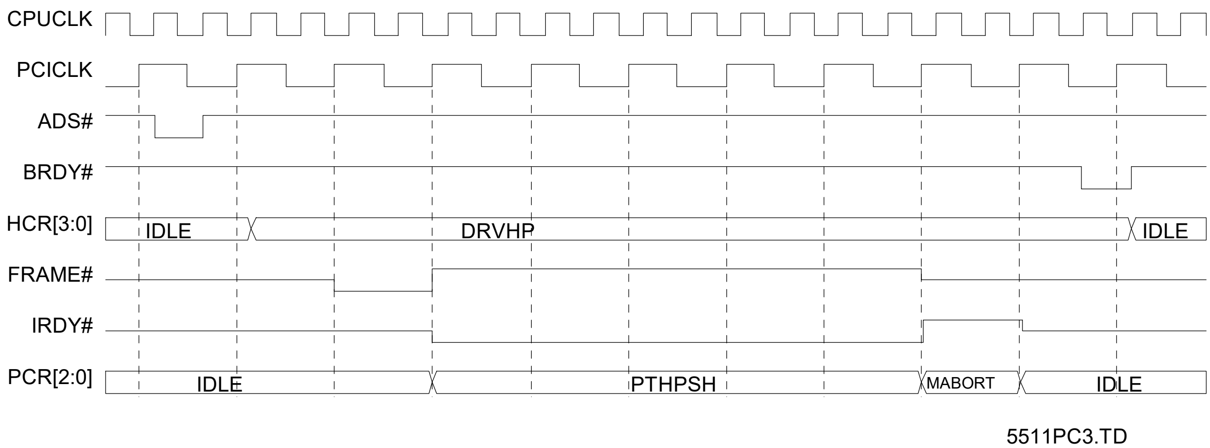


Figure 2.18

CPU Read PCI Cycle (Target Abort)

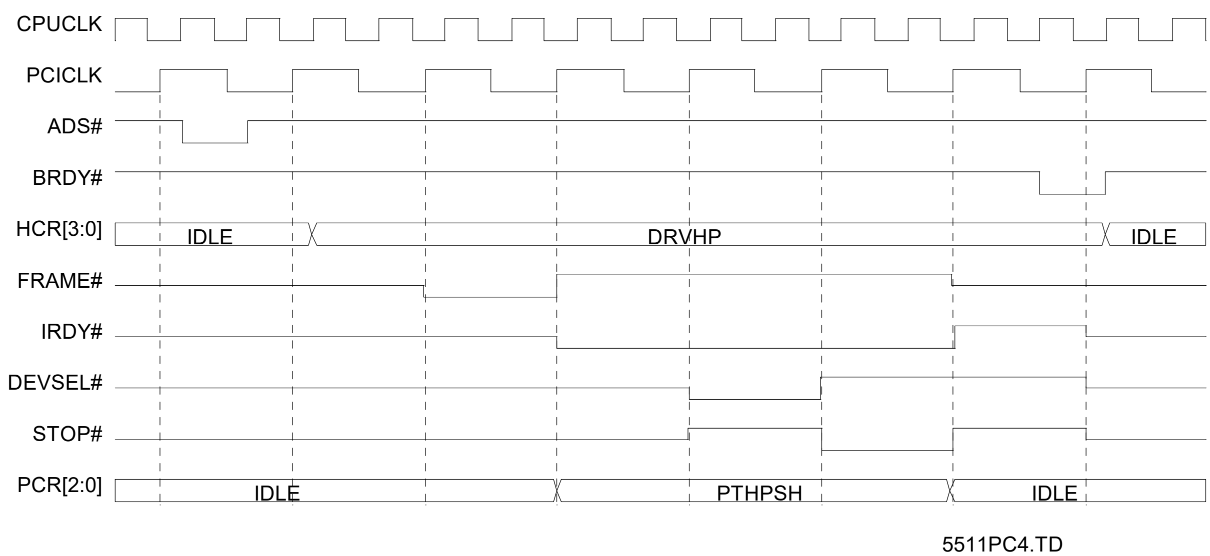


Figure 2.19

CPU Read Boff# Cycle

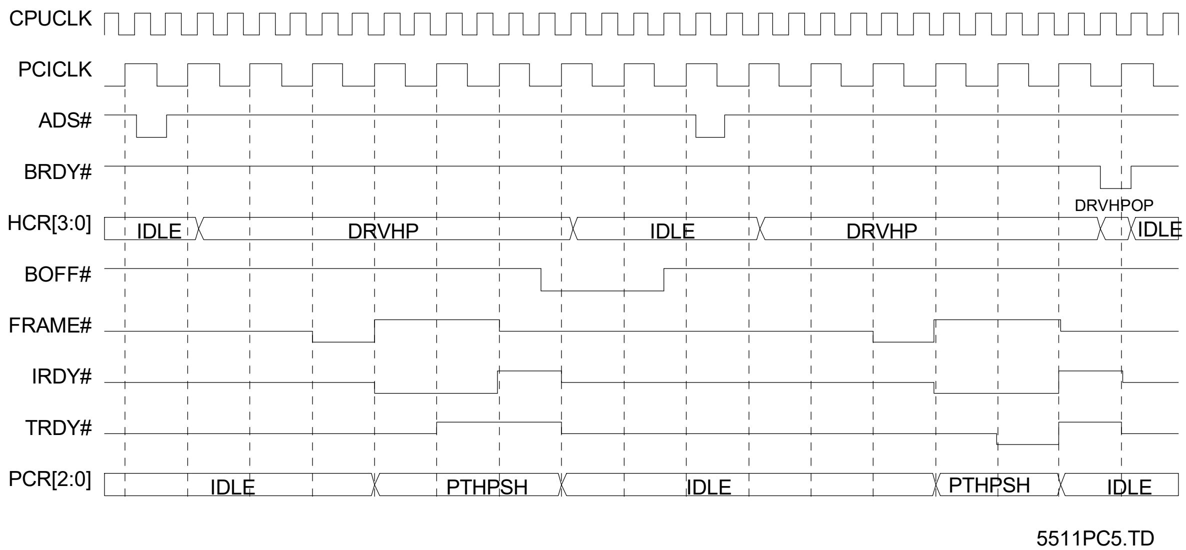


Figure 2.20

CPU Read IDE Cycle

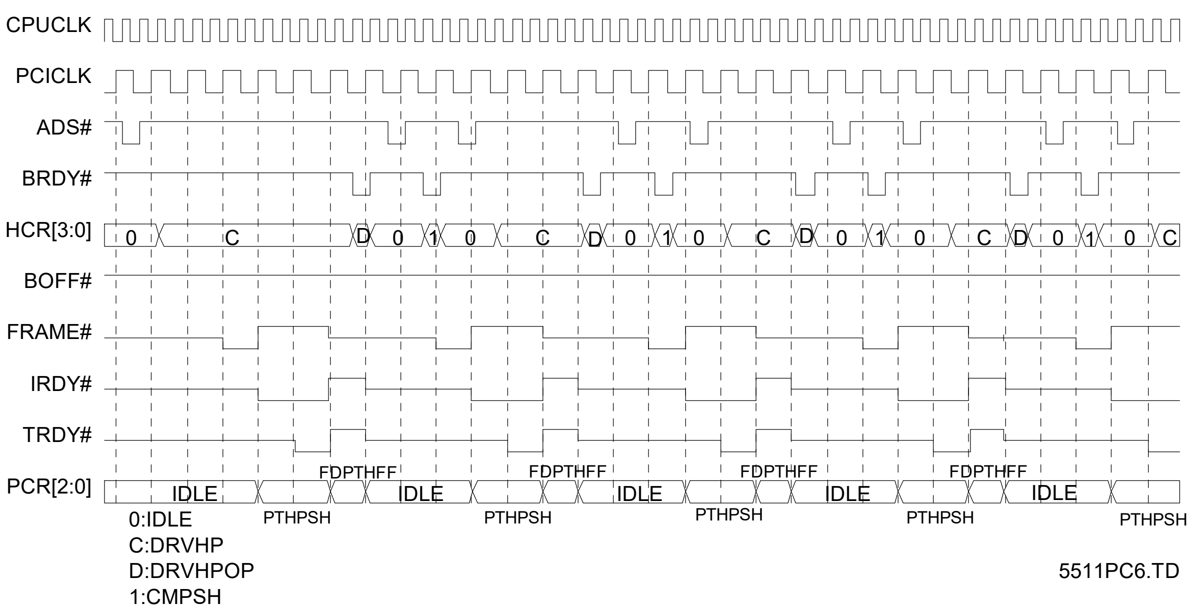


Figure 2.21

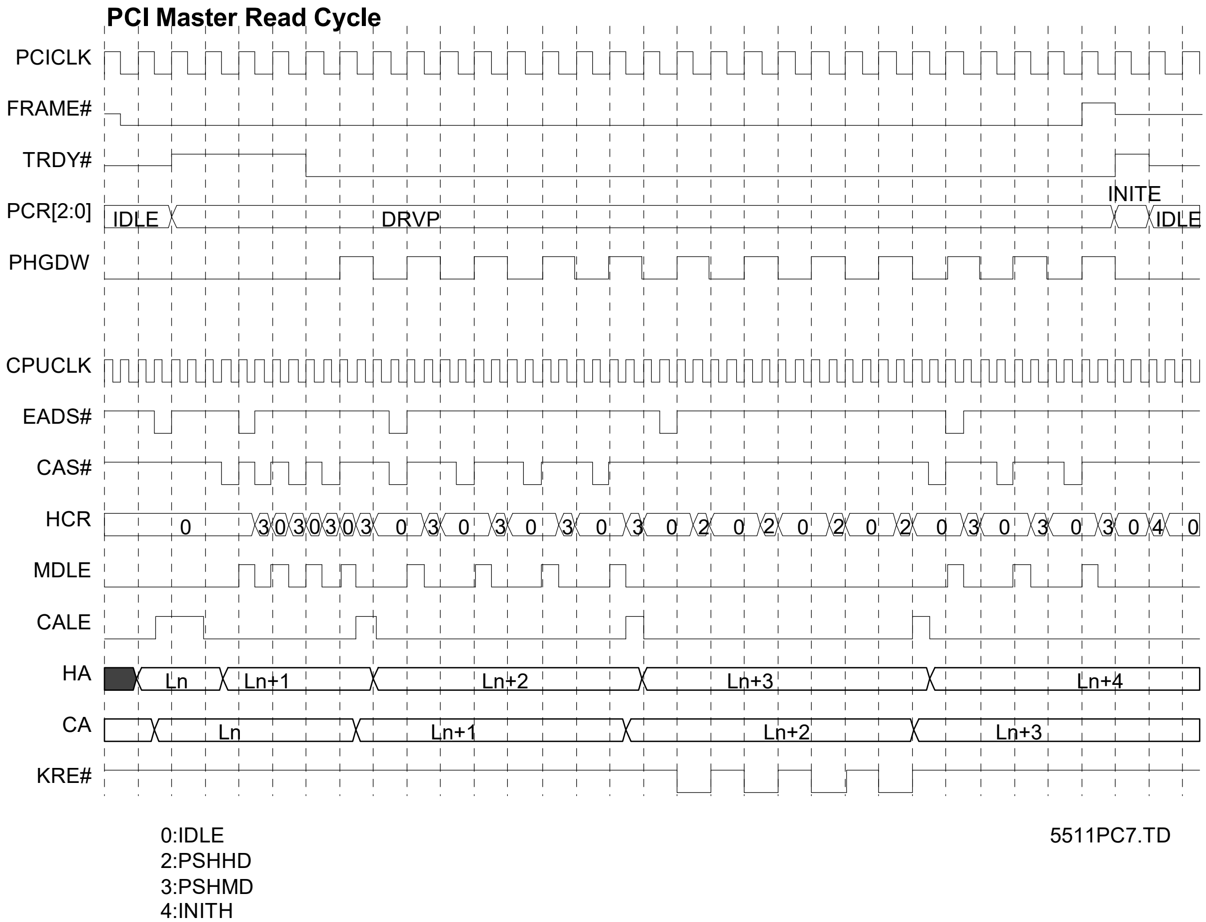

PCI Master Read Cycle
PCICLK
FRAME#
TRDY#
PCR[2:0]
IDLE
DRVP
INITE
IDLE
PHGDW
CPUCLK
EADS#
CAS#
HCR
MDLE
CALE
HA
Ln
Ln+1
Ln+2
Ln+3
Ln+4
CA
KRE#
0:IDLE
2:PSHHD
3:PSHMD
4:INITH
5511PC7.TD

Figure 2.22

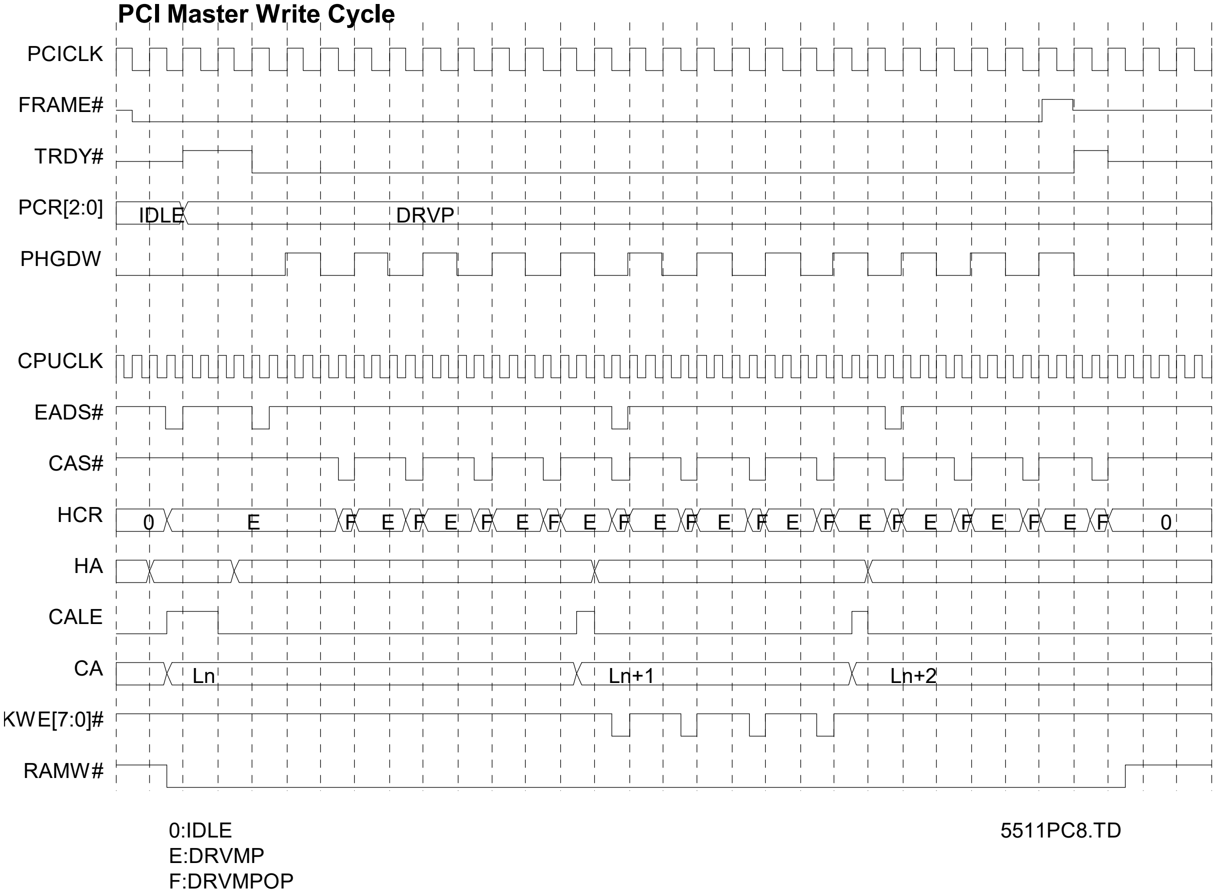


Figure 2.23

2.5 Electrical Characteristics

2.5.1 Absolute Maximum Ratings

Table 2-24 Absolute Maximum Ratings

Parameter	Min	Max	Unit
Ambient operating temperature	0	70	^oC
Storage temperature	-40	125	^oC
Input voltage	-0.3	5.5	V
Output voltage	-0.5	5.5	V
Power Dissipation		1	W

NOTE:

Stress above these listed may cause permanent damage to device. Functional operation of this device should be restricted to the conditions described under operating conditions.

2.5.2 DC Characteristics

Table 2-25 DC Characteristics

Ta= 0 - 70°C, Gnd= 0V, Vcc5= 5V±5%, Vcc3=3.3V±5%, Vcc35=3.3/5V±5%

Symbol	Parameter	Min	Max	Unit	Notes
V_{IL1}	Input Low Voltage	-0.3	0.8	V	Note 1, V_{CC3}=3.3V ±5%
V_{IH1}	Input High Voltage	2.2	V_{CC3}+0.3	V	Note 1
V_{IL2}	Input Low Voltage	-0.3	0.8	V	Note 2
V_{IH2}	Input High Voltage	2.2	V_{CC5}+0.3	V	Note 2
V_{T1-}	Schmitt Trigger Threshold Voltage Falling Edge	1.6		V	Note 3
V_{T1+}	Schmitt Trigger Threshold Voltage Rising Edge	3.2	V		Note 3
V_{H1}	Hysteresis Voltage	0.3	1.2	V	Note 3
V_{OL1}	Output Low Voltage		0.45	V	Note 4
V_{OH1}	Output High Voltage	2.4		V	Note 4
V_{OL2}	Output Low Voltage		0.4	V	Note 5
V_{OH2}	Output High Voltage	2.0	2.4	V	Note 5
V_{OL3}	Output Low Voltage		0.4	V	Note 6
V_{OH3}	Output High Voltage	2.0	V_{CC35}	V	Note 6
I_{OL1}	Output Low Current	8		mA	Note 7
I_{OH1}	Output High Current	-8		mA	Note 7
I_{OL2}	Output Low Current	4, 8		mA	Note 8, 12
I_{OH2}	Output High Current	-4,-8		mA	Note 8, 12
I_{OL3}	Output Low Current	16		mA	Note 9
I_{OH3}	Output High Current	-16		mA	Note 9
I_{OL4}	Output Low Current	8,12		mA	Note 10, 12
I_{OH4}	Output High Current	-8,-12		mA	Note 10, 12
I_{OL5}	Output Low Current	4		mA	Note 11
I_{OH5}	Output high Current	-4		mA	Note 11
I_{IH}	Input Leakage Current		-10	mA	
I_{IL}	Input Leakage Current		+10	mA	
C_{IN}	Input Capacitance		12	pF	Fc=1 Mhz
C_{OUT}	Output Capacitance		12	pF	Fc=1 Mhz
$C_{I/O}$	I/O Capacitance		12	pF	Fc=1 Mhz

NOTE:

1. V_{IL1} and V_{IH1} are applicable to the following signals: HA[31:3], W/R#, PAR, HBE[7:0]#, HITM#, D/C#, ADS#, CPUHLDA, SMIACT#, CACHE#, M/IO#
2. V_{IL2} and V_{IH2} are applicable to the following signals: TA[7:0], ALT, AD[31:0], C/BE[3:0]#, REQ[3:0]#, STOP#, DEVSEL#, TRDY#, IRDY#, FRAME#, LOCK#, PCICLK, PHLDA#, PHOLD#, WAKEUP[1:0], KBRST#, OSC, CPUCLK, ACLK.
3. V_{T1-}, V_{T1+} and V_{H1} are applicable to PWRGD
4. V_{OL1} and V_{OH1} are applicable to the following signals: TA[7:0], ALT, TAGWE#, MHGDW, CMPOP, MDLE, CPUHLDA, AD[31:0], GNT[3:0]#, STOP#, DEVSEL#, TRDY#, IRDY#, FRAME#, PAR, SERR#, PCIRST#, SMOUT, HCR[3:0], PCR[2:0]
5. V_{OL2} and V_{OH2} are applicable to the following signals: CALE, KA3/ADSC#, KA4/ADSV#, KWE[7:0]#, KRE#, STPCLK#, INIT, SMI#, HA[31:3], CPURST, W/R#, A20M#, EADS#, CPUHOLD, NA#, BRDY#, KEN#, BOFF#
6. V_{OL3} and V_{OH3} are applicable to the following signals: RAS[3:0]#, CASE[7:0]#, CASO[7:0]#, RAMW#, MA[11:0]
7. I_{OL1} and I_{OH1} are applicable to the following signals: TA[7:0], ALT, TAGWE#, MHGDW, CMPOP, MDLE, CPUHLDA, AD[31:0], C/BE[3:0]#, GNT[3:0]#, STOP#, DEVSEL#, TRDY#, FRAME#, SERR#, SMOUT, WAKEUP[1:0], CPURST, IRDY#
8. I_{OL2} and I_{OH2} are applicable to the following signals: CASE[7:0]#, CASO[7:0]#
9. I_{OL3} and I_{OH3} are applicable to the following signals: KA3/ADSC#, KA4/ADSV#, KWE[7:0]#
10. I_{OL4} and I_{OH4} are applicable to the following signals: RAS[3:0]#
11. I_{OL5} and I_{OH5} are applicable to the following signals: RAMW#, MA[11:0], HCR[3:0], CALE, KRE#, STPCLK#, INIT, SMI#, HA[31:3], W/R#, EADS#, CPUHOLD, NA#, BRDY#, KEN#, PAR, A20M#, PCIRST#, PHLDA#, PCR[2:0], BOFF#
12. The driving current of CASE[7:0]#, CASO[7:0]#, and RAS[3:0]# are programmed. Please refer to register description.

2.5.3 AC Characteristics

Table 2-26 AC Characteristics

Sym	Parameter	Typ	Max	Unit	CL
T1	BRDY# Active delay from CPUCLK	8	10	ns	35pf
T2	BRDY# Inactive delay from CPUCLK	9	12	ns	35pf
T3	KEN# Active delay from CPUCLK	6	8	ns	35pf
T4	KEN# Inactive delay from CPUCLK	6	8	ns	35pf
T5	NA# Active delay from CPUCLK	7	9	ns	35pf
T6	NA# Inactive delay from CPUCLK	8	10	ns	35pf
T7	CALE# Active delay from CPUCLK	9	11	ns	35pf
T8	CALE# Inactive delay from CPUCLK	7	9	ns	35pf
T9	EADS# Active delay from CPUCLK	6	8	ns	35pf
T10	EADS# Inactive delay from CPUCLK	6	8	ns	35pf
T11	CPUHOLD Active delay from CPUCLK	7	9	ns	35pf
T12	CPUHOLD Inactive delay from CPUCLK	6	8	ns	35pf
T13	CPURST Inactive delay from CPUCLK	6	8	ns	35pf
T14	CPURST High Pulse Width	33	43	cpuclk	35pf

T15	KRE# Active delay from CPUCLK (*1)	8	11	ns	90pf
T16	KRE# Inactive delay from CPUCLK (*1)	10	12	ns	90pf
T17	KWE[7:0]# Active delay from CPUCLK (*1)	7	9	ns	35pf
T18	KWE[7:0]# Inactive delay from CPUCLK (*1)	6	8	ns	35pf
T19	KWE[7:0]# Active delay from ACLK (*1)	5	7	ns	35pf
T20	KWE[7:0]# Inactive delay from ACLK (*1)	6	8	ns	35pf
T21	KA3 Low Valid delay from CPUCLK (*1)	10	13	ns	90pf
T22	KA3 High Valid delay from CPUCLK (*1)	9	12	ns	90pf
T23	KA4 Low Valid delay from CPUCLK (*1)	10	13	ns	90pf
T24	KA4 High Valid delay from CPUCLK (*1)	10	13	ns	90pf
T25	KA3 Low Valid delay from ACLK (*1)	8	10	ns	90pf
T26	KA3 High Valid delay from ACLK (*1)	8	10	ns	90pf
T27	KA4 Low Valid delay from ACLK (*1)	8	10	ns	90pf
T28	KA4 High Valid delay from ACLK (*1)	9	11	ns	90pf
T29	ADSC# Low Valid delay from CPUCLK (*1)	9	11	ns	90pf
T30	ADSC# High Valid delay from CPUCLK (*1)	10	12	ns	90pf
T31	ADSV# Low Valid delay from CPUCLK (*1)	8	10	ns	90pf
T32	ADSV# High Valid delay from CPUCLK (*1)	9	11	ns	90pf
T33	TAGWE# Active delay from CPUCLK	9	11	ns	35pf
T34	TAGWE# Inactive delay from CPUCLK	8	10	ns	35pf
T35	Tag Output Valid delay from CPUCLK when Hit Cycle	11	15	ns	35pf
T36	Tag Output Valid delay from CPUCLK in Update Cycle	13	17	ns	35pf
T37	ALT Output Valid delay from CPUCLK	11	14	ns	35pf
T38	RAS[3:0]# Active delay from CPUCLK	9	12	ns	150pf
T39	RAS[3:0]# Inactive delay from CPUCLK	8	10	ns	150pf
T40	CASE[7:0]#, CASO[7:0]# Active delay from CPUCLK	12	15	ns	90pf
T41	CASE[7:0]# CASO[7:0]# Inactive delay from CPUCLK	9	11	ns	90pf
T42	MA[11:0] Low Valid delay from CPUCLK	9	12	ns	35pf
T43	MA[11:0] High Valid delay from CPUCLK	9	12	ns	35pf
T44	MA[11:0] Propagation delay from A[27:3]	8	11	ns	35pf
T45	MDLE High Active delay from rising CPUCLK (*2)	6	8	ns	35pf
T46	MDLE High Inactive delay from rising CPUCLK (*2)	8	10	ns	35pf
T47	MDLE High Active delay from falling CPUCLK (*2)	6	8	ns	35pf
T48	MDLE High Inactive delay from falling CPUCLK (*2)	8	10	ns	35pf

T49	MHGDW High Active delay from rising CPUCLK (*3)	8	10	ns	35pf
T50	MHGDW High Inactive delay from rising CPUCLK (*3)	9	12	ns	35pf
T51	MHGDW High Active delay from falling CPUCLK (*3)	8	9	ns	35pf
T52	MHGDW High Inactive delay from falling CPUCLK (*3)	9	11	ns	35pf
T53	RAMW# Active delay from CPUCLK	9	11	ns	35pf
T54	RAMW# Inactive delay from CPUCLK	10	13	ns	35pf
T55	CMPOP Active delay from CPUCLK	7	9	ns	35pf
T56	CMPOP Inactive delay from CPUCLK	8	10	ns	35pf
T57	A20M# Active delay from CPUCLK	11	14	ns	35pf
T58	A20M# Inactive delay from CPUCLK	12	16	ns	35pf
T59	HCR[3:0] Active delay from CPUCLK	7	9	ns	35pf
T60	HCR[3:0] Inactive delay from CPUCLK	8	10	ns	35pf
T61	PCR[3:0] Active delay from PCICLK	6	8	ns	35pf
T62	PCR[3:0] Inactive delay from PCICLK	7	10	ns	35pf
T63	PHGDW Active delay from PCICLK in PCI Master Cycle	6	8	ns	35pf
T64	PHGDW Inactive delay from PCICLK in PCI Master Cycle	7	9	ns	35pf
T65	PHGDW Active delay from PCICLK in CPU Cycle	8	10	ns	35pf
T66	PHGDW Inactive delay from PCICLK in CPU Cycle	9	12	ns	35pf
T67	GNT[3:0]# Active delay from PCICLK	7	9	ns	50pf
T68	GNT[3:0]# Inactive delay from PCICLK	6	8	ns	50pf
T69	PHLDA# Active delay from PCICLK	7	9	ns	50pf
T70	PHOLD# Inactive delay from PCICLK	6	8	ns	50pf
T71	PAR Active delay from PCICLK	9	12	ns	50pf
T72	PAR Inactive delay from PCICLK	9	12	ns	50pf
T73	STPCLK# Active delay from PCICLK	10	15	ns	50pf
T74	STPCLK# Inactive delay from PCICLK	10	15	ns	50pf
T75	SMI# rise time to CPUCLK	8	10	ns	35pf
T76	SMI# fall time to CPUCLK	8	10	ns	35pf
T77	SMOUT Active delay from CPUCLK	10	15	ns	50pf
T78	AD[31:0], C/BE[3:0]# Active delay from PCICLK	9	11	ns	50pf
T79	AD[31:0], C/BE[3:0]# Inactive delay from PCICLK	9	11	ns	50pf
T80	FRAME# Active delay from PCICLK	8	10	ns	50pf

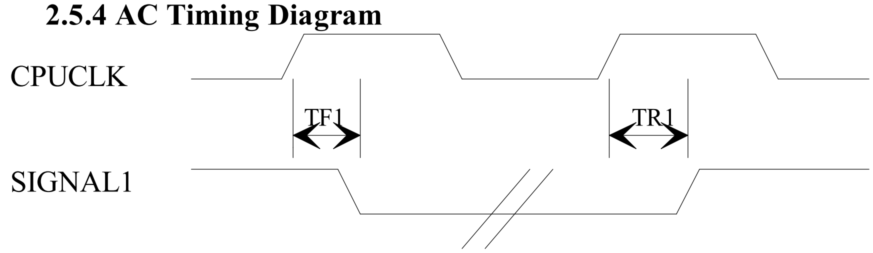
T81	FRAME# Inactive delay from PCICLK	6	8	ns	50pf
T82	IRDY# Active delay from PCICLK	8	10	ns	50pf
T83	IRDY# Inactive delay from PCICLK	6	8	ns	50pf
T84	TRDY# Active delay from PCICLK	9	11	ns	50pf
T85	TRDY# Inactive delay from PCICLK	7	9	ns	50pf
T86	DEVSEL# Active delay from PCICLK	8	10	ns	50pf
T87	DEVSEL# Inactive delay from PCICLK	6	8	ns	35pf
T88	STOP# Active delay from PCICLK	8	10	ns	35pf
T89	STOP# Inactive delay from PCICLK	7	9	ns	50pf
T90	BOFF# Active delay from CPUCLK	6	8	ns	50pf
T91	BOFF# Inactive delay from CPUCLK	7	8	ns	50pf
T92	INIT Active delay from CPUCLK	6	8	ns	35pf
T93	INIT Inactive delay from CPUCLK	6	8	ns	35pf
T94	HA[31:3] Output valid delay from CPUCLK	15	20	ns	50pf
T95	PCIRST# Active delay from CPUCLK	9	11	ns	50pf

*1: Please refer to Table 2-4 in Part II to check the relationship between these signals and CPUCLK/ACLK.

*2. The timing of MDLE depends on the configuration register's settings.

*3. In the case of writing EDO DRAM (CAS# pulse width = 1T), the MHGDW refers to the falling edges of CPUCLK. Otherwise, it refers to the rising edges of CPUCLK.

2.5.4 AC Timing Diagram

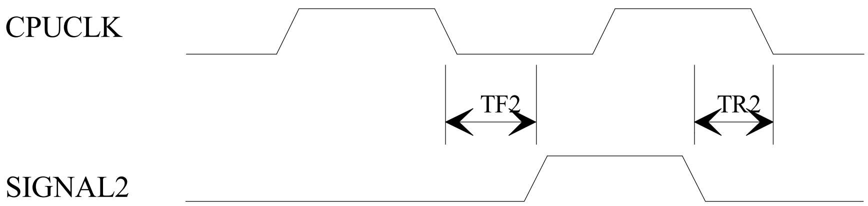


TF1 = T1, T3, T5, T9, T12, T13, T15, T17, T21, T23, T29, T31, T33, T35, T36, T37, T38, T40, T42, T46, T49, T53, T56, T57, T60, T76, T90, T93, T95

TR1 = T2, T4, T6, T8, T10, T11, T16, T18, T22, T24, T30, T32, T34, T39, T41, T43, T45, T50, T54, T55, T58, T59, T75, T77, T91, T92, T94

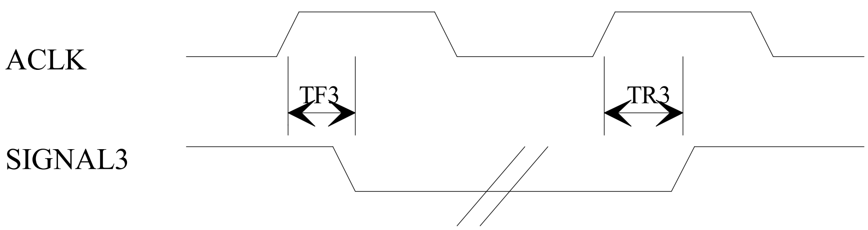
SIGNAL1 = BRDY#, KEN#, NA#, CALE#, EADS#, CPUHOLD, CPURST, KRE#, KWE[7:0]#, KA3, KA4, ADSC#, ADSV#, TAGWE#, TAG, ALT, RAS[3:0]#, CASE[7:0]#, CASO[7:0]#, MA[11:0], MDLE, MHGDW, RAMW#, CMPOP, A20M#, HCR[3:0], SMI#, SMOUT, BOFF#, INIT, HA[31:3], PCIRST#

Figure 2.24



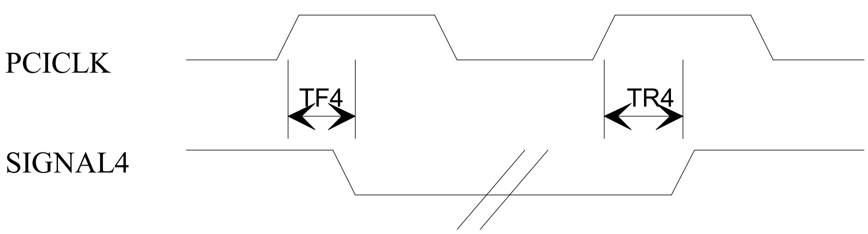
TF2 = T47, T51
TR2 = T48, T52
SIGNAL2 = MDLE, MHGDW

Figure 2.25



TF3 = T19, T25, T27
TR3 = T20, T26, T28
SIGNAL3 = KWE[7:0]#, KA3, KA4

Figure 2.26



TF4 = T62, T64, T66, T67, T69, T72, T73, T78, T80, T82, T84, T86, T88
TR4 = T61, T63, T65, T68, T70, T71, T74, T79, T81, T83, T85, T87, T89
SIGNAL4 = PCR[3:0], PHGDW, GNT[3:0]#, PHLDA#, PAR, STPCLK#, AD[31:0], C/BE[3:0]#, FRAME#, IRDY#, TRDY#, DEVSEL#, STOP#

Figure 2.27

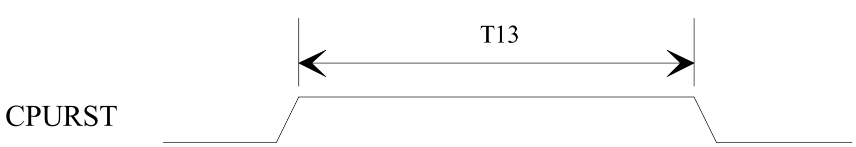


Figure 2.28

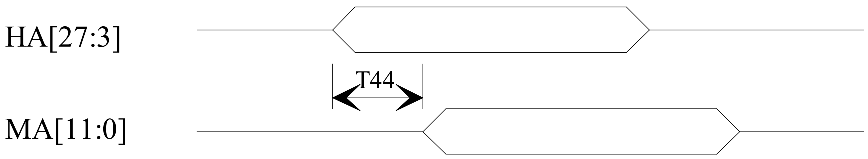


Figure 2.29

3. SiS5512 PCI Local Data Buffer (PLDB)

3.1 SiS5512 Overview

3.1.1 Features

- **Supports the Full 64-bit Pentium Processor data Bus**
- **Provides a 64/32 bit Interface to DRAM Memory**
- **Provides a 32-bit Interface to PCI**
- **Three Integrated 4 QW Deep FIFO, CTMFF, CTPFF, and PTHFF to Increase System Performance**
  - -1 level CPU-to-Memory Posted Write Buffer (CTMFF) with 4 Qw Deep
  - -4 level CPU-to-PCI Posted Write Buffer(CTPFF) with 4 Dw Deep
  - -1 level CPU-to-PCI IDE Read Prefetch Buffer(PTHFF) with 1 Dw Deep
  - -1 level PCI-to-Memory Posted Write Buffer(PTHFF) with 4 Qw Deep
  - -1 level PCI-to-Memory Read Prefetch Buffer(CTPFF) with 4 Qw Deep
- **Always Sustains 0 Wait Performance on CPU-to-Memory.**
- **Always Streams 0 Wait Performance on PCI-to/from-Memory Access**
- **Built-in one 32-bit General Purpose Register**
- **Includes an 8-bit IPI Vector Register to Support SLiC Interrupt Dispatcher**
- **Provides Parity Generation for Memory Writes**
- **Provides Optional Parity Checker for Memory Reads**
- **208-Pin PQFP**
- **0.6 um CMOS Technology**

3.1.2 Functional Block Diagram

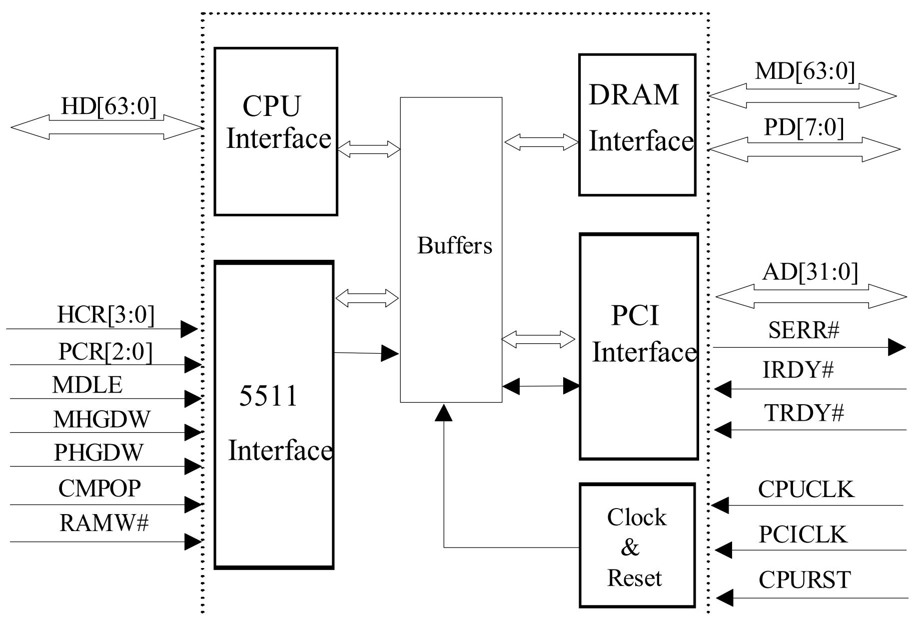


Figure 3.1 SiS5512 Functional Block Diagram

3.2 Functional Description

The SiS5512 PCI Local Data Buffer (PLDB) provides a bi-directional data buffering among the 64-bit Host Data Bus, the 64/32-bit Memory Data Bus, and the 32-bit PCI Address/Data bus. The PLDB incorporates three FIFOs and one read buffer among the bridges of the CPU, PCI, and memory buses. This buffering scheme smoothes the differences in access latencies and bandwidths among three buses, therefore improves the overall system performance. A four level/4Dws deep write buffer (CTPFF) provides buffering on CPU write to PCI bus. A one level/4Qws deep write buffer(CTMFF) is used for buffering write data from CPU to memory. A one level/4Qws deep read prefetch buffer(CTPFF) is used to buffer read data from L2/DRAM for PCI master read cycle. A one level/4Qws deep write buffer(PTHFF) is used for buffering writ data from PCI to memory. A one level/1Dw deep buffer(PTHFF) is allocated for IDE read prefeching. In CPU read DRAM cycle, a one Qw read buffer (CTMRB) is used to latch the DRAM data onto host bus. During bus operation between the Host, PCI, and Memory, the PLDB receives control signals from the PCMC, performs functions such as latching data, forwarding data to destination bus, data assemble and disassemble. Besides, an 8 bit register is reserved for IPI vector to support two processors system.
Figure x shows the PLDB block diagram.

3.2.1 Functional Overview

The SiS5512 mainly contains storage elements. Its behavior is always controlled by SiS5511. The following paragraph will explain Host bus control commands and PCI bus control commands in detail, the other control signals, please refer to pin description.

NOTE: The data structure of SiS5512 is a circular queue. A queue is an order list in which all insertions take place at the rear end, while all deletions take place at the front end. The first element which is inserted into the queue will be the first one to be removed, so queue is also known as First In First Out (FIFO) lists.
There are some terminology used in the explanation of control commands. They are explained as follows.
Front element: the first element to be moved out of buffer.
Front pointer: used to point out the next moved out element.
Rear element: the last element has been pushed into the buffer..
Rear pointer: used to point to the location for next pushed in element.

3.2.2 Host Bus Control Commands

Table 3-1 HCR[3:0]:Host bus control commands

0000	Idle	HD is input to 5512.
0001	Cmpsh	Push HD into the rear element of the CTMFF on the CPUCLK rising edge that the code is sampled active for buffering CPU write data to DRAM. The rear pointer is also forwarded on the same clock rising edge.
0010	Pshhd	Push HD into the rear element of CTPFF on the CPUCLK rising edge that the code is sampled active, for buffering CPU write data to PCI bus, or for buffering data to PCI bus in PCI master read L2. The rear pointer is forwarded on the same rising edge.

0011	Pshmd	Push MD into the rear element of CTPFF on the CPUCLK rising edge that the code is sampled active, for buffering the prefetch data in PCI master read DRAM. On the same CPUCLK rising edge, 5512 also forwards the rear pointer of the CTPFF.
0100	Inith	Initialize CTPFF push tracker for PCI master read cycles. 5511 always issues the command by the end of a PCI master read cycles to initialize the CTPFF rear pointer to flush the prefetching data. Actually, the rear pointer is adjusted to address the first entry of the CTPFF on the CPUCLK rising edge that the code is sampled active.
0101	Winv	CPU Writes IPI 8-bit register. The IPI register is updated on the CPUCLK rising edge that the code is sampled asserted.
0110	Wgreg	Write General Purpose Register in PLDB. This register can be accessed through R/W 5511 configuration register port 90-93h. Specifically, bit 31 of the register is used to enable/disable parity checker in PLDB. By default, the value is logic 0, and disable parity checking. Note that 5512 only supports 64-bit parity checking. BIOS should not turn on the bit when 32-bit DRAM is employed. The rest of this 32 bit register can be used as a general purpose register.
0111	Fshpop	Forward read pointer of IDE prefetch buffer(PTHFF). 5511 issues the command when that an non IDE read data cycle occurs when IDE prefetch data is valid.
1000	Drvhm	This code is issued in CPU read DRAM cycle to combinatorially enable 5512 output the CTMRB data onto the host bus.
1100	Drvhp	This code is issued in CPU read PCI cycle to combinatorially enable 5512 output PTHFF data onto the host bus.
1101	Drvhpop	This code is issued at the same time that 5511 returns BRDY# to CPU in CPU read PCI cycles. 5512 will forward the front pointer of the PTHFF on the CPUCLK rising edge that the code is sampled active.
1011	Reserved.	
1110	Drvmp	This code is issued in PCI master write L2/DRAM to combinatorially enable 5512 output the front element of the PTHFF onto the host bus.
1111	Drvmpop	PCI write L2 or DRAM and pop; 5511 generates the code to notify 5512 to forward the PTHFF front pointer on the CPUCLK rising edge that the code is sampled active. The code is always asserted for 1 CPUCLK, and also keeps driving the host data bus.
1001	Rinv	CPU read IPI register; This code combinatorially selects the IPI register data onto the HD bus.
1010	Rgreg	Read General Purpose Register(Port 90-93h in 5511 cnfg. space) This code combinatorially selects the IPI register data onto the HD bus.

3.2.3 PCI Bus Control Commands

Table 3-2 PCR[2:0]: PCI bus control commands

000	Idle	AD is input to 5512
001	Pthpsh	When the code is asserted, 5512 pushes AD into the rear entry of the PTHFF on the PCICLK rising edge that IRDY# and TRDY# is sampled active, in CPU Read PCI or PCI master write cycles. Whether the data is pushed into the low or high Dw of the entry depends on the logic of PHGDW. Moreover, the rear pointer of the PTHFF is forwarded on the PCICLK rising edge that IRDY# and TRDY# are asserted and PHGDW is logic 1.
010	Fdpthff	Forward the rear pointer of PTHFF to treat the last Dw pushed into PTHFF is low Dw. In this case, 5512 does not forward the rear pointer because PHGDW is logic 0. Hence, we add this code to control the PTHFF push tracker to forward pointer.
011	Initp	Initialize CTPFF Pop tracker. 5512 initializes the front pointer to address the first entry of the CTPFF on the rising edge of PCICLK that the code is sampled. This action is always taken by the end of a PCI master read cycles.
100	Drvp	When the code is asserted, 5512 keeps driving the current front element of the CTPFF onto AD bus. Whether low or high Dw of the front entry is driven depends on the logic of PHGDW. 5512 also forwards the front pointer of the CTPFF on the PCICLK rising edge that IRDY# and TRDY# are sampled active and PHGDW is logic high. 5511 generates this code only in the PCI master read cycles.
101	Drvpop	5512 treats the code in a similar manner that it does Drvp except that 5512 forwards the front pointer on the PCICLK rising edge that IRDY# and TRDY# are sampled asserted, independent of PHGDW logic. 5511 issues the code only in CPU post or nonpost PCI cycles.
110	Fdhtpff	Forward the front pointer of CTPFF to treat master abort, target abort, or retry encountered in CPU write PCI cycles to discard data.
111	Mabort	Push FFFF into the rear entry of PTHFF on the rising edge of PCICLK that the code is sampled active. 5511 generates the code when it detects master abort in CPU read PCI cycles.

Table 3-3 Address Flow and Data Flow of Basic Cycles

Cycles	Address Flow	Data Flow
1. CPU/R/PCI	HAÜ5511ÜAD	ADÜ5512ÜHD
2. CPU/W/PCI	HAÜ5511ÜAD	HDÜ5512ÜAD
3. CPU/R/ISA	HAÜ5511ÜADÜ5513Ü LA,SA	SDÜ5513ÜADÜ5512ÜHD
4. CPU/W/ISA	HAÜ5511ÜADÜ5513Ü LA,SA	HDÜ5512ÜADÜ5513ÜSD
5. CPU/R/DRAM	HAÜ5511ÜMA	MDÜ5512ÜHD
6. CPU/W/DRAM	HAÜ5511ÜMA	HDÜ5512ÜMD
7. CPU/R/L2	Independent	Independent
8. CPU/W/L2	Independent	Independent
9. CPU/R/PCI(master abort)	HAÜ5511ÜAD	5512ÜHD
10. PCI/R/L2	ADÜ5511ÜHA	HDÜ5512ÜAD

11. PCI/W/L2	ADÜ5511ÜHA	ADÜ5512ÜHD
12. PCI/R/DRAM	ADÜ5511ÜHA	MDÜ5512ÜAD
13. PCI/W/DRAM	ADÜ5511ÜHA	ADÜ5512ÜMD
14. ISA/R/L2	LA,SAÜ5513ÜADÜ5511Ü HA	HDÜ5512ÜADÜ5513ÜSD
15. ISA/W/L2	LA,SAÜ5513ÜADÜ5511Ü HA	SDÜ5513ÜADÜ5512ÜHD
16. DMA/R/L2	5513ÜADÜ5511ÜHA, 5513ÜLA,SA	HDÜ5512ÜADÜ5513ÜSD
17. DMA/W/L2	5513ÜADÜ5511ÜHA, 5513ÜLA,SA	SDÜ5513ÜADÜ5512ÜHD
18. ISA/R/DRAM	LA,SAÜ5513ÜADÜ5511Ü MA	MDÜ5512ÜADÜ5513ÜSD
19. ISA/W/DRAM	LA,SAÜ5513ÜADÜ5511Ü MA	SDÜ5513ÜADÜ5512ÜMD
20. DMA/R/DRAM	5513ÜADÜ5511ÜMA, 5513ÜLA,SA	MDÜ5512ÜADÜ5513ÜSD
21. DMA/W/DRAM	5513ÜADÜ5511ÜMA, 5513ÜLA,SA	SDÜ5513ÜADÜ5512ÜMD
22. ISA Refresh	5513ÜSA	

3.3 Register Description

3.3.1 SLiC Memory Mapped Register

IPI Vector Register

(1) Address: XFC00008h.
(2) 8-bit register.
(3) This register contains the vector which is associated with interprocessor interrupts (IPIs).

3.3.2 General Purpose Register

The 32-bit general purpose register can be accessed via read or write 5511 configuration space with index (90~93h). The bit 31 is used as parity check enable bit. By default, this bit is 0 and disables parity check. The bit 30 is used to define the 32-bit and 64-bit DRAM organization. When it is set to 0, the organization is 64-bit and the default value of bit 30 is 0.

Register 93, default is 0

Bit 7 **Disable/enable parity**

1: to enable

0: to disable

Bit 6 **32 bit DRAM bank support**

1: to enable 32 bit DRAM bank

0: to disable 32 bit DRAM bank

if it supports 32/64 bit mixed auto bank, it's always enabled.

Bits 4:1 **internal MDLE latch delay time adjust**

0000: 0 ns

0001: 1 ns

0010: 2ns

~~~

0100: 4 ns

~~~

1000: 8 ns

~~~~

Default setting is 0111, those bits are only worked at 32/64 bit auto bank

Bit 0 **Select 32/64 bit auto bank**

0: to disable 32/64 bit auto bank, but it's compatible with 5512A

1: to enable 32/64 bit auto bank

3.4 Pin Assignment and Description

3.4.1 Pin Assignment
~~~~

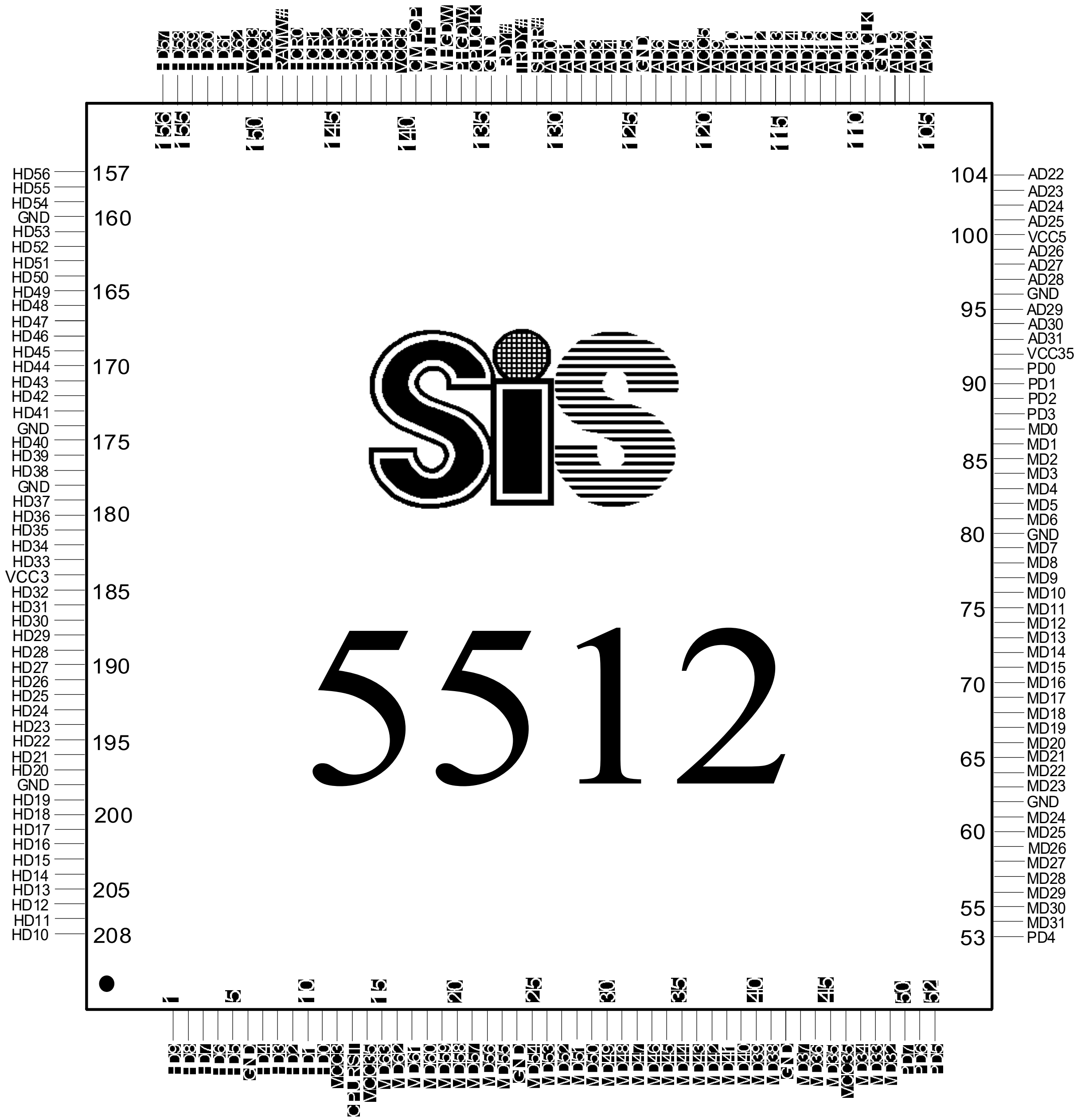


Figure 3.2

3.4.2 Pin Listing (# means active low)

Table 3-4

1=HD9	3V	48=MD33	3V/5V	95=AD29	5V
2=HD8	3V	49=MD32	3V/5V	96=GND	VSS
3=HD7	3V	50=PD7	3V/5V	97=AD28	5V
4=HD6	3V	51=PD6	3V/5V	98=AD27	5V
5=HD5	3V	52=PD5	3V/5V	99=AD26	5V
6=GND	VSS	53=PD4	3V/5V	100=VCC5	5V
7=HD4	3V	54=MD31	3V/5V	101=AD25	5V
8=HD3	3V	55=MD30	3V/5V	102=AD24	5V
9=HD2	3V	56=MD29	3V/5V	103=AD23	5V

10=HD1	3V	57=MD28	3V/5V	104=AD22	5V
11=HD0	3V	58=MD27	3V/5V	105=AD21	5V
12=VCC3	3V	59=MD26	3V/5V	106=AD20	5V
13=CPURST	3V	60=MD25	3V/5V	107=AD19	5V
14=VCC35	3V/5V	61=MD24	3V/5V	108=GND	VSS
15=MD63	3V/5V	62=GND	VSS	109=PCICLK	5V
16=MD62	3V/5V	63=MD23	3V/5V	110=AD18	5V
17=MD61	3V/5V	64=MD22	3V/5V	111=AD17	5V
18=MD60	3V/5V	65=MD21	3V/5V	112=AD16	5V
19=MD59	3V/5V	66=MD20	3V/5V	113=AD15	5V
20=MD58	3V/5V	67=MD19	3V/5V	114=AD14	5V
21=MD57	3V/5V	68=MD18	3V/5V	115=AD13	5V
22=MD56	3V/5V	69=MD17	3V/5V	116=AD12	5V
23=MD55	3V/5V	70=MD16	3V/5V	117=AD11	5V
24=GND	VSS	71=MD15	3V/5V	118=AD10	5V
25=MD54	3V/5V	72=MD14	3V/5V	119=AD9	5V
26=MD53	3V/5V	73=MD13	3V/5V	120=VCC5	5V
27=MD52	3V/5V	74=MD12	3V/5V	121=AD8	5V
28=MD51	3V/5V	75=MD11	3V/5V	122=AD7	5V
29=MD50	3V/5V	76=MD10	3V/5V	123=AD6	5V
30=MD49	3V/5V	77=MD9	3V/5V	124=GND	VSS
31=MD48	3V/5V	78=MD8	3V/5V	125=AD5	5V
32=MD47	3V/5V	79=MD7	3V/5V	126=AD4	5V
33=MD46	3V/5V	80=GND	VSS	127=AD3	5V
34=MD45	3V/5V	81=MD6	3V/5V	128=AD2	5V
35=MD44	3V/5V	82=MD5	3V/5V	129=AD1	5V
36=MD43	3V/5V	83=MD4	3V/5V	130=AD0	5V
37=MD42	3V/5V	84=MD3	3V/5V	131=SERR#	5V
38=MD41	3V/5V	85=MD2	3V/5V	132=TRDY#	5V
39=MD40	3V/5V	86=MD1	3V/5V	133=IRDY#	5V
40=MD39	3V/5V	87=MD0	3V/5V	134=GND	VSS
41=MD38	3V/5V	88=PD3	3V/5V	135=CPUCLK	5V
42=GND	VSS	89=PD2	3V/5V	136=PHGDW	5V
43=MD37	3V/5V	90=PD1	3V/5V	137=MHGDW	5V
44=MD36	3V/5V	91=PD0	3V/5V	138=MDLE	5V
45=MD35	3V/5V	92=VCC35	3V/5V	139=CMPOP	5V
46=VCC35	3V/5V	93=AD31	5V	140=VCC5	5V
47=MD34	3V/5V	94=AD30	5V	141=PCR2	5V

142=PCR1	5V	176=HD39	3V
143=PCR0	5V	177=HD38	3V
144=HCR3	5V	178=GND	VSS
145=HCR2	5V	179=HD37	3V
146=HCR1	5V	180=HD36	3V
147=HCR0	5V	181=HD35	3V
148=RAMW#	5V	182=HD34	3V
149=HD63	3V	183=HD33	3V
150=VCC3	3V	184=VCC3	3V
151=HD62	3V	185=HD32	3V
152=HD61	3V	186=HD31	3V
153=HD60	3V	187=HD30	3V

154=HD59	3V	188=HD29	3V
155=HD58	3V	189=HD28	3V
156=HD57	3V	190=HD27	3V
157=HD56	3V	191=HD26	3V
158=HD55	3V	192=HD25	3V
159=HD54	3V	193=HD24	3V
160=GND	VSS	194=HD23	3V
161=HD53	3V	195=HD22	3V
162=HD52	3V	196=HD21	3V
163=HD51	3V	197=HD20	3V
164=HD50	3V	198=GND	VSS
165=HD49	3V	199=HD19	3V
166=HD48	3V	200=HD18	3V
167=HD47	3V	201=HD17	3V
168=HD46	3V	202=HD16	3V
169=HD45	3V	203=HD15	3V
170=HD44	3V	204=HD14	3V
171=HD43	3V	205=HD13	3V
172=HD42	3V	206=HD12	3V
173=HD41	3V	207=HD11	3V
174=GND	VSS	208=HD10	3V
175=HD40	3V		

3.4.3 Pin Description

Table 3-5

Pin No.	Symbol	Typ	Function
149, 151-159, 161-173, 175-177, 179-183, 185-197, 199-208, 1-11	HD[63:0]	I/O	CPU data bus.
15-23, 25-41, 43-45, 47-49, 54-61, 63-79, 81-87,	MD[63:0]	I/O	Memory data bus.
93-95, 97-99, 101-107, 110-119, 121-123, 125-130,	AD[31:0]	I/O	PCI address/data bus.
50-53, 88-91	PD[7:0]	I/O	Parity bit bus.
144-147	HCR[3:0]	I	Host data bus control signals.

141-143	PCR[2:0]	I	PCI data bus control signals.
133	IRDY#	I	PCI IRDY#
132	TRDY#	I	PCI TRDY#
131	SERR#	OD	System error is an open drain output for reporting parity error
138	MDLE	I	Memory Data Read Latch Enable.
137	MHGDW	I	Memory High Double Word Indicator. When high, the high DW of the rear element of the CTMFF or PTHFF is driven onto the Low Dw of the MD bus.
136	PHGDW	I	PCI High Double Word Indicator
139	CMPOP	I	When this signal is sampled active on CPUCLK rising edge, the rear pointer of the CTMFF is forwarded.
148	RAMW#	I	DRAM Write Enable.
13	CPURST	I	CPU Reset.
135	CPUCLK	I	CPU Clock.
109	PCICLK	I	PCI Bus Clock.
100, 120, 140	VCC5		+5V DC power
12, 150, 184	VCC3		+3.3V DC power
14, 46, 92	VCC35		These power pins can be connected to +5V or +3V power which depend on the DRAM type.
6, 24, 42, 62, 80, 96, 108, 124, 134, 160, 174, 178, 198	GND		Ground

3.5 Electrical Characteristics

3.5.1 Absolute Maximum Ratings

Table 3-6

Parameter	Min	Max	Unit
Ambient operating temperature	0	70	^{0}C
Storage temperature	-40	125	^{0}C
Input voltage	-0.3	5.5	V
Output voltage	-0.5	5.5	V
Power dissipation		1	W

Stress above these listed may cause permanent damage to device. Functional operation of this device should be restricted to the conditions described under operating conditions.

3.5.2 DC Characteristics

Table 3-7

TA=0 - 70 ^{0}C, VSS=0V, VCC3=3.3+/-5%, VCC5=5.0+/-5%, VCC35=3.3/5V±5%

Symbol	Parameter	Min	Max	Unit	Condition
V_{IL1}	Input Low Voltage	-0.3	0.8	V	Note1
V_{IH1}	Input High Voltage	2.2	VCC3+0.3	V	Note1
V_{IL2}	Input Low Voltage	-0.3	0.8	V	Note2
V_{IH2}	Input High Voltage	2.2	VCC5+0.3	V	Note2
V_{IL3}	Input Low Voltage	-0.3	0.8	V	Note3
V_{IH3}	Input High Voltage	2.2	VCC35+0.3	V	Note3
V_{OL1}	Input Low Voltage		0.45	V	Note4
V_{OH1}	Output High Voltage	2.4		V	Note4
V_{OL2}	Output Low Voltage		0.4	V	Note5
V_{OH2}	Output High Voltage	2.0	VCC3	V	Note5
V_{OL3}	Output Low Voltage		0.4	V	Note6
V_{OH3}	Output High Voltage	2.0	VCC35	V	Note6
I_{OL1}	Output Low Current	4		mA	Note7
I_{OH1}	Output High Current	-4		mA	Note7
I_{IH}	Input Leakage Current		-10	mA	
I_{IL}	Input Leakage Current		+10	mA	
C_{IN}	Input Capacitance		12	pF	Fc=1MHz

C_{OUT}	Output Capacitance		12	pF	Fc=1MHz
$C_{I/O}$	I/O Capacitance		12	pF	Fc=1MHz

NOTE:

1. V_{IL1} and V_{IH1} are applicable to HD[63:0].
2. V_{IL2} and V_{IH2} are applicable to RAMW#, AD[31:0], CPURST, CPUCLK, HCR[3:0], PCR[2:0], MDLE, CMPOP, IRDY#, TRDY#, MHGDW, PHGDW, and PCICLK.
3. V_{IL3} and V_{IH3} are applicable to MD[63:0], and PD[7:0].
4. V_{OL1} and V_{OH1} are applicable to AD[31:0], and SERR#.
5. V_{OL2} and V_{OH2} are applicable to HD[63:0].
6. V_{OL3} and V_{OH3} are applicable to MD[63:0] and PD[7:0]
7. I_{OL1} and I_{OH1} are applicable to HD[63:0], MD[63:0], AD[31:0], PD[7:0], and SERR#.

3.5.3 AC Characteristics

All of the signal-loading are based on AD with 50 pF, MD with 90 pF, and HD with 50 pF, except the Notes to specify the different test condition. AD, HD, MD, PD, HCR, PCR are the abbreviations for AD[31:0], HD[63:0], MD[63:0], PD[7:0], HCR[3:0] and PCR[2:0].

Table 3-8

Sym	Parameter	min	typ	max	Fig	Notes
t1	MD setup time from MDLE falling	2			3.1	
t2	MD hold time from MDLE falling	2			3.1	
t3	HD data valid delay from MD data valid		11.5		3.1	
t4	HD setup time from CPUCLK rising	2			3.5	
t5	HD hold time from CPUCLK rising	2			3.5	
t6	HCR setup time from CPUCLK rising	2			3.5	
t7	HCR hold time from CPUCLK rising	2			3.5	
t8	MD data valid delay from CPUCLK rising		12		3.2	
t8a	MD data valid delay from CPUCLK rising		17		3.2	180pF
t9	MD data valid delay from CPUCLK rising that CMPOP is asserted		13		3.4	
t9a	MD data valid delay from CPUCLK rising that CMPOP is asserted		18		3.4	180pF
t10	MD data output delay from the assertion of RAMW#		10.5		3.3	
t10a	MD data output delay from the assertion of RAMW#		15		3.3	180pF
t11	PD data valid delay from CPUCLK rising		14		3.2	
t11a	PD data valid delay from CPUCLK rising		19		3.2	180pF

t12	PD data valid delay from CPUCLK rising that CMPOP is asserted		15		3.4	
t12a	PD data valid delay from CPUCLK rising that CMPOP is asserted		20		3.4	180pF
t13	PD data output delay from asserted RAMW#		10		3.3	
t13a	PD data output delay from asserted RAMW#		14.5		3.3	180pF
t14	CMPOP setup time from CPUCLK rising	2			3.4	
t15	CMPOP hold time from CPUCLK rising	2			3.4	
t16	MD setup time from CPUCLK rising	2			3.10	
t17	MD hold time from CPUCLK rising	2			3.10	
t18	AD data valid delay from CPUCLK rising		10		3.5	
t19	PCR setup time from PCICLK rising	2			3.6, 3.11	
t20	PCR hold time from PCICLK rising	2			3.6, 3.11	
t21	AD data valid delay from PCICLK rising that PCR is asserted without toggling PHGDW		11		3.6, 3.11	
t22	AD output data hold time from PCICLK rising that PCR is asserted without toggling PHGDW		9		3.6, 3.11	
t23	AD setup time from PCICLK rising that IRDY# and TRDY# are asserted.	2			3.8	
t24	AD hold time from PCICLK rising that IRDY# and TRDY# are asserted.	2			3.8	
t25	IRDY# setup time from PCICLK rising	3			3.8, 3.11	
t26	IRDY# hold time from PCICLK rising	0			3.8, 3.11	
t27	TRDY# setup time from PCICLK rising	3			3.8, 3.11	
t28	TRDY# hold time from PCICLK rising	0			3.8, 3.11	
t29	HD output hold time from PCICLK rising that IRDY# and TRDY# are asserted		13.5		3.8	
t30	HD data valid delay from PCICLK rising that IRDY# and TRDY# are asserted		15		3.8	
t31	MD output hold time from PCICLK rising that IRDY# and TRDY# are asserted		12.5		3.8	

t31a	MD output hold time from PCICLK rising that IRDY# and TRDY# are asserted		17.5		3.8	180pF
t32	MD data valid delay from PCICLK rising that IRDY# and TRDY# are asserted		13.5		3.8	
t32a	MD data valid delay from PCICLK rising that IRDY# and TRDY# are asserted		18.5		3.8	180pF
t33	HD output hold time from CPUCLK rising that the code of HCR is DRVMPOP		11		3.9	
t34	HD data valid delay from CPUCLK rising that the code of HCR is DRVMPOP		13		3.9	
t35	MD output hold time from CPUCLK rinsing that the code of HCR is DRVMPOP		9		3.9	
t35a	MD output hold time from CPUCLK rinsing that the code of HCR is DRVMPOP		13		3.9	180pF
t36	MD data valid delay from CPUCLK rising that the code of HCR is DRVMPOP		12.5		3.9	
t36a	MD data valid delay from CPUCLK rising that the code of HCR is DRVMPOP		17.5		3.9	180pF
t37	AD output hold time due to toggling PHGDW		5.5		3.6	
t38	AD data valid delay due to toggling PHGDW		8		3.6	
t39	MHGDW setup time from MDLE falling	2			3.1	
t40	MHGDW hold time from MDLE falling	2			3.1	
t41	MD output hold time due to toggling MHGDW (only in 32-bit single SIMM DRAM path)		9		3.2	
t42	MD data valid delay due to toggling MHGDW (only in 32-bit single SIMM DRAM path)		14		3.2	
t43	PD output hold time due to toggling MHGDW (only in 32-bit single SIMM DRAM path)		9		3.2	
t44	PD data valid delay due to toggling MHGDW (only in 32-bit single SIMM DRAM path)		14		3.2	

t45	HD output float delay from the negative of HCR3		7.5		3.7	

3.5.4 AC Timing Diagram

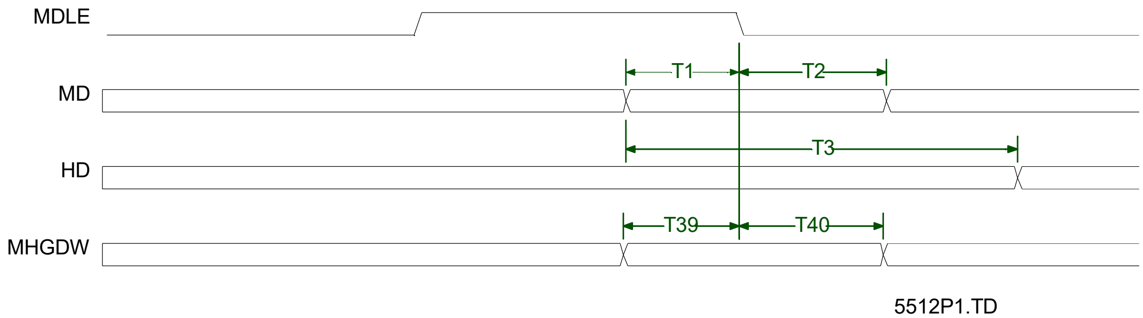


Figure 3.3 CPU Read DRAM Cycle

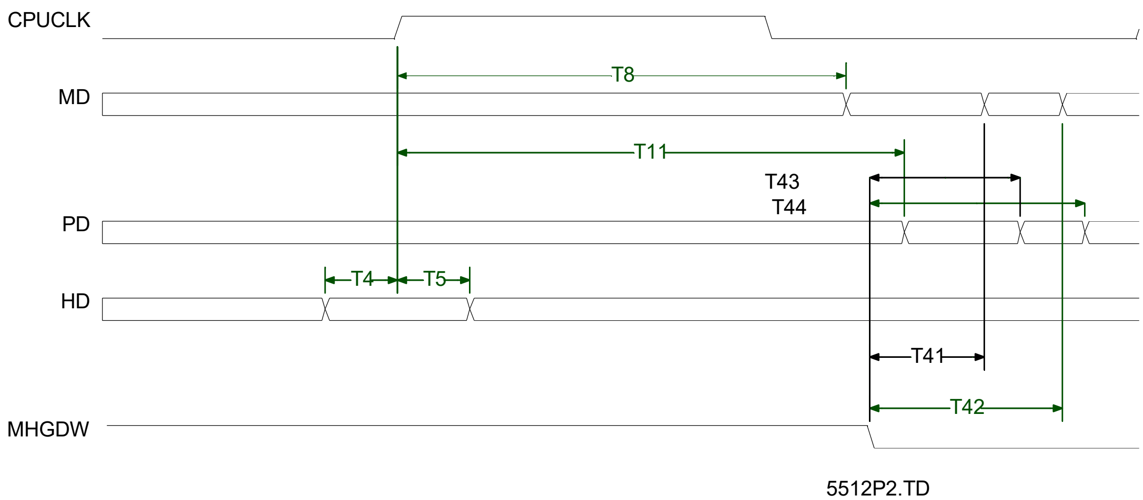


Figure 3.4 CPU Write DRAM Cycle

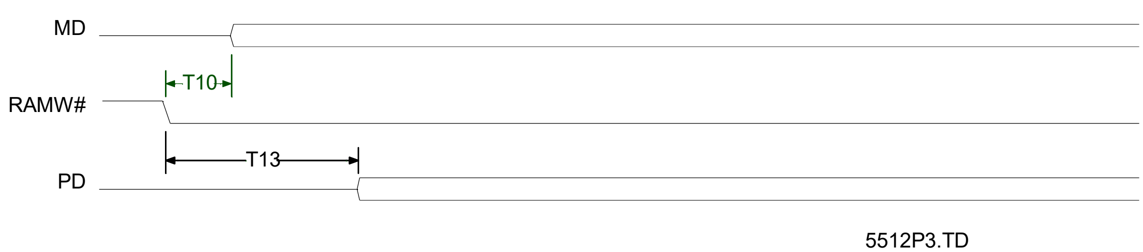
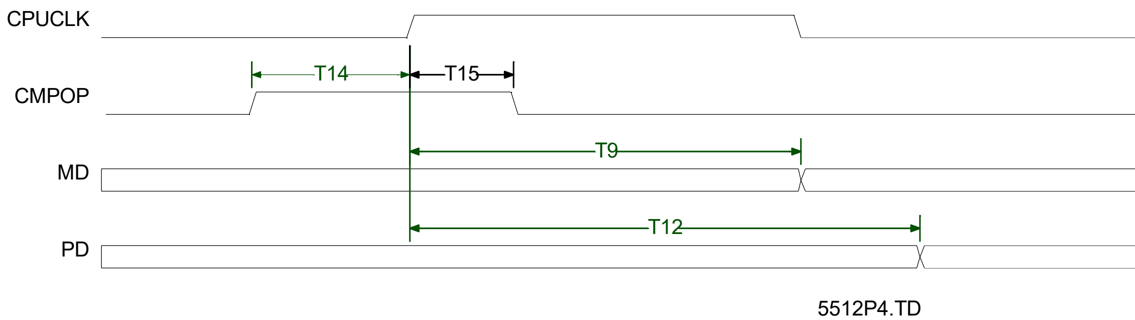
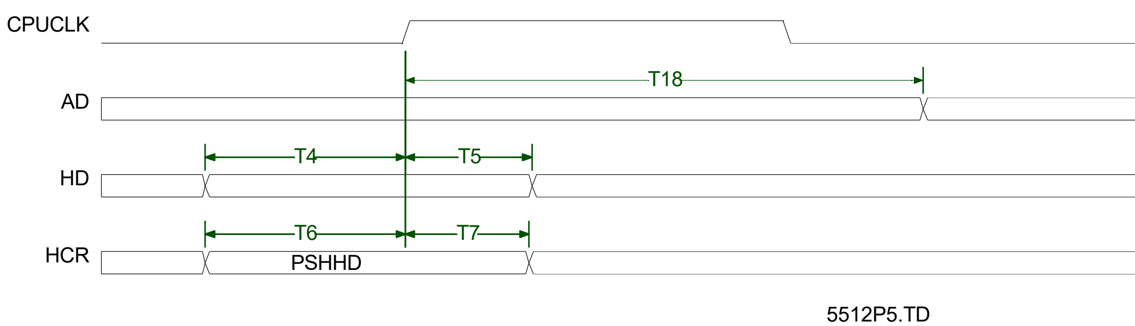


Figure 3.5 MD Output Delay From RAMW# Asserted



NOTE: RAMW# is asserted.

Figure 3.6 Write DRAM Cycle with CMPOP Asserted



NOTE: The code of PCR is DRVP.

Figure 3.7 CPU Write PCI Cycle

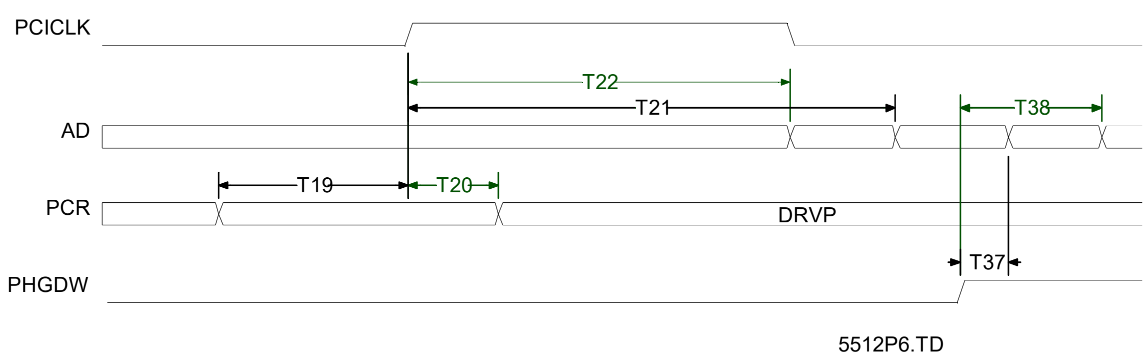
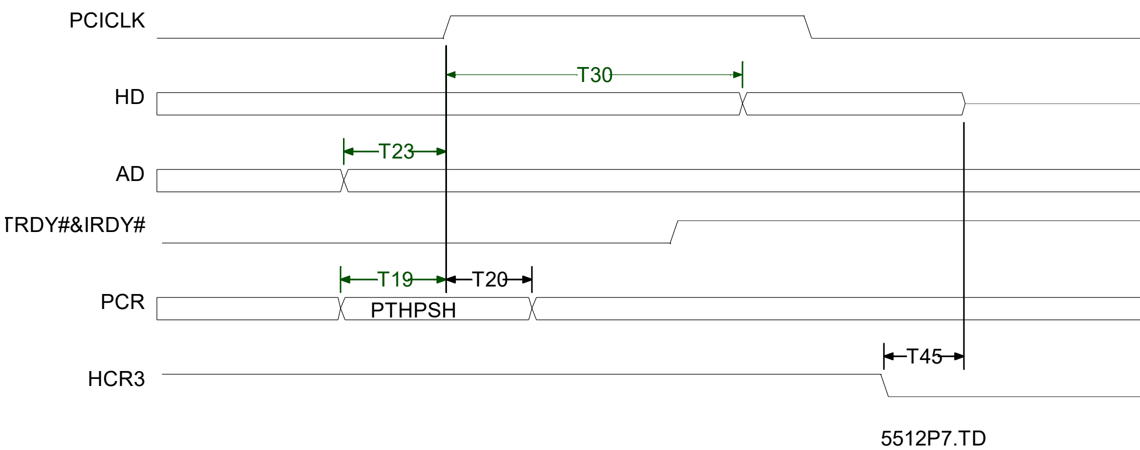


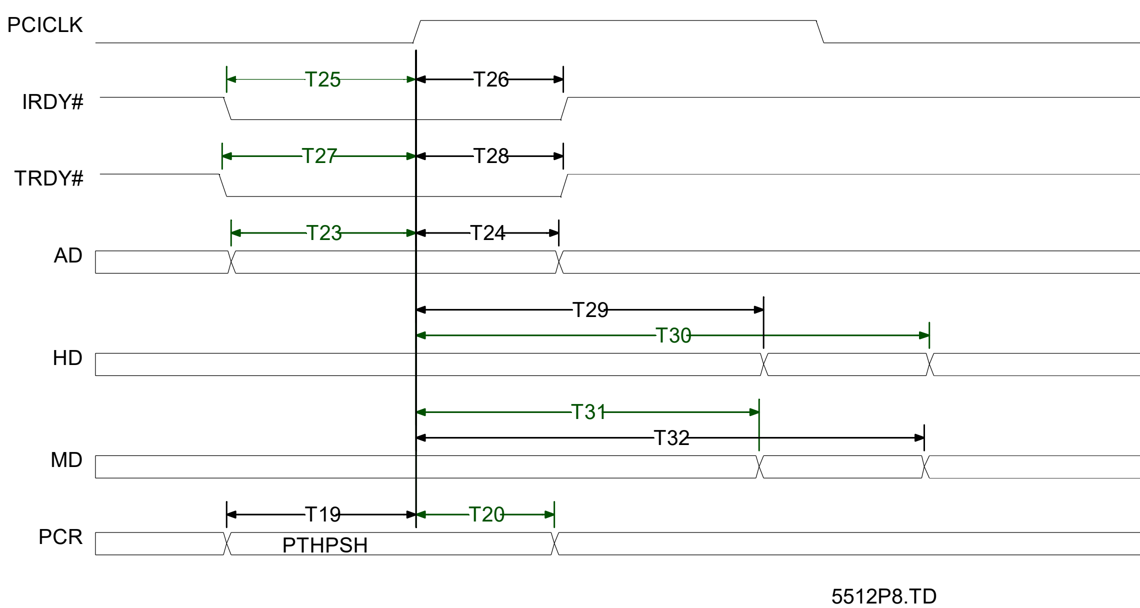
Figure 3.8 CPU Post Write PCI Cycle



NOT

E: The code of HCR is DRVHP, and the code of PCR is PTHPSH.

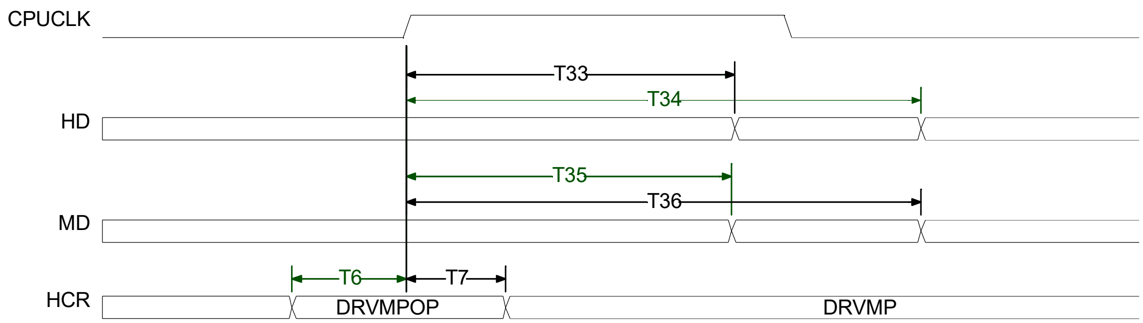
Figure 3.9 CPU Read PCI Cycle



NOT

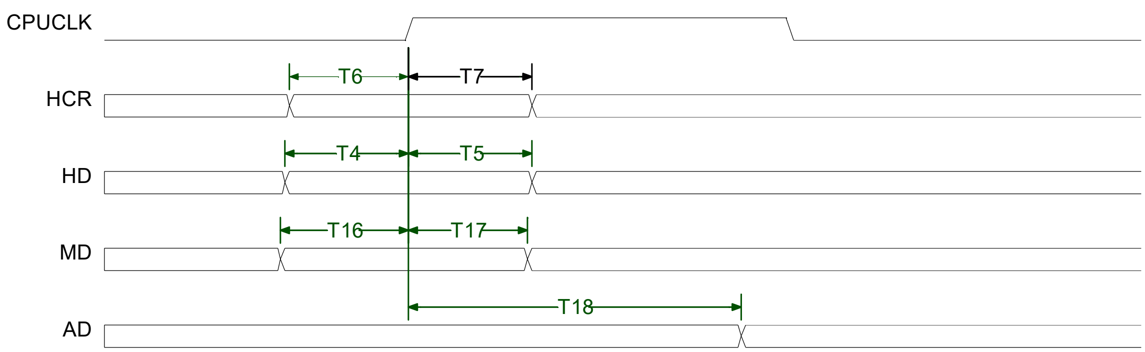
E: RAMW# is asserted.

Figure 3.10 PCI Master Write Cycle



5512P9.TD

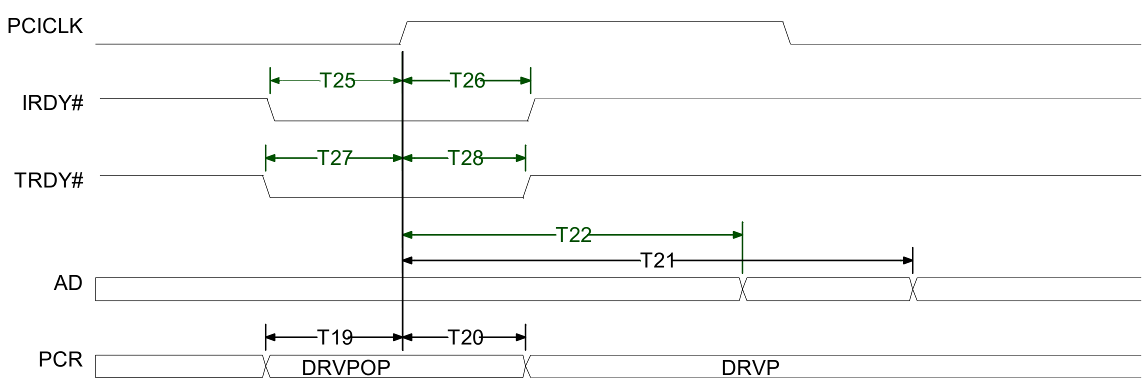
Figure 3.11 PCI Master Write Cycle



5512P10.TD

NOTE: The code of HCR is PSHMD when read MD and MDLE is asserted. The code of HCR is PSHHD when read L2

Figure 3.12 PCI Master Read Cycle



5512P11.TD

NOT

E: The code of PCR is DRVP.

Figure 3.13 PCI Master Read Cycle

4. SiS5513 PCI System I/O (PSIO)

4.1 SiS5513 Overview

4.1.1 Features

- **Integrated Bridge Between PCI Bus and ISA Bus**

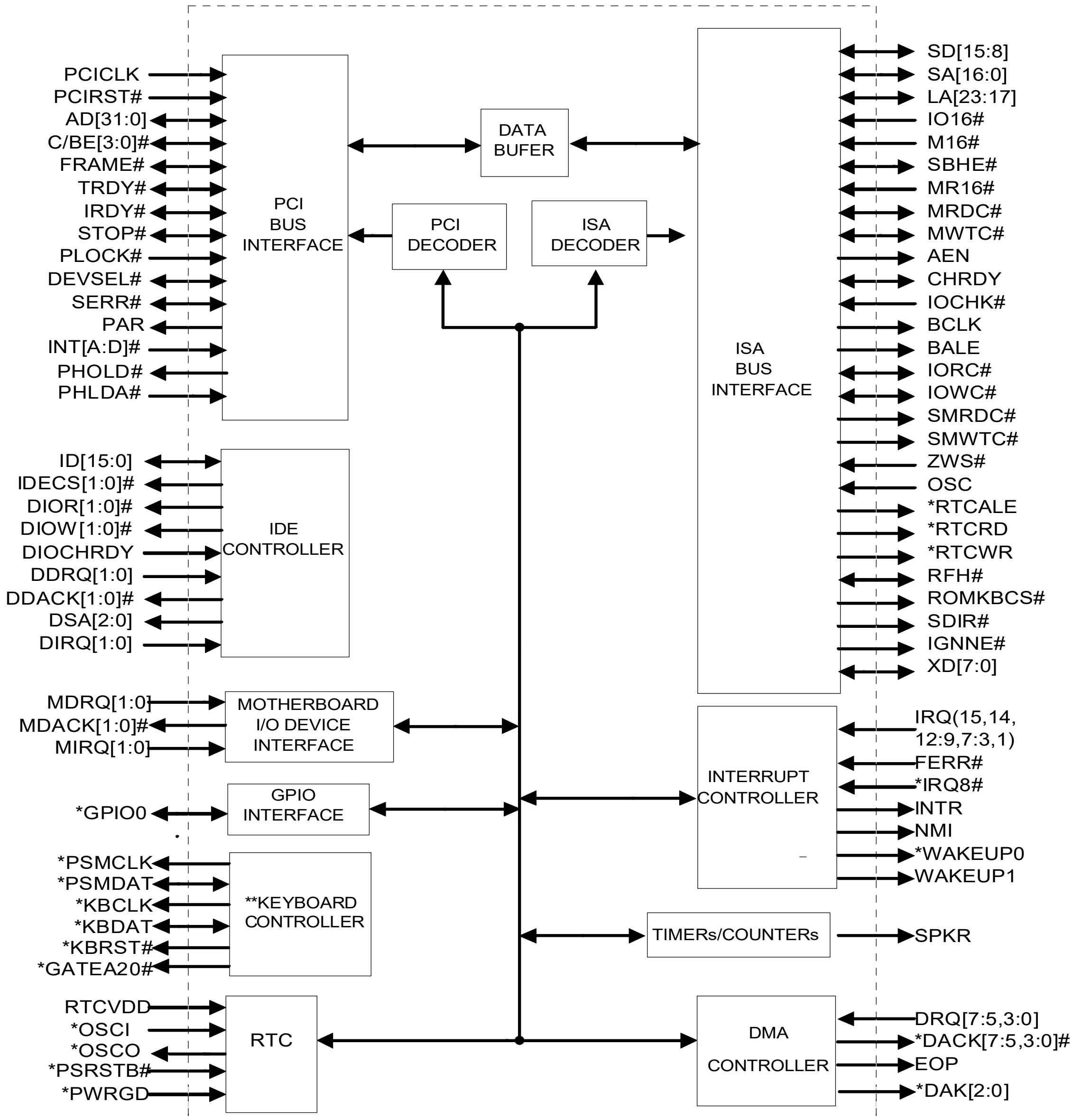
- Translates PCI Bus Cycles into ISA Bus Cycles
- Translates ISA Master or DMA Cycles into PCI Bus Cycles
- Provides PCI-to-ISA Memory one DoubleWord Posted Write Buffer

- **Integrated ISA Bus Compatible Logic**

- ISA Bus Controller
- ISA Arbiter for ISA Master, DMA Devices, and Refresh
- Built-in Two 8237 Compatible DMA Controllers
- Built-in Two 8259A Compatible Interrupt Controllers
- Built-in One 8254 Timer

- **Supports Reroutibility of four PCI Interrupts to Any Unused IRQ Interrupt**
- **Supports Flash ROM**
- **Built-in RTC with 256 Bytes CMOS SRAM**
- **Built-in Keyboard Controller (in the future)**
- **Built-in PCI Master/Slave IDE Controller**
  - Fully compatible with PCI Local Bus Specification V2.1
  - Supports PCI Bus Mastering
  - Plug and Play Compatible
  - Supports Scatter and Gather
  - Supports Dual Mode Operation, Native Mode and Compatible Mode
  - Supports IDE PIO Timing Mode 0, 1, 2 of ANSI ATA Specification
  - Supports Mode 3 and Mode 4 Timing Proposal on Enhanced IDE Specification
  - Supports Multiword DMA Mode 0, 1, 2
  - Separate IDE Bus
  - Two 8x32-bit FIFO for PCI Burst Read/Write Transfers.
- **On-Board Plug and Play Port**
  - Supports Two Steerable DMA Channels
  - Supports Two Steerable Interrupts
  - Supports One Programmable Chip Select
  - Supports DMA Type F Timing (in the future)
- **208-Pin PQFP**
- **0.6 µm CMOS Technology**

4.1.2 Functional Block Diagram



* Multi-function pin

IRQ8#/OSCI	GATEA20#/DACK6#
RTCRD/PSRSTB#	KBCLK/DACK5#
RTCALE/PWRGD	KBDAT/DACK3#
RTCWR/OSCO	DAK2/DACK2#
GPIO0/PSMCLK	DAK1/DACK1#
WAKEUP0/PSMDAT	DAK0/DACK0#
KBRST#/DACK7#	IRQ1/KBLOCK

** Built-in in future version.

Figure 4.1 SiS5513 Functional Block Diagram

4.2 Functional Description

The SiS5513 is a highly integrated PCI/ISA system I/O (PSIO) device that integrates all the necessary system control logic used in PCI/ISA specific applications. The SiS5513 consists of: a PCI bridge that translates PCI cycles onto ISA bus, and ISA master/DMA device cycles onto PCI bus; a seven-channel programmable DMA Controller, a sixteen-level programmable interrupt controller, a programmable timer with three counters, a built-in RTC with 256 bytes CMOS SRAM, a on-board Plug and Play port, and a built-in PCI master/slave IDE interface.

Since 5513 includes a PCI to ISA bridge and a PCI IDE, it naturally becomes a multifunction device. The PCI/ISA bridge is defined as a function 0 device while PCI IDE is a function 1 device. The following two examples describe how to write register XX in PCI to ISA bridge configuration space and register YY in PCI IDE configuration space.

Example 1:

```
MOV EAX, 800010XXh
OUT 0CF8h, EAX
MOV AL, data
OUT 0CFDh, AL
```

Example 2:

```
MOV EAX, 800011YYh
OUT 0CF8h, EAX
MOV AL, data
OUT 0CFDh, AL
```

4.2.1. PCI Bridge

The SiS5513 PCI bus interface provides the interface between PSIO and the PCI bus. It contains both PCI master and slave bridge to the PCI bus. When PHLDA# is asserted, the master bridge translates the ISA master, DMA cycles or PCI IDE Master cycles onto the PCI bus based on the decoding status from ISA address decoder. When PHLDA# is negated, the slave bridge accepts these cycles initiated on the PCI bus targeted to the PSIO internal registers or ISA bus , and then forwards the cycles to the ISA Bus Interface that further translates them onto the ISA Bus. The PCI address decoder provides the information on which the slave bridge depends to respond and process the cycle initiated by PCI Masters.

PCI Slave Bridge

As a PCI slave, PSIO responds to both I/O and memory transfers. PSIO always target-terminates after the first data phase for any bursting cycle.

SiS5513 always converts the single interrupt acknowledge cycle (from 5511) into two cycles that the internal 8259 pair can respond to.

The PSIO is assigned as the subtractive decoder in the Bus 0 of the SiS PCI/ISA system by accepting all accesses not positively decoded by some other agent. In reality, the PSIO only subtractively responds to low 64K I/O or low 16M memory accesses. PSIO also positively decodes

I/O addresses for internal registers, and BIOS memory space by asserting DEVSEL# on the medium timing.

PCI Master Bridge

As long as PHLDA# is asserted, the PCI master bridge on behalf of DMA devices or ISA Masters starts to drive the AD bus, C/BE[3:0]# and PAR signal. When MRDC# or MWTC# is asserted, the PSIO will generate FRAME#, and IRDY# to PCI bus if the targeted memory is not on the ISA side. The valid address and command are driven during the address phase, and PAR is asserted one clock after that phase. PSIO always activated FRAME# for 2 PCLKs because it does not conduct any bursting cycle.

The ISA address decoder is used to determine the destination of ISA master or DMA devices. This decoder provides the following options as they are defined in configuration registers 48 to 4B.

a. Memory: 0-512K
b. Memory: 512K-640K
c. Memory: 640K-768K(video buffer)
d. Memory: 768K-896K in eight 16K sections(Expansion ROM)
e. Memory: 896K-960K(lower BIOS area)
f. Memory: 1M-XM-16M within which a hole can be opened. Access to the hole is not forwarded to PCI bus.
g. Memory:>16Mb automatically forwards to PCI.

4.2.2 ISA Bus Controller

The SiS5513 ISA Bus Interface accepts those cycles from PCI bus interface and then translates them onto the ISA bus. It also requests the PCI master bridge to generate PCI cycle on behalf of DMA or ISA master. The ISA bus interface thus contains a standard ISA Bus Controller and a Data Buffering logic. IBC provides all the ISA control, such as ISA command generation, I/O recovery control, wait-state insertion, and data buffer steering. The PCI to/from ISA address and data bus bufferings are also all integrated in SiS5513. The SiS5513 can directly support six ISA slots without external data or address buffering.

Standard ISA bus refresh is requested by Counter 1, and then performed via the IBC. IBC generates the pertinent command and refreshes address to the ISA bus. Since the ISA refresh is transparent to the PCI bus and the DMA cycle, an arbiter is employed to resolve the possible conflicts among PCI cycles, refresh cycles, and DMA cycles.

4.2.3 DMA Controller

The SiS5513 contains a seven-channel DMA controller. The channel 0 to 3 is for 8-bit DMA devices while channel 5 to 7 is for 16-bit devices. The channels can also be programmed for any of the four transfer modes, which include single, demand, block, and cascade. Except in cascade mode, each of the three active transfer modes can perform three different types of transfers, which include read, write, and verify. The address generation circuitry in SiS5513 can only support 24-bit address for DMA devices.

4.2.4 Interrupt Controller

The SiS5513 provides an ISA compatible interrupt controller that incorporates the functionality of two 82C59 interrupt controllers. The two controllers are cascaded so that 14 external and two internal interrupts are supported. The master interrupt controller provides IRQ<7:0> and the slave one provides IRQ<15:8>. The two internal interrupt are used for internal functions only and are not available externally. IRQ2 is used to cascade the two controllers together and IRQ0 is used as a system timer interrupt and is tied to interval Counter 0. The remaining 14 interrupt lines are available for external system interrupts.

Table 4-1

Priority	Label	Controller	Typical Interrupt Source
1	IRQ0	1	Timer/Counter 0 Out
2	IRQ1	1	Keyboard
3-10	IRQ2	1	Interrupt from Controller 2
3	IRQ8#	2	Real Time Clock
4	IRQ9	2	Expansion bus pin B04
5	IRQ10	2	Expansion bus pin D03
6	IRQ11	2	Expansion bus pin D04
7	IRQ12	2	Expansion bus pin D05
8	IRQ13	2	Coprocessor Error Ferr#
9	IRQ14	2	Fixed Disk Drive Controller Expansion bus pin D07
10	IRQ15	2	Expansion bus pin D06
11	IRQ3	1	Serial port 2, Expansion Bus B25
12	IRQ4	1	Serial port 1, Expansion Bus B24
13	IRQ5	1	Parallel Port 2, Expansion Bus B23
14	IRQ6	1	Diskette Controller, Expansion Bus B22
15	IRQ7	1	Parallel Port, Expansion Bus B21

In addition to the ISA features, the ability to do interrupt sharing is included. Two registers(ECLR) located at 4D0h and 4D1h are defined to allow edge or level sense selection to be made on an individual channel by channel basis instead of on a complete bank of channels. Note that the default of IRQ0, IRQ1, IRQ2, IRQ8# and IRQ13 is edge sensitive, and can not be programmed. Also, each PCI Interrupt(INTx#) can be programmed independently to route to one of the eleven ISA compatible interrupts(IRQ<7:3>, IRQ<15:14>, and IRQ<12:9>) through configuration registers 41h to 44h.

4.2.5 Timer/Counter

The SiS5513 contains 3 channel counter/timer that is equivalent to those found in the 82C54 programmable interval timer. The counters use a division of 14.31818MHz OSC input as the clock source. The outputs of the timers are directed to key system functions. Counter 0 is connected to the interrupt controller IRQ0 and provides a system timer interrupt for a time-of-day, diskette time-out, or the other system timing function. Counter 1 generates a refresh-request signal and Counter 2 generates the tone for the speaker.

4.2.6 Built-in RTC

The 5513 incorporates a real-time clock and system configuration memory. The RTC combines:

- A complete time-of-day clock with alarm
- 100 year calendar
- Programmable periodic interrupt
- 14 bytes of clock and control registers and 242 bytes of lower power general purpose SRAM

The method of accessing the upper 128 bytes of CMOS SRAM is to write 50h to I/O port 22h and then setting bit 3 of I/O port 23h.

4.2.7 Built-in PCI Master/Slave IDE

Design of the built-in PCI IDE follows the PCI Local Bus Specification and PCI IDE Controller Specification.

Both primary and secondary channel may be programmed as Native mode or Compatibility mode via the Class Code Field in the controller's Configuration Space register.

In Compatibility mode, the interrupt requests for channel 0 and channel 1 are rerouted to IRQ 14 and IRQ 15 of the built-in Interrupt Controller.

Following table illustrates the accessing methods to the I/O ports in compatibility mode:

Table 4-2 Primary Channel

			READ		WRITE	
PORT	IDECS1#	IDECS0#	DIOR0#	DIOR1#	DIOW0#	DIOW1#
1F0	1	0	0	1	0	1
1F1	1	0	0	1	0	1
1F2	1	0	0	1	0	1
1F3	1	0	0	1	0	1
1F4	1	0	0	1	0	1
1F5	1	0	0	1	0	1
1F6	1	0	0	1	0	1
1F7	1	0	0	1	0	1
3F6	0	1	0	1	0	1

Table 4-3 Secondary Channel

			READ		WRITE	
PORT	IDECS1#	IDECS0 #	DIOR0#	DIOR1#	DIOW0#	DIOW1#
170	1	0	1	0	1	0
171	1	0	1	0	1	0
172	1	0	1	0	1	0
173	1	0	1	0	1	0
174	1	0	1	0	1	0
175	1	0	1	0	1	0
176	1	0	1	0	1	0
177	1	0	1	0	1	0
376	0	1	1	0	1	0

In Native mode, the interrupt requests of both channels share the same PCI interrupt pin. The interrupt pin may be rerouted to any one of eleven ISA compatible interrupts (IRQ[15:14], IRQ[12:9], and IRQ[7:3]) via programming 5513 Configuration Register 63h.

Meanwhile, accessing of the I/O ports are via the addresses programmed in Base Address Registers 10h, 14h, 18h and 1Ch in IDE configuration space.

While serving as a bus master device, the IDE controller may transfer data between IDE devices and main memory directly. By performing the DMA transfer, IDE offloads the CPU and improves system performance. Bus master DMA programming is according to the information specification "Programming Interface for Bus Master IDE Controller".

Master PIO mode, which means that PCI site is running in Master mode, while the IDE site is running in PIO mode, is also supported here. Master PIO mode for Primary Channel and Secondary Channel may be enabled via IDE configuration register 4A, bit 4 and bit 3.

Under master mode, IDE controller shares the same request (PHOLD) and acknowledge (PHLDA) signals with PSIO via a high performance hidden arbitration scheme.

The built-in IDE controller contains PCI configuration header and registers to meet PCI specifications. The internal PCI IDE supports PCI type 0 configuration cycles of configuration mechanism #1.

Proper cycle timing is generated to fit PCI Bus speed and different modes of IDE drive. All cycle timing can be controlled by software programming.

As a slave device, IDE decodes and interprets PCI cycles and generate signals to start and terminate IDE cycles. This block responds only to cycles that belong to IDE I/O address space. It supports both 16-bit and 32-bit I/O data transfer at address 1F0/170. All other IDE registers read or write operations are 8-bit only.

There are two 8×32 bits FIFOs, which support post write and pre-fetch operations, in the internal PCI IDE. Prefetch and post write operations for each channel may be activated via Register 4B in IDE Configuration Space. The two FIFOs may operate independently.

The posted write operations can enhance the transfer rate of the PCI Bus interface to IDE interface write operation by decoupling the wait-states effect from the slower IDE side to the faster PCI Bus side.

The prefetch operations can eliminate the idle cycle of the PCI Bus side to improve read operation.

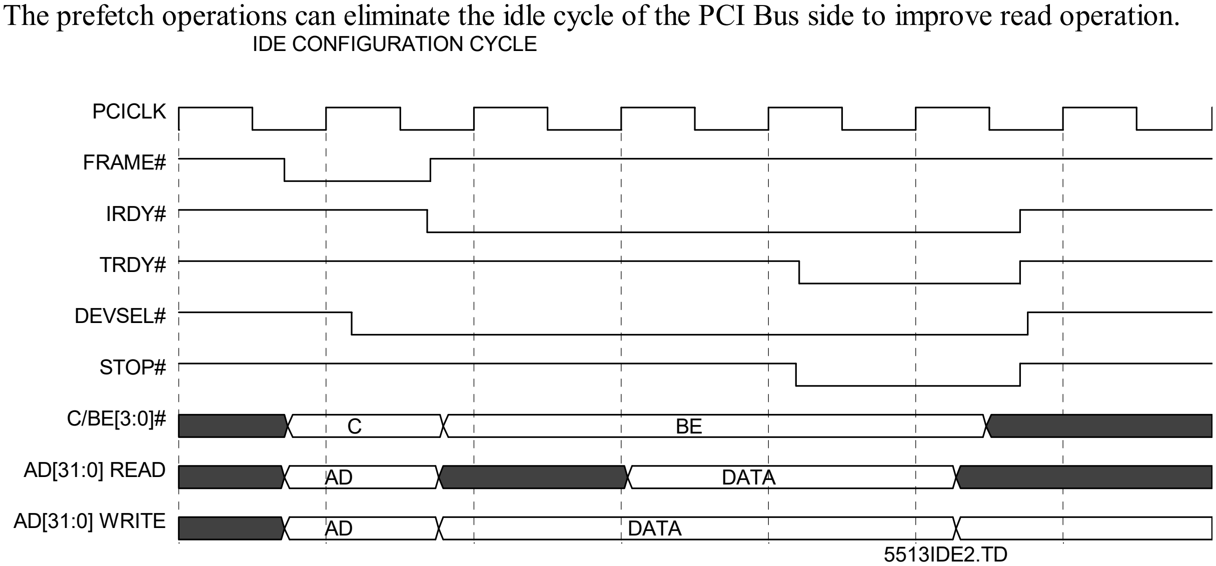


Figure 4.2

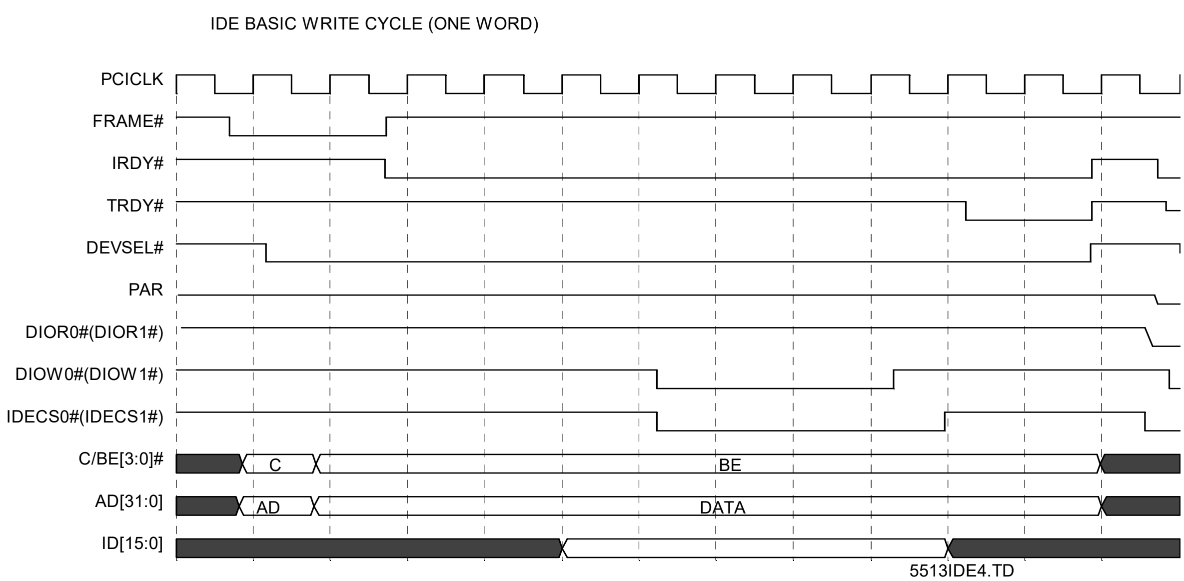


Figure 4.3

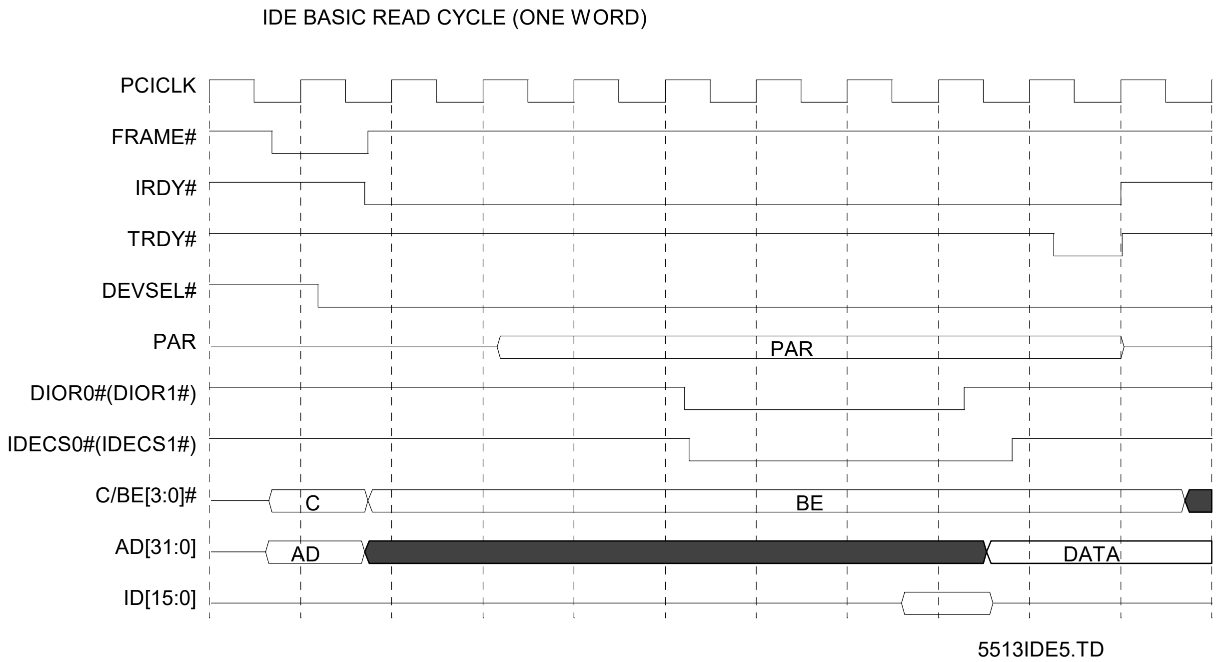


Figure 4.4

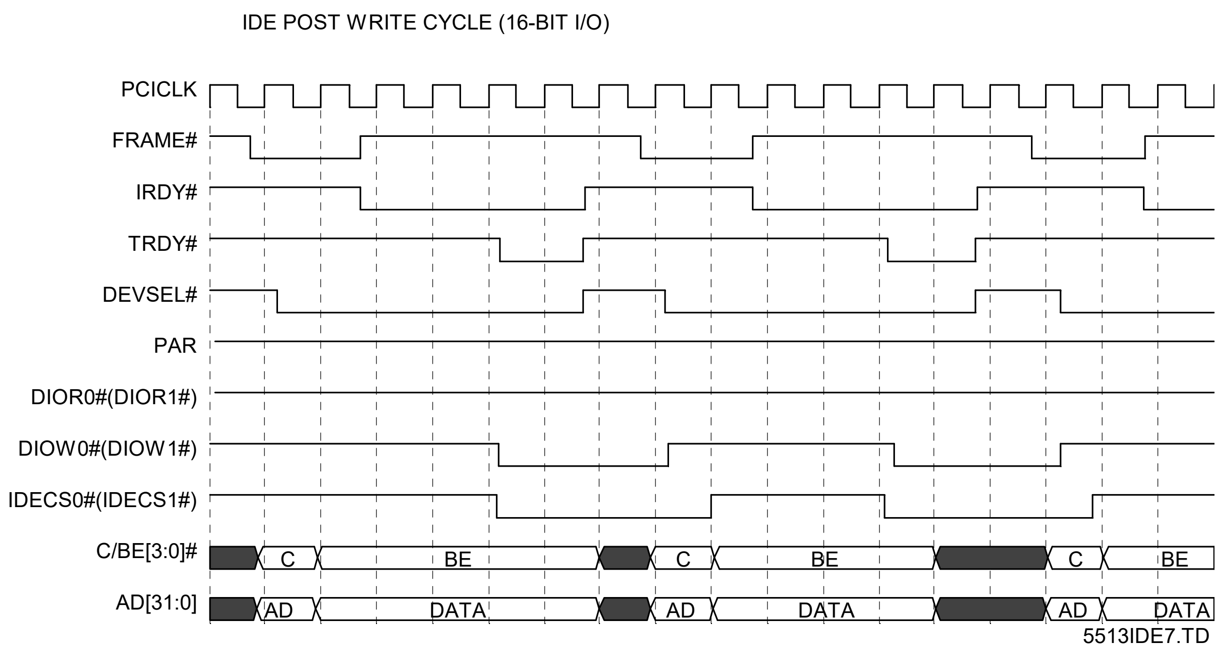


Figure 4.5

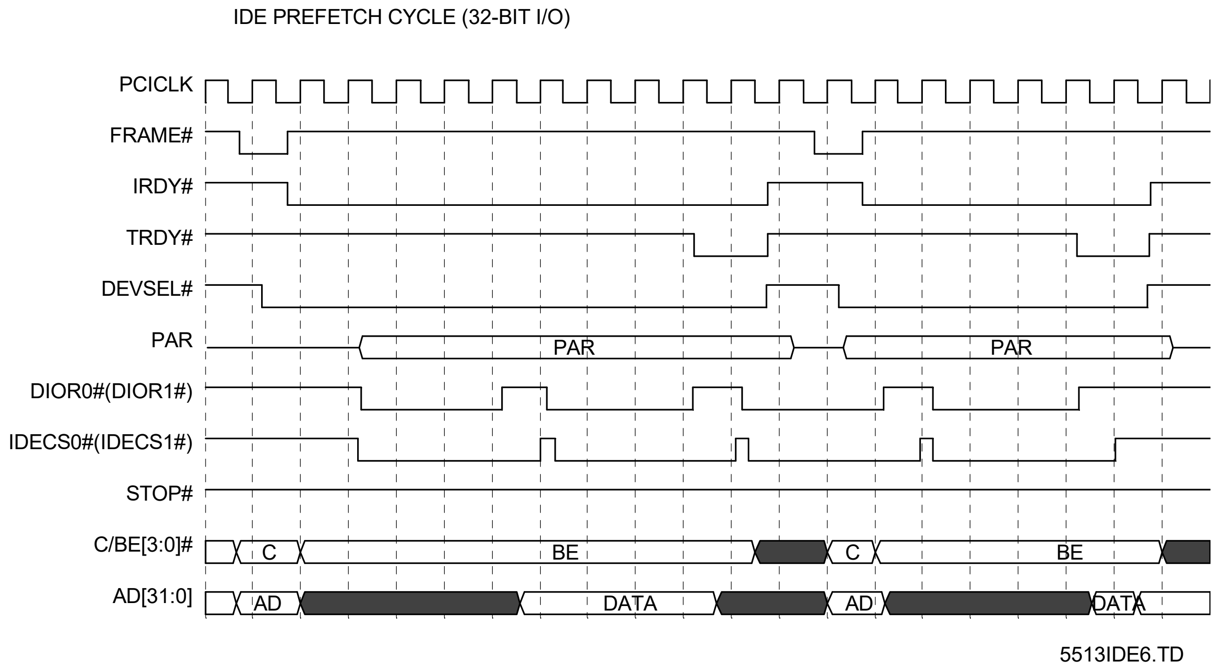


Figure 4.6

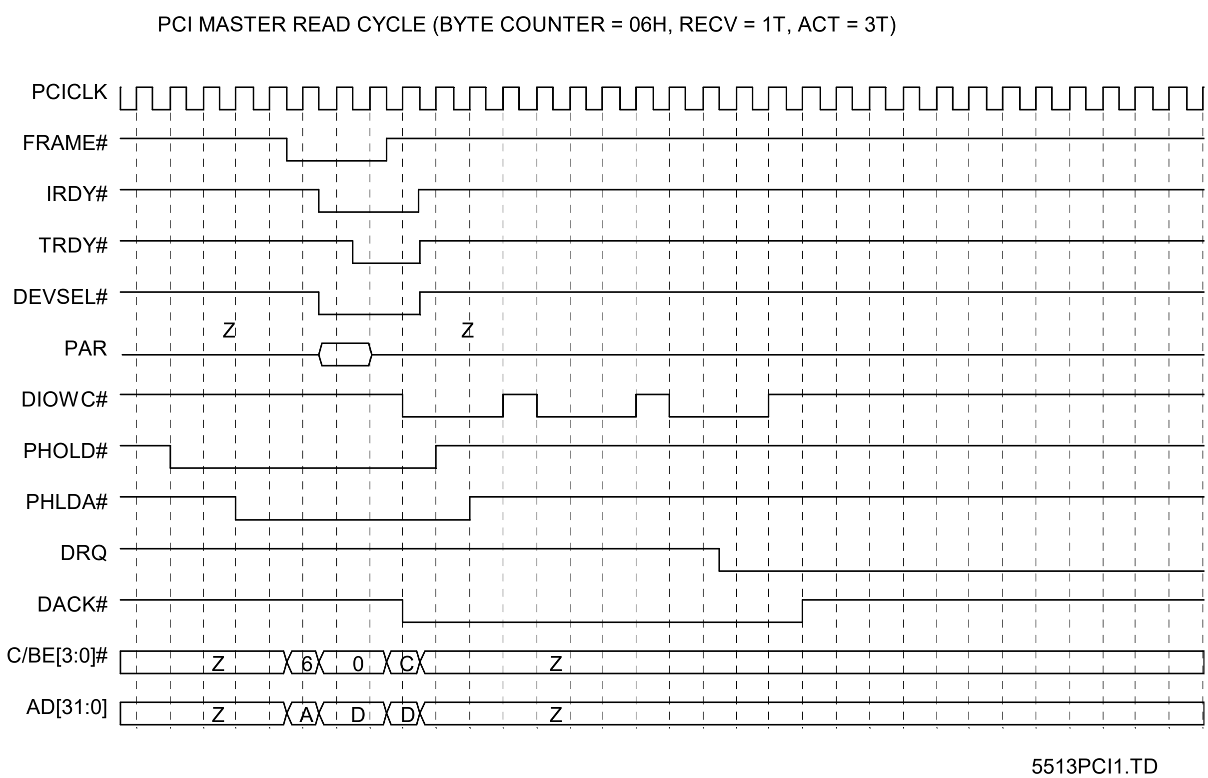


Figure 4.7

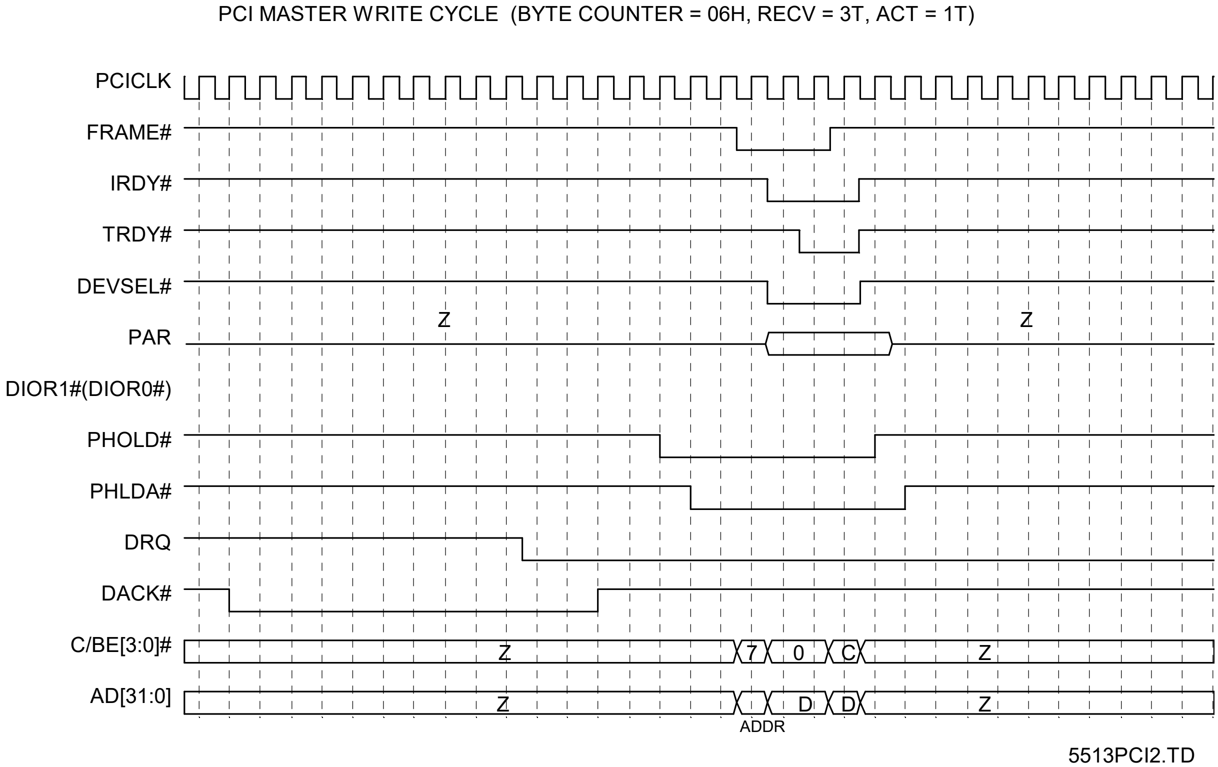


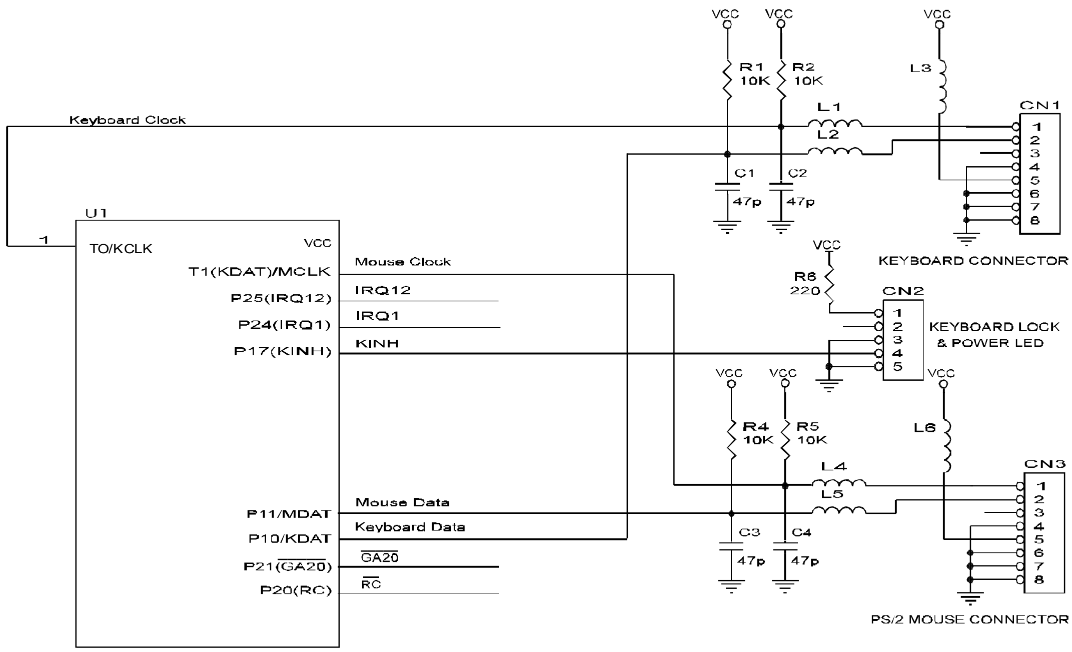
Figure 4.8

4.2.8 Internal Keyboard

Hardwired methodology is used instead of software implementation as in traditional 8042 keyboard BIOS. This enable KBC to have instant response to all the commands send from keyboard to the CPU BIOS. It also has Fast Gate-20 and Fast Reset features. The computer's performance will be improved if the response time of *Gate-A20 and Fast Reset switching speed are fast especially for programs that use memory running above 1M bytes or switch between real mode and protect mode.

The internal KBC resembles traditional keyboard BIOS as if is running at 160MHz. Normally the speed of keyboard BIOS is only 8 MHz. There is a 20 times improvement of keyboard BIOS response, Programs such as VDISK, Autocad, Microsoft Window 3.1, Novell, and many others will run faster.

Block Diagram



INTERNAL KBC. APPLICATION CIRCUIT

Status Register

The status register is an 8 bits read only register at I/O address hex 64. It has information about the state of the keyboard controller and interface. It may be read at any time.

Bit	Bit Definition	Function
0	Output Buffer Full	0 -- Output Buffer Empty 1 -- Output Buffer Full. The controller has placed data into its output buffer but the system has not yet read data
1	Input Buffer Full	0 -- Input Buffer Empty nput Buffer Full. Data has been written into the buffer but the controller has not read the data
2	System Flag	This bit may be set to 0 or 1 by writing to system flag bit in the keyboard controller's command byte. It is set to 0 after a power on reset
3	Command/Data	0 -- Data Byte. Writing to I/O 60h 1 -- Command Byte. Writing to I/O 64h
4	Inhibit Switch	0 -- Keyboard is Inhibited 1 -- Keyboard is not Inhibited
5	Auxiliary Output Buffer Full	0 -- Keyboard Data 1 -- Mouse Data
6	Time-out Error	0 -- No Transmission Time-out Error 1 -- Transmission Time-out Error
7	Parity Error	0 -- Odd Parity (No Parity Error) 1 -- Even Parity (Parity Error)

Input / Output Buffer

Input Buffer

The input buffer is an 8 bits write only register at I/O address hex 60 or 64. Writing to address hex 60 sets a flag, that indicates a data write; writing to address hex 64 sets a flag, indicating a command write. Data written to I/O address hex 60 is sent to the keyboard, unless the keyboard controller is expecting a data byte following a controller command. Data should be written to the controller's input buffer only if the input buffer's full bit in the status register equal 0. The next command are valid keyboard controller commands.

Output Buffer

The output buffer is an 8 bits read only register at I/O address hex 60. The keyboard controller uses the output buffer to send scan codes received from the keyboard, and data bytes requested by command to the system. The output buffer should be read only when output buffer's full bit in the status register set to 1.

Commands (I/O Address 64H)

Write I/O Address 64h that is Keyboard BIOS Command:

<table>
<tr><th>Command</th><th colspan="2">Keyboard Mode</th><th>Keyboard PS/2 Mode</th></tr>
<tr><td>00-1F</td><td colspan="3">Read Internal RAM -- The controller sends value of RAM to output buffer.</td></tr>
<tr><td>20</td><td colspan="3">Read Keyboard Controller's Command Byte -- The controller sends its current Command byte to its output buffer.</td></tr>
<tr><td>21-3F</td><td colspan="3">Read Internal RAM -- The controller sends value of RAM to output buffer.</td></tr>
<tr><td>40-5F</td><td colspan="3">Write Internal RAM -- The next byte of data written to I/O 60h is placed into Internal RAM.</td></tr>
<tr><td rowspan="10">60</td><td colspan="3">Write Keyboard Controller's Command Byte -- The next byte of data written to I/O 60h is placed in the controller's command byte.</td></tr>
<tr><td>Bit</td><td colspan="2">Bit Definitions</td></tr>
<tr><td>0</td><td colspan="2">Enable Keyboard Output-Buffer-Full Interrupt.
Generates an interrupt when it places keyboard data into its output buffer.</td></tr>
<tr><td>1</td><td colspan="2">In Keyboard Mode: Reserved to 0.
In Keyboard PS/2 Mode:
1 -- Enable Mouse-Buffer-Full Interrupt.
Generates an interrupt when it places mouse data into its output buffer.</td></tr>
<tr><td>2</td><td colspan="2">1 -- The controller generates an System Flag.</td></tr>
<tr><td>3</td><td colspan="2">The value written to this bit is placed in the system flat bit of the controller's status register.</td></tr>
<tr><td>4</td><td colspan="2">1 -- Disable Keyboard.
Disable the Keyboard interface by driving the 'clock' line low.
Data is not sent or received.</td></tr>
<tr><td>5</td><td colspan="2">1 -- Disable Mouse.
Disable the mouse interface by driving the 'clock' line low.
Data is not sent or received.</td></tr>
<tr><td>6</td><td colspan="2">1 -- IBM Personal Computer Compatibility Mode.
Convert the scan codes received from keyboard to IBM PC.
This includes converting a two-byte sequence to the one-byte IBM Personal Computer format.</td></tr>
<tr><td>7</td><td colspan="2">0 -- Reserved.</td></tr>
<tr><td>61-7F</td><td colspan="3">Write Internal RAM -- The next byte of data written to I/O 60h is placed into Internal RAM.</td></tr>
<tr><td>A0</td><td colspan="3">Read Internal ROM -- The controller sends value to its output buffer that end with a "0".</td></tr>
</table>

A1	Read Keyboard Controller's Version - The version code result will be placed to its output buffer.	
A4	Reset Internal Register B to 0.	Test Password Installed -- The controller sends value to its output buffer: FAh -- Password installed F1h -- Password not installed
A5	Reset Internal Register B to 0.	Load Password -- The next byte of data written to I/O 60h is placed into password stream (max 16 byte) that ends with "0". The password is store in scan code format.
A6	Read Internal Register B -- The controller sends value to its output buffer.	Enable Password Security -- The keyboard data does not sends to output buffer until the keyboard data are match the password stream..
A7	Set Internal Register C to 0.	Disable Mouse Device -- This disable the mouse interface by driving the mouse clock line low.

Commands (I/O Address 64H)

Command	Keyboard Mode	Keyboard PS/2 Mode
A8	Set Internal Register C to 1.	Enable Mouse Device -- This enable the mouse interface by driving the mouse clock line float.
A9	Read Internal Register C -- The controller sends value to its output buffer.	Mouse Device Interface Test -- Test the controller's mouse clock and data line and place the result to output buffer as follows : 00 -- No error detected. 01 -- The 'Mouse Clock' line is stuck low. 02 -- The 'Mouse Clock' line is stuck high. 03 -- The 'Mouse Data' line is stuck low. 04 -- The 'Mouse Data' line is stuck high.
AA	Self-Test - This commands the controller to perform internal diagnostic tests. A hex 55 is placed in the output buffer if no errors are detected.	
AB	Keyboard Interface Test -- This commands the controller to test the keyboard clock and data line. The test result is placed in the output buffer as follows : 00 -- No error detected. 01 -- The 'Keyboard Clock' line is stuck low. 02 -- The 'Keyboard Clock' line is stuck high. 03 -- The 'Keyboard Data' line is stuck low. 04 -- The 'Keyboard Data' line is stuck high.	
AD	Disable Keyboard Feature -- This command sets bit 4 of the controller's command byte. This disable the keyboard interface by driving the clock line low. Data will not be sent or received.	
AE	Enable Keyboard Interface -- This command clears bit 4 of command byte which release the keyboard interface	
B0	Set P10 to 0	Not Valid
B1	Set P11 to 0	Not Valid
B8	Set P10 to 1 (Default)	Not Valid
B9	Set P11 to 1 (Default)	Not Valid
C0	Read Input Port -- This command the controller to read its input port and place the data in its output buffer. This command should be used only if the output buffer is empty.	
C1*	Set Port P17 to 0 & KINH disbale	Set Port P17 to 0 & KINH disable
C2	Not Valid	Place Bit 7-4 of Input Port to status register
C3	Not Valid	Place Bit 3-0 of Input Port to status register
C7*	Set Port P17 to 1	Set Prot P17 to 1

Commands (I/O Address 64H)

Command	Keyboard Mode	Keyboard PS/2 Mode
CA	Read Internal Register D -- The Internal Register will be placed into its output buffer.	
CB	Write Internal Register D -- The next byte of data written to I/O 60h is placed in the controller's output port.	
D0	Read Output Port -- This command causes the controller to read its output port and place data in its output buffer. This command should be issued only if the output buffer is empty.	
D1	Write Output Port -- The next byte of data written to I/O 60h is placed in the controller's output port.	
D2	Not Valid	Write Keyboard Output Buffer - The next byte of data written to I/O 60h is placed in output buffer as it receive from keyboard.
D3	Not Valid	Write Mouse Output Buffer - The next byte of data written to I/O 60h is placed in output buffer as it receive from mouse.
D4	Not Valid	Write Mouse Device - The next byte of data written to I/O 60h is transmitted mouse device.
D6*	Enable P17(KINH) Keyboard Inhibit Switch (Default)	
D7*	Disable P17(KINH) Keyboard Inhibit Switch, P17 define to I/O by C1 & C7 command	
F0-FF	Pulse Output Port -- Bits 0 through 3 of controller's output port may be pulsed low for approximately 6us. Bits 0 through 3 of this command indicate which bits are to be pulsed. A 0 indicates that the bit should be pulsed, and a 1 indicate the bit should not be modified.	

I/O Ports

Input Port Definitions

Bit	Bit Definitions
0	P10 -- User Define I/O at Standard Keyboard BIOS Mode Keyboard Data Input at Keyboard PS/2 Mode
1	P11 -- User Define I/O at Standard Keyboard BIOS Mode Mouse Data Input at Keyboard PS/2 Mode
7	P17 -- KINH Keyboard Inhibit Switch or User Define I/O 0 = Keyboard Inhibited 1 = Keyboard Not Inhibited

Output Port Definitions

Bit	Bit Definitions
0	P20 -- RC System Reset
1	P21 -- GA20 Gate A20
4	P24 -- INT Keyboard Output Buffer Full IRQ1, 1 is active.
5	P25 -- No Function at Standard Keyboard BIOS Mode Mouse Interrupt IRQ12 at Keyboard PS/2 Mode, 1 is active.

4.3 Register Description

4.3.1 PCI Configuration Register (PCI to ISA Bridge)

Registers 00h, 01h   Vendor ID

Bits 15:0 = 1039h (Read Only)

Registers 02h, 03h   Device ID

Bits 15:0 = 0008h (Read Only)

Registers 04h, 05h Command = 07h

Bits 15:4 **Reserved. Read as 0's**

Bit 3 **Monitor Special Cycle Enable = 0**

Bit 2 **Behave as Bus Master Enable = 1**

Bit 1 **Respond to Memory Space Accesses = 1**

Bit 0 **Respond to I/O Space Accesses = 1**

Registers 06h, 07h Status

Bits 15:14 **Reserved. Read as 0's**

Bit 13 **Received Master-Abort**

When the 5513 generates a master-abort, this bit is set to a 1. This bit is cleared to 0 by writing a 1 to this bit.

Bit 12 **Received Target-Abort**

When the 5513 receives a target-abort, this bit is set to a 1. Software clears this bit to 0 by writing a 1 to this bit location.

Bit 11 **Reserved. Read as a 0**

Bits 10:9 **DEVSEL# Timing**

The 5513 always generates DEVSEL# with medium timing, these two bits are always set to 01.

Bits 8:0 **Reserved. Read as 0's.**

Register 08h Revision ID

Bits 7:0 = 00h (Read Only)

Register 0B~09h Class Code

Bits 23:0 **060100h (Read Only)**

Register 0Eh Header Type

Bits 7:0 **80h (Read Only)**

Register 40h BIOS Control Register

Bit 7 **Reserved. Read as a 0.**

Bit 6 **Reserved. Read as a 0.**

Bit 5

When ISA MASTER retries, Arbiter deasserts PHLDA#. This bit defaults to 0.

Bit 4 PCI Posted Write Buffer Enable

The default value is 0 (disabled).

Bits [3:0] determine how the 5513 responds to F segment, E segment, and extended segment (FFF80000-FFFDFFFF) accesses. 5513 will positively respond to extended segment access when bit 0 is set. Bit 1, combining with bits [3:2], enables 5513 to respond to E segment access.

Bit 3 Positive Decode of Upper 64K BYTE BIOS Enable.

Bit 2 BIOS Subtractive Decode Enable.

Bits [3:2]	F segment		E segment		Comment
	+	-	+	-	
00			√ *		5513 positively responds to E segment access.
01		√			5513 subtractively responds to F segment access.
10	√		√ *		5513 positively responds to E and F segment access.
11	√				5513 positively responds to F segment access.

*: enabled if bit 1 is set.

Bit 1 Lower BIOS Enable.

Bit 0 Extended BIOS Enable. (FFF80000~FFFDFFFF)

Register 41h INTA# Remapping Control Register

Bit 7 Remapping Control

When enabled, INTA#, is remapped to the PC compatible interrupt signal specified in IRQ remapping table. This bit is set to 1 after reset.

0: Enable

1: Disable

Bits 6:4 Reserved. Read as 0's.

Bits 3:0 IRQx Remapping table.

Bits	IRQx#	Bits	IRQx#	Bits	IRQx#	Bits	IRQx#
0000	reserved	0101	IRQ5	1010	IRQ10	1111	IRQ15
0001	reserved	0110	IRQ6	1011	IRQ11		
0010	reserved	0111	IRQ7	1100	IRQ12		
0011	IRQ3	1000	reserved	1101	reserved		
0100	IRQ4	1001	IRQ9	1110	IRQ14		

Register 42h INTB# Remapping Control Register

Bit 7 Remapping Control

Bits 6:4 Reserved. Read as 0's.

Bits 3:0 IRQ Remapping table.

Register 43h INTC# Remapping Control Register

Bit 7 Remapping Control

Bits 6:4 Reserved. Read as 0's.

Bits 3:0 IRQ Remapping table.

Register 44h INTD# Remapping Control Register

Bit 7 Remapping Control

Bits 6:4 Reserved. Read as 0's.

Bits 3:0 IRQ Remapping table.

NOTE: The difference INT[A:D]# can be remapped to the same IRQ signal, but this IRQ signal should be set to level sensitive.

Register 48h ISA Master/DMA Memory Cycle Control Register 1

The ISA master or DMA memory access cycles will be forwarded to PCI bus when the address fall within the programmable region defined by bits[7:4]. The base address of the programmable region is 1Mbyte, and the top addresses is programmed in 1MByte increments from 1MByte to 16MByte. All memory cycles will be forwarded to PCI bus besides the cycle fall within memory hole defined in register 4Ah and 4Bh.

Bits 7:4

Bit 7	Bit 6	Bit 5	Bit 4	Top of Memory
0	0	0	0	1 MByte
0	0	0	1	2 MByte
0	0	1	0	3 MByte
0	0	1	1	4 MByte
0	1	0	0	5 MByte
0	1	0	1	6 MByte
0	1	1	0	7 MByte
0	1	1	1	8 MByte
1	0	0	0	9 MByte
1	0	0	1	10 MByte
1	0	1	0	11 MByte
1	0	1	1	12 MByte
1	1	0	0	13 MByte
1	1	0	1	14 MByte
1	1	1	0	15 MByte
1	1	1	1	16 MByte

ISA master and DMA memory cycles to the following memory regions will be forwarded to PCI bus if they are enabled.

Bit 3 **E0000h-EFFFFh Memory Region**

0: Disable

1: Enable, the cycle is forwarded to PCI bus.

Bit 2 **A0000h-BFFFFh memory Region**

0: Disable

1: Enable, the cycle is forwarded to PCI bus.

Bit 1 **80000h-9FFFFh Memory Region**

0: Disable

1: Enable

The cycle is forwarded to PCI bus.

Bit 0 **00000h-7FFFFh Memory Region**

0: Disable

1: Enable

The cycle is forwarded to PCI bus.

Register 49h ISA Master/DMA Memory Cycle Control Register 2

ISA master and DMA memory cycles to the following memory regions will be forwarded to PCI bus if they are enabled.

Bit 7 DC000h-DFFFFh Memory region

0: Disable

1: Enable

Bit 6 D8000h-DBFFFh Memory Region

0: Disable

1: Enable

Bit 5 D4000h-D7FFFh Memory Region

0: Disable

1: Enable

Bit 4 D0000h-D3FFFh Memory Region

0: Disable

1: Enable

Bit 3 CC000h-CFFFFh Memory Region

0: Disable

1: Enable

Bit 2 C8000h-CBFFFh Memory Region

0: Disable

1: Enable

Bit 1 C4000h-C7FFFh Memory Region

0: Disable

1: Enable

Bit 0 C0000h-C3FFFh Memory Region

0: Disable

1: Enable

Register 4Ah ISA Master/DMA Memory Cycle Control Register 3

Register 4Ah and register 4Bh are used to define the ISA address hole. The ISA address hole is located between 1Mbyte and 16MByte, and sized in 64KByte increments. ISA master and DMA memory cycles fall within this hole will not be forwarded to PCI bus. Register 4Ah and 4Bh are used to define the bottom and top address of the hole respectively. The hole is located between top and bottom address, and the bottom and top address must be at or above 1MByte. If bottom address is greater than top address, the ISA address hole is disabled.

Bits 7:0 **A23~A16**

Register 4Bh ISA Master/DMA Memory Cycle Control Register 4

This register is used to define the top address of the ISA Address hole.

Bits 7:0 **A23~A16**

Registers 4Ch-4Fh

Bits 7:0 **ICW1 to ICW4 of the built-in interrupt controller (master) can be read from 4Ch to 4Fh.**

Registers 50h-53h

Bits 7:0 **ICW1 to ICW4 of the built-in interrupt controller (slave) can be read from 50h to 53h.**

Registers 54h-55h

Bits 7:0 **OCW2 to OCW3 of the built-in interrupt controller (master) can be read from 54h to 55h.**

Registers 56h-57h

Bits 7:0 **OCW2 to OCW3 of the built-in interrupt controller (slave) can be read from 56h to 57h.**

Register 58h

Bits 7:0 **Low byte of the initial count number of Counter 0 in the built-in CTC can be read from 58h.**

Register 59h

Bits 7:0 **High byte of the initial count number of Counter 0 in the built-in CTC can be read from 59h.**

Register 5Ah

Bits 7:0 **Low byte of the initial count number of Counter 1 in the built-in CTC can be read from 5Ah.**

Register 5Bh

Bits 7:0 **High byte of the initial count number of Counter 1 in the built-in CTC can be read from 5Bh.**

Register 5Ch

Bits 7:0 **Low byte of the initial count number of Counter 2 in the built-in CTC can be read from 5Ch.**

Register 5Dh

Bits 7:0 **High byte of the initial count number of Counter 2 in the built-in CTC can be read from 5Dh.**

Register 5Eh

Bits 7:0 **Control word (43h) of the built-in CTC can be read from 5Eh.**

Register 5Fh

Bits 7:0 **Indicates the status whether the LSB or MSB is read or written when Read/Write word function has been processed for the corresponding counter.**

Register 60h MIRQ0 Remapping Control Register

This register controls the remapping of MIRQ0 to the PC/AT compatible IRQ inputs of the interrupt controller. MIRQ0 can be remapped to any one of the 11 interrupts.

Bit 7 **MIRQ0 Remapping Control**

When enabled, MIRQ0 is remapped to the PC compatible interrupt signal specified in IRQ remapping table.

0: Enable

1: Disable

Bit 6 **MIRQ0/IRQx Sharing Control**

0: Enable

1: Disable

The interrupt specified by IRQ remapping table is masked when this bit is disabled and MIRQ0 remapping is enabled.

When sharing and remapping are both enalbed, MIRQ0 will be remapped to the IRQ channel programmed via Register 60h.

While MIRQ0 is enalbed, the interrupt channel for ISA will automatically be masked. MIRQ0, MIRQ1 and INTA#, INTB#, INTC#, INTD# may be asserted at the same time.

Bits 5:4 **Reserved. Read as zero.**

Bits 3:0 **Interrupt Remapping Table**

This field is used to define the MIRQ0 remapping to one of the eleven PC compatible interrupts.

bits [3:0]	IRQ remapped	bits [3:0]	IRQ remapped
0000	Reserved	1000	Reserved
0001	Reserved	1001	IRQ9
0010	Reserved	1010	IRQ10
0011	IRQ3	1011	IRQ11
0100	IRQ4	1100	IRQ12
0101	IRQ5	1101	Reserved
0110	IRQ6	1110	IRQ14
0111	IRQ7	1111	IRQ15

Register 61h MIRQ1 Remapping Control Register

bit 7 MIRQ1 Remapping Control

0 : Enable

1 : Disable

bit 6 MIRQ1/IRQx Sharing Control

0: Enable

1: Disable

When sharing and remapping are both enalbed, MIRQ0 will be remapped to the IRQ channel programmed via Register 61h.

While MIRQ1 is enalbed, the interrupt channel for ISA will automatically be masked. MIRQ0, MIRQ1 and INTA#, INTB#, INTC#, INTD# may be asserted at the same time.

bits 5:4 Reserved. Read as zero.

bits 3:0 Interrupt Remapping Table

bits [3:0]	IRQ remapped	bits [3:0]	IRQ remapped
0000	Reserved	1000	Reserved
0001	Reserved	1001	IRQ9
0010	Reserved	1010	IRQ10
0011	IRQ3	1011	IRQ11
0100	IRQ4	1100	IRQ12
0101	IRQ5	1101	Reserved
0110	IRQ6	1110	IRQ14
0111	IRQ7	1111	IRQ15

Operation Rules for Rerouting Circuits:

1. If any one IDE channel is in compatibility mode, the IRQ channel mapped to that IDE channel should be assigned as the IDE IRQ channel. Nobody can share it.

   e.g.: IDE channel 0 is in compatibility mode, while channel 1 is native.

IRQ 14 should be always used by IDE channel 0. Nobody can share it. IDE channel 1 should use DIRQ to reroute it's interrupt requests.

2. If channel n is rerouted by any one of MIRQ[1:0]: (Edge & Level trigger can be both applied)

   CASE 1: Share disable:
   Except for MIRQ1 or 0, nobody can share the channel.
   MIRQ0 and MIRQ1 can't be rerouted to the same channel.

   CASE 2: Share enable:
   Including MIRQ0, MIRQ1, INTA#, INTB#, INTC#, INTD#, and DIRQ may be rerouted to the same channel.

   Here, ISA IRQn will be automatically masked.

3. If both MIRQ[1:0] are disabled:

   PCI INTA#, INTB#, INTC#, and INTD# can be rerouted to the same channel. But ISA IRQn will be automatically masked.

4. None of MIRQ[1:0], PCI interrupt pins, DIRQ are rerouted to channel n:
   ISA IRQn can be enabled.

Register 62h - On-board Device DMA Control Register

This register is used to control the remapping of MDRQ[1:0] and MDACK[1:0]# to the DREQ and DACK# signals of the 8237 DMA controller.

Bit 7 MDRQ1/MDACK1# Remapping Control

0: Disable

1: Enable

Bits 6:4 DMA Channel Remapping Table of MDRQ1/MDACK1#

Bits[6:4]	**DMA Channel**
000	Channel 0
001	Channel 1
010	Channel 2
011	Channel 3
100	Disable
101	Channel 5
110	Channel 6
111	Channel 7

If a MDRQ/MDACK# pair is programmed to one of DMA channels, the corresponding DREQ/DACK# pins are masked for that channel.

Bit 3 MDRQ0/MDAK0# Remapping Control

0: Disable

1: Enable

When this bit set to 1,the MDRQ0/MDACK0# are mapped to the compatible ISA channel specified in bits[6:4]. When this bit set to 0, the ISA DREQ/DACK# pair is used for that channel.

Bits 2:0 DMA Channel Remapping Table of MDRQ0/MDACK0#

The following table is used to select the DMA channel for MDRQ0/MDACK0#

Bits[2:0]	DMA Channel
000	Channel 0
001	Channel 1
010	Channel 2
011	Channel 3
100	Disable
101	Channel 5
110	Channel 6
111	Channel 7

Register 63h - IDEIRQ Remapping Control Register

Bit 7 IDEIRQ Remapping Control

1: Disable

0: Enable

Bit 6:4 Reserved. Read as zero.

Bits 3:0 Interrupt Remapping Table

Bits [3:0]	remapped IRQ	Bits [3:0]	remapped IRQ
0000	Reserved	1000	Reserved
0001	Reserved	1001	IRQ9
0010	Reserved	1010	IRQ10
0011	IRQ3	1011	IRQ11
0100	IRQ4	1100	IRQ12
0101	IRQ5	1101	Reserved
0110	IRQ6	1110	IRQ14
0111	IRQ7	1111	IRQ15

Register 64h - GPIO0 Control Register

Bit 7 GPIO0 Mode Control

0: Output mode

1: Input mode (default)

Bit 6 GPIO0 Input Active Level Control

0: Active low

1: Active high

Bit 5 **GPIO0 Input Bounce-Free Control**

0: Disable

1: Enable

When this bit set to 1,the GPIO0 input goes through a de-bounce circuit.

Bit 4 **Reserved. Read as zero.**

Bits 3:0 **De-bounce Count for GPIO0 De-Bounce Circuit.**

The minimum value is 2. The timer-expire interval is calculated by the following equation: The timer-expire interval=(Count-1)x0.6s

Register 65h

Bit 7 **Enable bit of the arbiter between SIO and built-in IDE Master. While disabled, IDE Master will not work.**

0: Disabled

1: Enable (default)

Bits 6:0 **Reserved (Read as 0)**

Registers 66h, 67h - GPIO0 Output Mode Control Register

A 16-bit I/O space base address defined in bit[15:2] is used to cause GPIO0 to assert "ative low" signal for subtractively decoded I/O cycles generated by PCI masters that fall in the range specified by this register. This register is available only when GPIO0 is set to output mode.

Bits 15:2 **A[15:2] of GPIO0 I/O Space Base Address**

Bits 1:0 **GPIO0 I/O Space Address Mask**

00: Mask A1, A0

01: Mask A2, A1, A0

10: Disable GPIO0 output mode function

11: Mask A3, A2, A1, A0

Registers 68h, 69h - Reserved

Register 6Ah - GPIO Status Register

Bits 7:5 **Reserved. Read as zero.**

Bit 4 **Arbiter Operating Mode**

0: Fixed priority mode.

1: Rotating priority mode.

Bit 3 **Reserved (This bit should be programmed as 0.)**

Bit 2 **Built-in RTC Status (Read Only)**

0: Not used

1: Used

When built-in RTC is used, this bit is set to 1.

Bit 1 **Reserved**

Bit 0 **GPIO0 Status**

This bit is set when GPIO0 is active and should be cleared at the end of SMI handler. This bit is meaningful only when register 65h bit 7 set to 1.

4.3.2 Non-Configuration Registers

DMA Registers

These registers can be accessed from PCI bus.

Address	Attribute	Register Name
0000h	R/W	DMA1 CH0 Base and Current Address Register
0001h	R/W	DMA1 CH0 Base and Current Count Register
0002h	R/W	DMA1 CH1 Base and Current Address Register
0003h	R/W	DMA1 CH1 Base and Current Count Register
0004h	R/W	DMA1 CH2 Base and Current Address Register
0005h	R/W	DMA1 CH2 Base and Current Count Register
0006h	R/W	DMA1 CH3 Base and Current Address Register
0007h	R/W	DMA1 CH3 Base and Current Count Register
0008h	R/W	DMA1 Status(r) Command(w) Register
0009h	WO	DMA1 Request Register
000Ah	WO	DMA1 Write Single Mask Bit
000Bh	WO	DMA1 Mode Register
000Ch	WO	DMA1 Clear Byte Pointer
000Dh	WO	DMA1 Master Clear
000Eh	WO	DMA1 Clear Mask Register
000Fh	R/W	DMA1 Write All Mask Bits(w) Mask Status Register(r)
00C0h	R/W	DMA2 CH0 Base and Current Address Register
00C2h	R/W	DMA2 CH0 Base and Current Count Register
00C4h	R/W	DMA2 CH1 Base and Current Address Register
00C6h	R/W	DMA2 CH1 Base and Current Count Register
00C8h	R/W	DMA2 CH2 Base and Current Address Register
00CAh	R/W	DMA2 CH2 Base and Current Count Register
00CCh	R/W	DMA2 CH3 Base and Current Address Register
00CEh	R/W	DMA2 CH3 Base and Current Count Register
00D0h	R/W	DMA2 Status(r) Command(w) Register
00D2h	WO	DMA2 Request Register
00D4h	WO	DMA2 Write Single Mask Bit Register
00D6h	WO	DMA2 Mode Register

00D8h	WO	DMA2 Clear Byte Pointer
00DAh	WO	DMA2 Master Clear
00DCh	WO	DMA2 Clear Mask Register
00DEh	R/W	DMA2 Write All Mask Bits(w) Mask Status Register(r)

These registers can be accessed from PCI bus or ISA bus.

Address	Attribute	Register Name
0080h	R/W	Reserved
0081h	R/W	DMA Channel 2 Low Page Register
0082h	R/W	DMA Channel 3 Low Page Register
0083h	R/W	DMA Channel 1 Low Page Register
0084h	R/W	Reserved
0085h	R/W	Reserved
0086h	R/W	Reserved
0087h	R/W	DMA Channel 0 Low Page Register
0088h	R/W	Reserved
0089h	R/W	DMA Channel 6 Low Page Register
008Ah	R/W	DMA Channel 7 Low Page Register
008Bh	R/W	DMA Channel 5 Low Page Register
008Ch	R/W	Reserved
008Dh	R/W	Reserved
008Eh	R/W	Reserved
008Fh	R/W	Refresh Low Page Register

Interrupt Controller Registers (These registers can be accessed from PCI bus or ISA bus.)

Address	Attribute	Register Name
0020h	R/W	INT 1 Base Address Register
0021h	R/W	INT 1 Mask Register
00A0h	R/W	INT 2 Base Address Register
00A1h	R/W	INT 2 Mask Register

Timer Registers (These registers can be accessed from PCI bus or ISA bus.)

Address	Attribute	Register Name
0040h	R/W	Interval Timer 1 - Counter 0
0041h	R/W	Interval Timer 1 - Counter 1
0042h	R/W	Interval Timer 1 - Counter 2
0043h	WO	Interval Timer 1 - Control Word Register

Other Registers (These registers can be accessed from PCI bus or ISA bus.)

Address	Attribute	Register Name
0061h	R/W	NMI Status Register
0070h	WO	CMOS RAM Address and NMI Mask Register
00F0h	WO	Coprocessor Error Register

Register 4D0h IRQ Edge/Level Control Register 1

Bit 7 IRQ7

0: Edge sensitive

1: Level sensitive

Bit 6 IRQ6

0: Edge sensitive

1: Level sensitive

Bit 5 IRQ5

0: Edge sensitive

1: Level sensitive

Bit 4 IRQ4

0: Edge sensitive

1: Level sensitive

Bit 3 IRQ3

0: Edge sensitive

1: Level sensitive

Bit 2 IRQ2

This bit must be set to 0. Read as 0.

Bit 1 IRQ1

This bit must be set to 0. Read as 0.

Bit 0 IRQ0

This bit must be set to 0. Read as 0.

After reset this register is set to 00h.

Register 4D1h IRQ Edge/Level Control Register 2

Bit 7 **IRQ15**

0: Edge sensitive

1: Level sensitive

Bit 6 **IRQ14**

0: Edge sensitive

1: Level sensitive

Bit 5 **IRQ13**

This bit must be set to 0. Read as 0.

Bit 4 **IRQ12**

0: Edge sensitive

1: Level sensitive

Bit 3 **IRQ11**

0: Edge sensitive

1: Level sensitive

Bit 2 **IRQ10**

0: Edge sensitive

1: Level sensitive

Bit 1 **IRQ9**

0: Edge sensitive

1: Level sensitive

Bit 0 **IRQ8**

This bit must be set to 0. Read as zero.

After reset this register is set to 00h.

4.3.3 ISA Internal Register

ISA internal registers are accessed through an address/data registers pair. Address register located at port 22h is written with the index of ISA internal register. Then ISA internal register content can be read or written through the data register at port 23h. The port 22h can be read to get the last written-in value.

Register 50h

Bits 7:6 Bus clock selection

00: 7.159MHz

01: PCICLK/4

10: PCICLK/3

Bit 5 Flash EPROM Control bit 0 (Please refer to Register 80h bit 2 for details.)

Bit 4 Reserved

Bit 3 Access Upper 128 Bytes CMOS SRAM

0: Disable

1: Enable

Bit 2 Flash EPROM Control bit 1

Previous implementation on flash EPROM support limits that EPROM is flashed upon power on till bit 5 of register 50h is set to 1. The new added feature will allow EPROM to be flashed anytime. Bit 2 of the register 50h is added and the setting of both bit 2 and bit 5 will now control the EPROM flash operation.

Register 50h bit 5	Register 50h bit 2	Operation
0	0	EPROM can be flashed
1	0	EPROM can't be flashed again
X	1	EPROM can be flashed whenever bit 5 is 0

Bit 1 Reserved

Bit 0 ISA Slew Rate Control

The default value of the following ISA signals is 8mA(min), including SA[16-0], LA[23-17], SBHE#, MRDC#, MWTC#, SMRDC#, SMWTC#, IORC#, and IOWC#. Besides, Bit 0 of ISA configuration register 80h is used to program the currents of the above signals to 12mA(min) when it is set to 1.

Register 51h

Bits 7:6 16-bit I/O cycle command recovery time

00 : 5 BUSCLK

01 : 4 BUSCLK

10 : 3 BUSCLK

11 : 2 BUSCLK

Bits 5:4 8-bit I/O cycle command recovery time

00 : 8 BUSCLK

01 : 5 BUSCLK

10 : 4 BUSCLK

11 : 3 BUSCLK

Bit 3 **Reserved**

Bit 2 **16-bit memory, I/O wait state selection**

0 : 1 wait state

1 : 0 wait state

Bit 1 **Reserved**

Bit 0 **Reserved**

Register 53h

Bits 7:2 **Reserved**

Bit 1 **Reserved and must be set to 0.**

Bit 0 **PCI Output and Bidirectional Buffers Current Selection**

0: 50mA/2.2V (default value)

1: 95mA/2.2V

Register 54h BIOS Register

Bits 7:0 **BIOS can use this register to store data.**

Register 55h

Bits 7:0 **The same value as port 70h.**

Register 58h

Bits 7:3, 1 **Corresponds to the mask bits of the IRQ7-1.**

When disabled, any event from the corresponding IRQ will cause the system to exit the system standby state.

0: Disable

1: Enable

Bit 2 **GPIO0 Mask bit**

When disabled, GPIO0 will cause the system to exit the system standby state.

0: Disable

1: Enable

Bit 0 **Mask bit of the NMI.**

When disabled, an event from the NMI will cause the system to exit the system standby state.

0: Disable

1: Enable

Register 59h

Bits 7:0 Corresponds to the mask bits of the IRQ8-15.

When disabled, any event from the corresponding IRQ will cause the system to exit the system standby state.

Register 5Ah

Bits 7:1 Corresponds to the mask bits of the IRQ7-1.

When disabled, any event from the corresponding IRQ will cause the system to exit the monitor standby state.

Bit 0 Is the mask bit of the NMI.

When disabled, an event from the NMI will cause the system to exit the monitor standby state.

Register 5Bh

Bits 7:0 Corresponds to the mask bits of the IRQ8-15.

When disabled, any event from the corresponding IRQ will cause the system to exit the monitor standby state.

4.3.4 PCI IDE Configuration Registers

Register 00, 01h - Vendor ID

Bits 15:0 1039h(Read Only)

Register 02, 03h - Device ID

Bits 15:0 5513h(Read Only)

Register 04h - Command low byte

Bits 7:3 These bits are hardwired to 0.

Bit 2 Bus Master Enable

When set, the Bus master function is enabled. It is disabled by default.

Bit 1 Memory Space Enable

This bit is disabled by default. Read only.

Bit 0 I/O Space Enable

When enabled, the built-in IDE will respond to any access of the IDE legacy ports in the compatibilty mode, or to any access of the IDE relocatable ports in the native mode. Also, any access to the PCI bus master IDE registers are allowed. This bit is zero(disabled) on reset.

Register 05h - Command high byte

Bits 15:8 00h(Read Only)

Register 06h - Status low byte

Bits 7:6 These bits are hardwired to zero.

Bit 5 This is a reserved bit, and is recommend to program 0.

Bits 4:0 These bits are hardwired to zero.

Register 07h - Status high byte

Bits 7:6 These bits are hardwired to zero.

Bit 5 Master Abort Asserted

This bit is set when a PCI bus master IDE transaction is terminated by master abort. While this bit is set, IDE will issue an interrupt request. This bit can be cleared by writing a 1 to it.

Bit 4 Received Target Abort

The bit is set whenever PCI bus master IDE transaction is terminated with target abort.

Bit 3 Signaled Target Abort.

The bit will be asserted when IDE terminates a transaction with target abort.

Bits 2:1 DEVSEL# Timing DEVT.

These two bits define the timing of asserting DEVSEL#. The built-in IDE always asserts DEVSEL# in fast timing, and thus the two bits are hardwired to 0 per PCI Spec.

Bit 0 Reserved, Read as "0".

Register 08h - Revision Identification

Bits 7:0 **00h(Read Only)**

Register 09h - Programming Interface Byte

Bit 7 **Master IDE Device**

This bit is hardwird to one to indicate that the built-in IDE is capable of supporting bus master function.

Bits 6:4 **These bits are hardwired to zero.**

Bit 3 **Secondary IDE Programmable Indicator**

This bit is hardwired to one to indicate that the secondary IDE channel can be programmed to operate in compatible or native mode.

Bit 2 **Secondary IDE Operating Mode**

This bit defines the mode that the secondary IDE channel is operating in. Zero corrsponds to 'compatibility' while one means native PCI mode. By default, the bit is 0 and is programmable.

Bit 1 **Primary IDE Programmable Indicator**

This bit is hardwired to one to indicate that the primary IDE channel can be programmed to operate in compatible or native mode.

Bit 0 **Primary IDE Operating Mode**

This bit defines the mode that the primary IDE channel is operating in. Zero corrsponds to 'compatibility' whie one means native PCI mode. The powerup state for this bit is 0, and can be set if native mode is expected.

Register 0Ah - Subclass ID

Bits 7:0 **01h**

Register 0Bh - Class ID

Bits 7:0 **01h**

Register 0Ch - Cache Line Size

Bits 7:0 **00h**

Register 0Dh- Latency Timer

Bits 7:0 **Programmable (from 0 to 255). The default value is 0.**

Register 0Eh - Header Type

Bits 7:0 80h

Register 0Fh - BIST

Bits 7:0 00h

Register 10, 11, 12, 13h Primary Channel Base Address Register

Register 14, 15, 16, 17h Primary Channel Base Address Register

Register 18, 19, 1A, 1Bh Secondary Channel Base Address Register

Register 1C, 1D, 1E, 1Fh Secondary Channel Base Address Register

In the native mode, these four registers define the IDE base address for each of the two IDE devices in both the primary and secondary channels respectively. In the compatible mode, the four registers can still be programmed and read out, but it does not affect the IDE address decoding.

Register 20, 21, 22, 23h Bus Master IDE Control Register Base Address

Offset Register	Register Access
00h	Bus Master IDE Command Register (Primary)
01h	Reserved
02h	Bus Master IDE Status Register(Primary)
03h	Reserved
04-07h	Bus Master IDE PRD (*) Table Pointer (Primary)
08h	Bus Master IDE Command Register (Secondary)
09h	Reserved
0Ah	Bus Master IDE Status Register (Secondary)
0Bh	Reserved
0C-0Fh	Bus Master IDE PRD (*) Table Pointer (Secondary)

*PRD: Physical Region Descriptor

Register 24 to 2Bh These bits are hardwired to zero

Register 2C, 2Dh, 2E, 2Fh Reserved. Read as"0".

Register 30, 31, 32, 33h Expansion ROM Base Address

These four byte registers are recommended not to be programmed.

The following 10 registers define the speed of accessing IDE data and command registers. The four most significant bits of each Recovery Time Control byte are hardwired to zero, and the rest is R/W programmable with the following definition.

Recovery Time Control

Bits[3:0] Recovery Time

0000	12 PCICLK
0001	1 PCICLK
0010	2 PCICLK
0011	3 PCICLK
0100	4 PCICLK
0101	5 PCICLK
0110	6 PCICLK
0111	7 PCICLK
1000	8 PCICLK
1001	9 PCICLK
1010	10 PCICLK
1011	11 PCICLK
1100	13 PCICLK
1101	14 PCICLK
1110	15 PCICLK
1111	15 PCICLK

The five most significant bits of each Active Time Control byte are hardwired to zero, and the rest is R/W programmable with the following definition.

Active Time Control

Bits[2:0]	**Active Time**
000	8 PCICLK
001	1 PCICLK
010	2 PCICLK
011	3 PCICLK
100	4 PCICLK
101	5 PCICLK
110	6 PCICLK
111	12 PCICLK

Register 40h IDE Primary Channel/Master Drive Data Recovery Time Control.

Register 41h IDE Primary Channel/Master Drive DataActive Time Control.

Register 42h IDE Primary Channel/Slave Drive Data Recovery Time Control.

Register 43h IDE Primary Channel/Slave Drive Data Active Time Control

Register 44h IDE Secondary Channel/Master Drive Data Recovery Time Control.

Register 45h IDE Secondary Channel/Master Drive Data Active Time Control

Register 46h IDE Secondary Channel/Slave Drive Data Recovery Time Control.

Register 47h IDE Secondary Channel/Slave Drive Data Active Time Control

Register 48h IDE Command Recovery Time Control

Register 49h IDE Command Active Time Control

Register 4Ah IDE General Control Register 0

Bit 7	**Enable\disable burst mode. (1: Enable; default value = 0)**
Bits 6:5	**Reserved**
Bit 4	**Secondary Channel using Master PIO mode.**
Bit 3	**Primary Channel using Master PIO mode.**
Bit 2	**IDE Secondary Channel enable. (0 to disable)**
Bit 1	**IDE Primary Channel enable. (0 to disable)**
Bit 0	**Reserved.**

Register 4Bh IDE General Control register 1

Bits 7:6	**Bit 0 of IDE configuration register 3Dh Interrupt pin control** While bit 6 is enabled (1), programming bit 7 may change the value of bit 0 in 3Dh. While bit 7 is 0, configuration register 3Dh is 00. While the bit 7 is 1, 3Dh is 01. While any bit of bit 0 or bit 2 in programming interface (09h) is programmed to native mode (1), Interrupt pin will be automatically set to 01.
Bit 5	**Reset IDE state machine (default value = 0; disable)**
Bit 4	**Reserved**
Bit 3	**Secondary channel post write enable. (1 to enable)**
Bit 2	**Primary channel post write enable. (1 to enable)**
Bit 1	**Secondary channel prefetch enable. (1 to enable)**
Bit 0	**Primary channel prefetch enable. (1 to enable)**

(Following two 16-bit wide registers define the prefetching length of each IDE channel respectively.)

Register 4Ch Prefetch Count of Primary Channel (Low Byte)
Register 4Dh Prefetch Count of Primary Channel (High Byte)
Register 4Eh Prefetch Count of Secondary Channel (Low Byte)
Register 4Fh Prefetch Count of Secondary Channel (High Byte)

4.3.5 PCI Bus Master IDE Registers

The PCI Bus master IDE Registers use 16 bytes of I/O Space. These registers can be accessed through I/O R/W to the address defined in the Bus Master IDE register Base Address in the Configuration space. The definition of each register is described below.

Bus Master IDE Command Register

Bits 7:4 **Reserved. Return 0 on reads.**

Bit 3 **Read or Write Control. This bit defines the R/W control of the bus master transfer. When set to zero, PCI bus master reads are conducted. When set to one, PCI bus master writes are conducted.**

Bits 2:1 **Reserved.**

Bit 0 **Start/Stop Bus Master**

The 5513 builtin IDE Controller enables its bus master operation whenever it detects this bit changing from a zero to a one. The operation can be halted by writing a zero to this bit.

Bus Master IDE Status Register

Bit 7 **Simplex Only**

This bit is hardwired to zero to indicate that only one bus master channel can be operated at a time.

Bit 6 **Drive 1 DMA Capable**

This R/W bit can be set by BIOS or driver to indicate that drvie 1 for this channel is capable of DMA transfers.

Bit 5 **Drive 0 DMA Capable**

This R/W bit can be set by BIOS or driver to indicate that drvie 0 for this channel is capable of DMA transfers.

Bits 4:3 **Reserved. Return 0 on reads**

Bit 2 **Interrupt**

The bit is set by the rising edge of the IDE interrupt line to indicate that all data transferred from the drive is visible in the system memory. Writing a '1' to this bit can reset it.

Bit 1 **Error**

This bit is set when the IDE controller encounters an error during data transferring to/from memory.

Bit 0 **Bus Master IDE Device Active**

This bit is set when the start bit in the command register is set. It can be cleared when the last transfer of a region is perfomred, or the start bit is reset.

PRD Table Pointer Register

This 32-bit register contains address pointing to the starting address of the PRD table.

Bits 31:2 Base Address of the PRD Table

Bits 1:0 Reserved

4.4 Pin Assignment and Description

4.4.1 Hardware Trap

Table 4-4

Pin No	Symbol	Description
9	ROMKBCS#	If super I/O device with built-in RTC is used, this signal should be connected GND via a 10K ohms resistor.
14	RTCALI/PWRGD	This signal is an input pin upon power up, 5513 will strobe this pin on the rising edge of PCIRST# to identify that internal for external RTC is used. If this signal is sample high, internal RTC is used. If it is sample low, external RTC is employed.
15	RTCRD/PSRSTB	This signal is an input pin upon power up, 5513 will strobe this pin on the rising edge of PCIRST# to identify that internal for external RTC is used. If this signal is sample high, internal RTC is used. If it is sample low, external RTC is employed.
158	SDIR#	This pin is used to identify that internal KBC or external KBC is used. If this pin is pulled high, external KBC is used. If it is pulled low, internal KBC is used.

The actual function of multi-function pins in various configuration is summarized as follows:

Table 4-5

	Pin 15	Pin 14	Pin 13	Pin 12
Ext RTC	RTCRD	RTCALE	RTCWR	IRQ8
Int RTC	PSRSTB	-RTCPWRGD	OSC0	OSCI

4.4.2 Pin Assignment

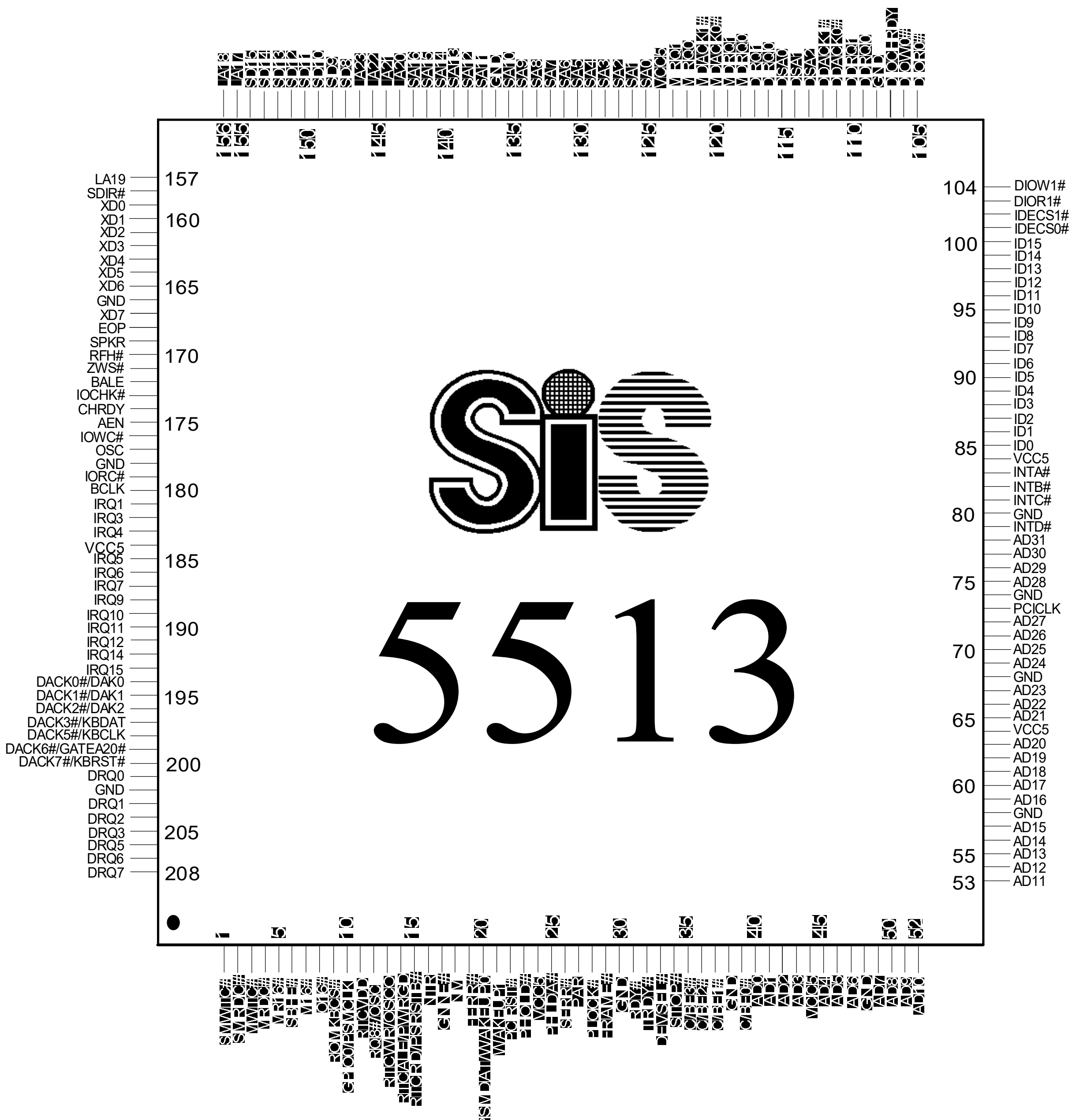


Figure 4.9

4.4.3 Pin Listing (# means active low)

Table 4-6

1=SMWTC#	5V	48=GND	VSS	95=ID10	5V
2=SMRDC#	5V	49=AD7	5V	96=ID11	5V
3=MWTC#	5V	50=AD8	5V	97=ID12	5V
4=MRDC#	5V	51=AD9	5V	98=ID13	5V
5=MR16#	5V	52=AD10	5V	99=ID14	5V
6=SBHE#	5V	53=AD11	5V	100=ID15	5V
7=M16#	5V	54=AD12	5V	101=IDECS0#	5V
8=IO16#	5V	55=AD13	5V	102=IDECS1#	5V
9=ROMKBCS#	5V	56=AD14	5V	103=DIOR1#	5V
10=GPIO0/PSMCLK	5V	57=AD15	5V	104=DIOW1#	5V
11=RTCVDD	5V	58=GND	VSS	105=DIOR0#	5V
12=IRQ8#/OSCI	5V	59=AD16	5V	106=DIOW0#	5V
13=RTCWR/OSCO	5V	60=AD17	5V	107=DIOCHRDY	5V
14=RTCALE/PWRGD	5V	61=AD18	5V	108=GND	VSS
15=RTCRD/PSRSTB#	5V	62=AD19	5V	109=DDRQ0	5V
16=INTR	5V	63=AD20	5V	110=DDRQ1	5V
17=IGNNE#	5V	64=VCC5	5V	111=DDACK0#	5V
18=NMI	5V	65=AD21	5V	112=DDACK1#	5V
19=FERR#	5V	66=AD22	5V	113=DSA2	5V
20=PSMDAT /WAKEUP0	5V	67=AD23	5V	114=DSA1	5V
21=WAKEUP1	5V	68=GND	VSS	115=DSA0	5V
22=PCIRST#	5V	69=AD24	5V	116=DIRQ0	5V
23=PHOLD#	5V	70=AD25	5V	117=DIRQ1	5V
24=VCC5	5V	71=AD26	5V	118=MDRQ0	5V
25=PHLDA#	5V	72=AD27	5V	119=MDRQ1	5V
26=SERR#	5V	73=PCICLK	5V	120=MDACK0#	5V
27=PAR	5V	74=GND	VSS	121=MDACK1#	5V
28=PLOCK#	5V	75=AD28	5V	122=MIRQ0	5V
29=FRAME#	5V	76=AD29	5V	123=MIRQ1	5V
30=GND	VSS	77=AD30	5V	124=VCC5	5V
31=IRDY#	5V	78=AD31	5V	125=SA0	5V
32=TRDY#	5V	79=INTD#	5V	126=SA1	5V
33=DEVSEL#	5V	80=GND	VSS	127=SA2	5V
34=STOP#	5V	81=INTC#	5V	128=SA3	5V
35=C/BE3#	5V	82=INTB#	5V	129=SA4	5V
36=C/BE2#	5V	83=INTA#	5V	130=SA5	5V
37=C/BE1#	5V	84=VCC5	5V	131=SA6	5V
38=GND	VSS	85=ID0	5V	132=SA7	5V
39=C/BE0#	5V	86=ID1	5V	133=SA8	5V
40=AD0	5V	87=ID2	5V	134=SA9	5V
41=AD1	5V	88=ID3	5V	135=SA10	5V
42=AD2	5V	89=ID4	5V	136=GND	VSS
43=AD3	5V	90=ID5	5V	137=SA11	5V
44=VCC5	5V	91=ID6	5V	138=SA12	5V
45=AD4	5V	92=ID7	5V	139=SA13	5V
46=AD5	5V	93=ID8	5V	140=SA14	5V
47=AD6	5V	94=ID9	5V	141=SA15	5V

142=SA16	5V	176=IOWC#	5V
143=LA20	5V	177=OSC	5V
144=LA21	5V	178=GND	VSS
145=LA22	5V	179=IORC#	5V
146=LA23	5V	180=BCLK	5V
147=SD8	5V	181=IRQ1	5V
148=SD9	5V	182=IRQ3	5V
149=SD10	5V	183=IRQ4	5V
150=SD11	5V	184=VCC5	5V
151=SD12	5V	185=IRQ5	5V
152=SD13	5V	186=IRQ6	5V
153=SD14	5V	187=IRQ7	5V
154=SD15	5V	188=IRQ9	5V
155=LA17	5V	189=IRQ10	5V
156=LA18	5V	190=IRQ11	5V
157=LA19	5V	191=IRQ12	5V
158=SDIR#	5V	192=IRQ14	5V
159=XD0	5V	193=IRQ15	5V
160=XD1	5V	194=DACK0#/DAK0	5V
161=XD2	5V	195=DACK1#/DAK1	5V
162=XD3	5V	196=DACK2#/DAK2	5V
163=XD4	5V	197=DACK3#/KBDAT	5V
164=XD5	5V	198=DACK5#/KBCLK	5V
165=XD6	5V	199=DACK6#/GATEA20#	5V
166=GND	VSS	200=DACK7#/KBRST#	5V
167=XD7	5V	201=DRQ0	5V
168=EOP	5V	202=GND	VSS
169=SPKR	5V	203=DRQ1	5V
170=RFH#	5V	204=DRQ2	5V
171=ZWS#	5V	205=DRQ3	5V
172=BALE	5V	206=DRQ5	5V
173=IOCHK#	5V	207=DRQ6	5V
174=CHRDY	5V	208=DRQ7	5V
175=AEN	5V		

4.4.4 Pin Description

Table 4-7 PCI Interface

Pin No.	Symbol	Typ	Function
39, 37-35	C/BE[3:0]#	I/O	Bus Command and Byte Enables. During the address phase of a transaction, C/BE[3:0]# define the bus command. During the data phase C/BE[3:0]# are used as byte enables. The 5513 drives C/BE[3:0]# as an initiator of a PCI bus cycle and monitors C/BE[3:0]# as a target.

78-75, 72-69, 67-59, 57-49	AD[31:0]	I/O	PCI Address/Data. AD[31:0] is a multiplexed address and data bus. During the first clock of a transaction, AD[31:0] contains a physical byte address. During the subsequent clocks, AD[31:0] contains data. When the 5513 is a target, AD[31:0] are inputs during the address phase of a transaction. During the following data phases, the 5513 supplies data on AD[31:0] for a PCI read, or accepts data for a PCI write. As an initiator, the 5513 drives a valid address on AD[31:2] during the address phase, and drives write data or latches read data on AD[31:0] during the data phases. The 5513 always drives AD[1:0] low as a master during the address phase.
29	FRAME#	I/O	FRAME# is an input to the 5513 when the 5513 is the target. FRAME# is an output when the 5513 is the initiator. FRAME# is tri-state during reset.
31	IRDY#	I/O	IRDY# indicates the 5513's ability, as an initiator, to complete the current data phase of the transaction. During a write, IRDY# indicates the 5513 has valid data present on AD[31:0]. During a read, it indicates the 5513 is prepared to latch the read data. IRDY# is an input to the 5513 when the 5513 is the target and an output when the 5513 is an initiator.
32	TRDY#	I/O	TRDY# indicates a slave's ability to complete the current data phase of the transaction. During a read, TRDY# indicates that the 5513, as a target, has placed valid data on AD[31:0]. During a write, it indicates that the 5513, as a target, is prepared to latch write data. TRDY# is an output when the 5513 is a target and an input when the 5513 is an initiator.
33	DEVSEL#	I/O	The 5513 asserts DEVSEL# to claim a PCI transaction through positive or subtractive decoding. As an output, the 5513 asserts DEVSEL# when it samples IDSEL active in configuration cycles to 5513 configuration registers. The 5513 also asserts DEVSEL# when an internal 5513 register is addressed or when the 5513 subtractively decodes a cycle. As an input, DEVSEL# indicates a PCI target has responded to a 5513 initiated transaction. The 5513 also samples this signal for all PCI transactions to decide to subtractively decode the cycle.

34	STOP#	I/O	STOP# indicates that the 5513, as a target, is requesting an initiator to stop the current transaction. As a master, STOP# causes the 5513 to stop the current transaction. STOP# is an output when the 5513 is a target and an input when the 5513 is an initiator.
27	PAR	O	PAR is an even parity calculated based on AD[31:0] and C/BE3-0#. PAR is an output during the address phase(delayed by one clock) for all 5513 initiated transactions. It is also an output during the data phase (delayed by one clock) when the 5513 is the initiator of a PCI write transaction, and when it is the target of a read transaction.
26	SERR#	I	SERR# can be pulsed active by any PCI device that detects a system error condition. Upon sampling SERR# active, the 5513 generates a non-maskable interrupt to the CPU.
28	PLOCK#	I	PLOCK# is always an input to the 5513. When the 5513 is the target of a transaction and samples PLCOK# negated during the address phase of a transaction, the 5513 considers itself a locked resource until it samples PLOCK# and FRAME# negated. When other masters attempt accesses while the 5513 is locked, the 5513 responds with a target initiated retry termination.
73	PCICLK	I	PCICLK provides timing for all transactions on the PCI bus. All other PCI signals are sampled on the rising edge of PCICLK, and all timing parameters are defined with respect to this edge. Frequencies supported by the 5513 include 25 and 33 MHz.
22	PCIRST#		PCIRST# forces the 5513 to a known state. All I/O and sustained tri-state I/O signals are forced to a high impedance state. All registers are set to their default values. PCIRST# may be asynchronous to PCICLK when asserted or negated. Although asynchronous, negation must be a clean, bounce-free edge. Note that PCIRST# must be asserted for more than 1us.
83-81, 79	INT[A:D]#	I	PCI Interrupt A to Interrupt D

Table 4-8 ISA Interface

Pin No.	Symbol	Typ	Function
142-137, 135-125	SA[16:0]	I/O	System address. They are inputs when an external bus master is in control and are outputs at all other times.
146-143, 157-155	LA[23:17]	I/O	Latched system address. They are inputs when an external bus master is in control and are outputs at all other times.
167, 165-159	XD[7:0]	I/O	Peripheral Data Bus lines.

154-147	SD[15:8]	I/O	System Data Bus are directly connected to the ISA slots.
8	IO16#	I	16-bit I/O chip select indicates that the AT bus cycle is a 16-bit I/O transfer when asserted or an 8-bit I/O transfer when it is negated.
7	M16#	I	16-bit memory chip select indicates a 16-bit memory transfer when asserted or an 8-bit memory transfer when it is negated.
6	SBHE#	I/O	Byte high enable signal indicates that the high byte has valid data on the ISA 16-bit data bus. This signal is an output except during ISA master cycles.
5	MR16#	I	Master* is an active low signal from AT bus. When active, it indicates that the ISA bus master has the control of the system. The address and control signals are all driven by the ISA bus master.
4	MRDC#	I/O	AT bus memory read command signal is an output pin during AT/DMA/refresh cycles and is an input pin in ISA master cycles.
3	MWTC#	I/O	AT bus memory write command signal is an output pin during AT/DMA cycles and is an input pin in ISA master cycles.
2	SMRDC#	I/O	AT bus memory read. It instructs the memory devices to drive data onto the data bus. It is active only when the memory being accessed is within the lowest 1MB.
1	SMWTC#	I/O	AT bus memory write. It instructs the memory devices to store the data presented on the data bus. It is active only when the memory being accessed is within the lowest 1MB.
179	IORC#	I/O	AT bus I/O read command signal is an output pin during AT or DMA cycles and is an input pin in ISA master cycles. When low, it strobes an I/O device to place data on the data bus.
176	IOWC#	I/O	AT bus I/O write command signal is an output pin during AT or DMA cycles and is an input pin in ISA master cycles. When low, it strobes data on the data bus into a selected I/O device.
175	AEN	O	Address Enable is driven high on the ISA bus to indicate the address lines are valid in DMA or ISA master cycles. It is low otherwise.

174	CHRDY	I/O	I/O channel ready is normally high. It can be pulled low by the slow devices on the AT bus to add wait states for the ISA memory or I/O cycles. When a DMA or an ISA master accesses a VL-Bus target, IORDY is an output to control the wait states.
173	IOCHK#	I	I/O channel Check is an active low input signal which indicates that an error has taken place on the I/O bus.
172	BALE	O	Bus address latch enable is used on the ISA bus to latch valid address from the CPU. Its falling edge starts the ISA command cycles.
171	ZWS#	I	Zero wait state is an active low signal. The system ignores the IORDY signal and terminates the AT bus cycle without additional wait state when it is asserted.
180	BCLK	I/O	ISA bus clock, for ISA bus controller, ISA bus interfaces and the DMA controller. It can be programmed to derive from the SYSCLK or from the 14MHz clock.
170	RFH#	I/O	Refresh signal is used to initiate a refresh cycle. This signal is an input in ISA bus master cycles and is an output in other cycles.
177	OSC	I	OSC is the buffered input of the external 14.318MHz oscillator.
168	EOP	O	Terminal Count of DMA. A pulse is generated by the DMA controller when the terminal count (TC) of any channel reaches 1. When a TC pulse occurs, the DMA controller will terminate the service, and if auto-initialize is enabled, the base registers will be written to the current registers of that channel. The mask bit and the TC bit in the Status word will be set for the currently active channel unless the channel is programmed for auto-initialize. In that case, the mask bit remains clear.
208-203, 201	DRQ7-5,3-0	I	DMA Request inputs are used by external devices to indicate when they need service from the internal DMA controllers.
181, 182-183, 185-193	IRQ1, 3-7, IRQ 9, 10-12,14,15	I	These are the asynchronous interrupt request inputs to the 8259 controller.

Table 4-9 IDE Interface & on Board Plug and Play

100-85	ID[15:0]	I/O	IDE data bus.
103, 105	DIOR[1:0]#	O	IDE I/O read cycle command.
104, 106	DIOW[1:0]#	O	IDE I/O write cycle command
107	DIOCHRDY	I	IDE I/O channel ready signal.
110, 109	DDRQ[1:0]	I	IDE DMA request signals.

112, 111	DDACK[1:0]#	O	IDE DMA acknowledge signals.
113	DSA[2:0]	O	IDE address [2:0].
117-116	DIRQ[1:0]	I	IDE interrupt request signals.
102-101	IDECS[1:0]#	O	IDE chip select signals.
123-122	MIRQ[1:0]	I	These are motherboard device interrupt request signals. They are programmable to be remapped to any one of eleven PC compatible interrupts through PCI configuration register 60h and 61h.
119-118	MDRQ[1:0]	I	These are motherboard device DMA request signals. They are programmable to be remapped to different DMA channel through PCI configuration register 62h.
121-120	MDACK[1:0]#	O	These are motherboard device DMA acknowledge signals. They are programmable to be remapped to different DMA channel through PCI configuration register 62h.

Table 4-10 Others

194	DAK0/ DACK0#	O	When built-in keyboard controller is disabled, this pin is used as DMA channel 0 acknowledge. Otherwise, it is used as DAK0.
195	DAK1/ DACK1#	O	When built-in keyboard controller is disabled, this pin is used as DMA channel 1 acknowledge. Otherwise, it is used as DAK1.
196	DAK2/DACK2#	O	When built-in keyboard controller is disabled, this pin is used as DMA channel 2 acknowledge. Otherwise, it is used as DAK2. When it is used as DACK2#, it indicates the DMA device or 16 bit master has been granted the bus. The DAK[2:0] are used to indicate that a request for DMA service has been granted by the 5513 or that a 16 bit master has been granted the bus. One external 74138 is necessary to decode these signals to DACK[7:5, 3:0]# for the corresponding peripheral.
197	KBDAT/ DACK3#	I/O	This pin is used as either keyboard data signal or DMA channel 3 acknowledge.
198	KBCLK/ DACK5#	O	This pin is used as either keyboard clock signal or DMA channel 5 acknowledge.
199	GATEA20#/ DACK6#	O	This pin is used as keyboard gate address 20 signal or DMA channel 6 acknowledge.
200	KBRST#/ DACK7#	O	This pin is used as keyboard reset signal or DMA channel 7 acknowledge.

12	IRQ8#/ OSCI	I	• When using internal RTC: This pin is used as the time base of the built-in RTC. This signal should be connected to 32.768 KHz crystal or oscillator input. • When using external RTC: This pin is used as IRQ8#, which is the asynchronous interrupt request input to the 8259 controller.
15	RTCRD/ PSRSTB#	I/O	• When using internal RTC: This signal is used as PSRSTB# (power strobe). PSRSTB# establishes the condition of the control register in RTC when power is first applied to the device. • When using external RTC: The signal is used as the data read strobe of RTC. It is used to drive the RTC data onto the XD bus when the CPU accesses the RTC.
14	RTCALE/ PWRGD	I/O	• When using internal RTC: The signal must be high for bus cycles in which the CPU accesses the RTC. All address, data, data strobe, and R/W pins of the internal RTC are disconnected from the processor when this signal is low. • When using external RTC: The signal is used to latch the address from the XD bus when CPU accesses RTC.
11	RTCVDD	I	Battery power for RTC internal RTC.
13	RTCWR/ OSCO	O	When using internal RTC: this pin should be connected the other end of the 32.768 KHz crystal or left unconnected if an oscillator is used. When using external RTC: This pin is used as data write strobe of RTC. It is used to store the data presented on the XD bus when CPU accesses the RTC. This pin must be connected to the R/W pin of RTC.
10	PSMCLK/ GPIO0	O	This pin is used as the PS/2 mouse clock signal or general purpose input/output 0.
20	PSMDAT/ WAKEUP0	I/O	This pin is used as the PS/2 mouse data signal or WAKEUP0. When it is used as WAKEUP0, the signal is directly connected to the 5511. When activated, it will cause 5511 to reload its monitor standby timer. If it is inactive and the monitor standby timer expires, the system will enter into monitor standby state. During the monitor standby state, if this input become active, the system will wake up from standby state and return back to normal state.

21	WAKEUP1	O	This signal is directly connected to the 5511. When activated, it will cause 5511 to reload its system standby timer. If it is inactive and the system standby timer expires, the system will enter into system standby state. During the system standby state, if this input become active, the system will wake up from standby state and return back to normal state.
23	PHOLD#	O	SIO Request. The 5513 asserts PHOLD# to request the PCI bus.
25	PHLDA#	I	SIO Grant. It is driven by the 5511 to indicate that the PCI arbiter has granted the use of the PCI bus to the 5513.
9	ROMKBCS#	O	Keyboard or System ROM Chip Select
169	SPKR	O	Speaker is the output for the speaker.
158	SDIR#	O	SD Low Byte Data Direction controls the direction of the low byte buffer between SD and XD. A high sets the data path direction from XD to SD and a low sets the data path direction from SD to XD.
19	FERR#	I	Floating point error from the CPU. It is driven active when a floating point error occurs.
17	IGNNE#	OD	IGNNE# is normally in high impedance state, and is asserted to inform CPU to ignore a numeric error. A resistor connected to 3.3V is required to maintain a correct voltage level to CPU.
18	NMI	OD	Non-maskable interrupt is rising edge trigger signal to the CPU and is generated to invoke a non-maskable interrupt. Normally, this signal is low. It goes high impedance state when a non-maskable interrupt source comes up. An external pull up resistor is required to be directly connected to CPU.
16	INTR	OD	Interrupt goes high impedance whenever a valid interrupt request is asserted. Hence, an external pull up resister is required to be directly connected to the CPU's interrupt pin.
24, 64, 84, 124, 44, 184	VCC5		+5V DC power
178, 74, 202, 166, 136, 108, 30, 38, 48, 58, 68, 80	GND		Ground

4.5 Electrical Characteristics

4.5.1 Absolute Maximum Ratings

Table 4-11

Parameter	Min	Max	Unit
Ambient operating temperature	0	70	^{o}C
Storage temperature	-40	125	^{o}C
Input voltage	-0.3	5.5	V
Output Voltage	-0.5	5.5	V
Power Dissipation		1	W

Note:
Stress above these listed may cause permanent damage to device. Functional operation of this device should be restricted to the conditions described under operating conditions.

4.5.2 DC Characteristics

TA = 0 - 70 ^{o}C, VSS = 0V , VDD=5V$\pm$5%

Table 4-12

Symbol	Parameter	Min	Max	Unit	Condition
V_{IL}	Input Low voltage	-0.3	0.8	V	
V_{IH}	Input High Voltage			V	
V_{OL}	Output Low Voltage		0.45	V	
V_{OH}	Output High Voltage	2.4		V	
I_{OL1}	Output Low Current	4		mA	***Note 1***
I_{OH1}	Output High Current	4		mA	***Note 1***
I_{OL2}	Output Low Current	8		mA	***Note 2***
I_{OH2}	Output High Current	8		mA	***Note 2***
I_{OL3}	Output Low Current	8,12		mA	***Note 3***
I_{OH3}	Output High Current	8,12		mA	***Note 3***
I_{OL4}	Output Low Current	6		mA	***Note 4***
I_{OH4}	Output High Current	6		mA	***Note 4***
I_{IL}	Input Leakage Current		+10	mA	
I_{OL}	Output Leakage Current		-10	mA	
C_{IN}	Input Capacitance		12	pF	
C_{OUT}	Output Capacitance		12	pF	
$C_{I/O}$	I/O Capacitance		12	pF	

Note:

1. I_{OL1} and I_{OH1} are applicable to ROMKBCS#, RTCWR/IDECYC#, RTCRD/PSRSTB#, RTCALE/PWRGD, WAKEUP0, WAKEUP1, PHOLD#, PAR, C/BE[3:0]#, AD[31:0], SDIR#, XD[7:0], SPKR, BCLK, DAK[2:0], MDAK[1:0]#, INT, NMI, IGNEE#.
2. I_{OL2} and I_{OH2} are applicable to RFH#, CHRDY, DIOCHRDY, AEN, BALE, EOP, SD[15:8].

3. I_{OL3} and I_{OH3} are applicable to SMWTC#, SMRDC#, MWTC#, MRDC#, SBHE#, LA[23:20], IDECS[1:0]#, LA[21:17], SA[16:0], IOWC#, IORC#, DIORC[1:0]#, DIOWC[1:0]#, Please refer to Register description.

4. I_{OL4} and I_{OH4} are applicable to FRAME#, IRDY#, TRDY#, DEVSEL#, STOP#.

4.5.3 AC Characteristics

The AC characteristic is measured under the following capacitive condition.

Capacitive load	Pin
35pf	BCLK, DAK[0:2], BALE, AEN, NMI, SDIR, EOP, SPKR, INT
50pf`	FRAME#, IRDY#, TRDY#, DEVSEL#, STOP#, C/BE[3:0]#, XD[7:0]
150pf	SD[15:8], SBHE#, RFH#, CHRDY, MWTC#, MRDC#, IORC#, IOWC#, SA[19:0], LA[23:17]

Table 4-13

Para-meter	Description	Min	Typ	Max	Unit
ISA Bus Interface Signals (Figure 4.9)					
	BCLK High		63.2		ns
	BCLK Low		56.8		ns
t1	BALE valid delay from BCLK		4.5	7	ns
t2	IORC#, IOWC#, MRDC#, MWTC# valid delay from BCLK		16.5	24	ns
t3	IORC#, IOWC#, MRDC#, MWTC# invalid delay from BCLK		12	18	ns
t5a	M16# setup time to BCLK rising		15		ns
t5b	M16# hold time from BCLK rising		6		ns
t6a	IO16# setup time to BCLK falling		19		ns
t6b	IO16# hold time from BCLK falling		6		ns
t7	16 bit IORC#, IOWC# pulse width		1.5		BCLK
	8 bit IORC#, IOWC# pulse width		4.5		BCLK
	16 bit MRDC#, MWTC# *1		2		BCLK
	8 bit MRDC#, MWTC#		4.5		BCLK
	ROM MRDC#, MWTC# *1		2		BCLK
Data Buffer Interface					
t8	SD, XD data set up time to IORC#, MRDC# inactive	10			ns
t9	SD, XD data hold time to IORC#, MRDC# inactive	3			ns
t10	SD, XD valid data delay from IOWC#, MWTC# active (for data swapping)	15	22		ns
t11a	SD, XD data hold time from IOWC#, MWTC# inactive in write disassembly cycle	15	22		ns

t11b	SD, XD data hold time from IOWC#, MWTC# inactive in write cycle		172		ns
t12	SD, XD valid to IOWC#, MWTC# active		142		ns
t13	SDIR deassertion to IORC#, MRDC# active (16-bit)		1	2	BCLK
	SDIR deassertion to IORC#, MRDC# active (8-bit)	1.5		2.5	BCLK
t14	SDIR assertion delay from IORC#, MRDC# inactive		2		BCLK
Address Buffer Interface					
t15	SA, LA propagation delay from PCICLK in Frame# address phase		34	51	ns
t16	SA0, SA1, SBHE# hold time from the negation of IORC#, IOWC#, MWTC#, MRDC#	10			ns
t17a	CHRDY setup time to BCLK rising		15.2		ns
t17b	CHRDY hold time to BCLK rising	14.8			ns
t44	ZWS# setup time to BCLK faling		10		ns
t45	ZWS# hold time to BCLK faling	20			ns
DMA Compatible Timings (Figure 4.10, 4.11)					
t18	DAK active to IORC# active		0.5		DMACLK
t19	DAK acive to IOWC# active		1.5		DMACLK
t20	DAK active hold from IORC# inactive		0.5		DMACLK
t21	DAK active hold from IOWC# inactive		0.5		DMACLK
t22a	AEN active to IORC# active		6		DMACLK
t22b	AEN active to IOWC# active		7		DMACLK
t23a	AEN inactive from IORC# inactive		3		DMACLK
t23b	AEN inactive from IOWC# inactive		4		DMACLK
t24a	BALE active to IORC# active		1.5		DMACLK
t24b	BALE active to IOWC# active		2.5		DMACLK
t25a	BALE inactive from IORC# inactive		1		DMACLK
t25b	BALE inactive from IOWC# inactive		1		DMACLK
t26a	LA, SA, SBHE# valid set up time to IORC#		1		DMACLK
t26b	LA, SA, SBHE# valid set up time to IOWC#		2		DMACLK
t27a	LA, SA, SBHE# valid hold from IORC#		0.5		DMACLK
t27b	LA, SA, SBHE# valid hold from IOWC#		0.5		DMACLK
t28	IORC# pusle width		4		DMACLK
t29	IOWC# pusle width		2		DMACLK
t30	MRDC# pusle width		3		DMACLK

t31	MWTC# pusle width		3		DMACLK
t32	MWTC# active from IORC# active		1		DMACLK
t33	IOWC# active from MRDC# active		1		DMACLK
t34	MWTC# inactive from IORC# inactive		0		ns
t35	IOWC# inactive from MRDC# inactive		1.5		ns
t36	Read data valid from IORC# active		267.5		ns
t37	Read data valid hold from IORC# inactive		32.2		ns
t38	Write data valid setup to IOWC# inactive		162.5		ns
t39	Write data valid hold from IOWC# inactive		13.2		ns
t40	EOP active delay from IOWC# active		-7.6		ns
t41	EOP active delay from IORC# active		112.3		ns
t42	EOP active delay from IOWC# inactive		0.7		ns
t43	EOP active delay from IORC# inactive		0.8		ns
Note: DMACLK = BCLK or BCLK/2 depends on bit 0 of ISA configuration register 01H.					
Refresh Timing (Figure 4.20)					
t44	RFH# active setup to MRDC# active		2		BCLK
t45	RFH# active hold from MRDC# inactive		0.5		BCLK
t46	AEN active to RFH# active delay		3		ns
Miscellaneous Timing (Figure 4.12 ~ 4.16, 4.18, 4.19)					
t47	SERR#, IOCHK# active to NMI output floating active			200	ns
t48	INT output floating delay from IRQ active			100	ns
t49	IRQ active pulse width	100			ns
t50	IGNEE# active from IOWC# active for port F0h access			220	ns
t51	IGNEE# inactive from FERR# inactive			150	ns
t52	SPKR valid delay from OSC timing			200	ns
t53	RTCALE pulse width		532.3		ns
t54	RTCALE active from IORW# active		4		ns
t54a	RTCWR active from IOWR# active		5		ns
t54b	RTCRD active from IORD# active		5		ns
t54c	RTCWR inactive from IOWR# inactive	3.5	5	10	ns
t54d	RTCRD inactive from IORD# inactive	3.5	5	10	ns

***1: No command delay**

PCI Bus AC Specifications (Figure 4.17) The following parameters are applicable to AD[31:0], C/Be[3:0], FRAME#, TRDY#, IRDY#, STOP#, LOCK#, IDSEL#, DEVSEL#, PAR, and SERR#.					
		Min	**Typ**	**Max**	**Units**
t57	Signal valid delay from PCICLK rising edge *2			11	ns
t58	Signal invalid delay from PCICLK rising edge	2			ns
t59	Hi-impedance to Active delay from PCICLK rising edge	2			ns
t60	Active to Hi-impedance delay from PCICLK rising edge			28	ns
t61	Setup time of input signal	7			ns
t62	Hold time of input signal	0			ns

***2: This parameter is measured under 50pf.**

PCI IDE Timing (Figure 4.21 ~ 4.22)					
Para-meter	**Description**	**Min**	**Typ**	**Max**	**Unit**
t63	Write Active Time	1		12	PCICLK
t64	Write Recovery Time	1		13	PCICLK
t65	Write Cycle Time (Post Write Buffer Enable)		5		PCICLK
t66	Read Active Time	1		12	PCICLK
t67	Read Recovery Time	1		13	PCICLK
t68	Read Cycle Time (Prefetch Buffer Enable)	3	5		PCICLK

4.5.4 AC Timing Diagram

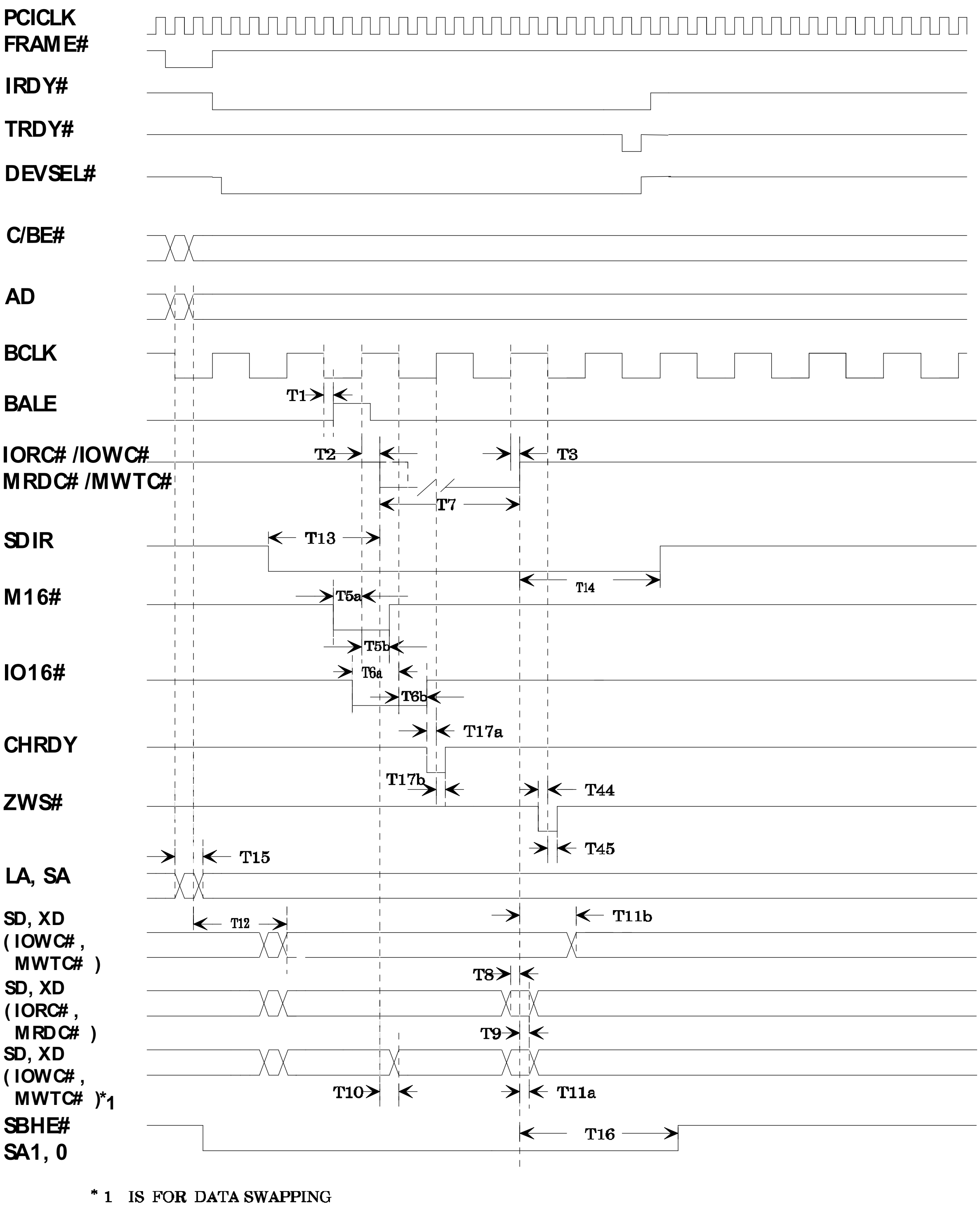
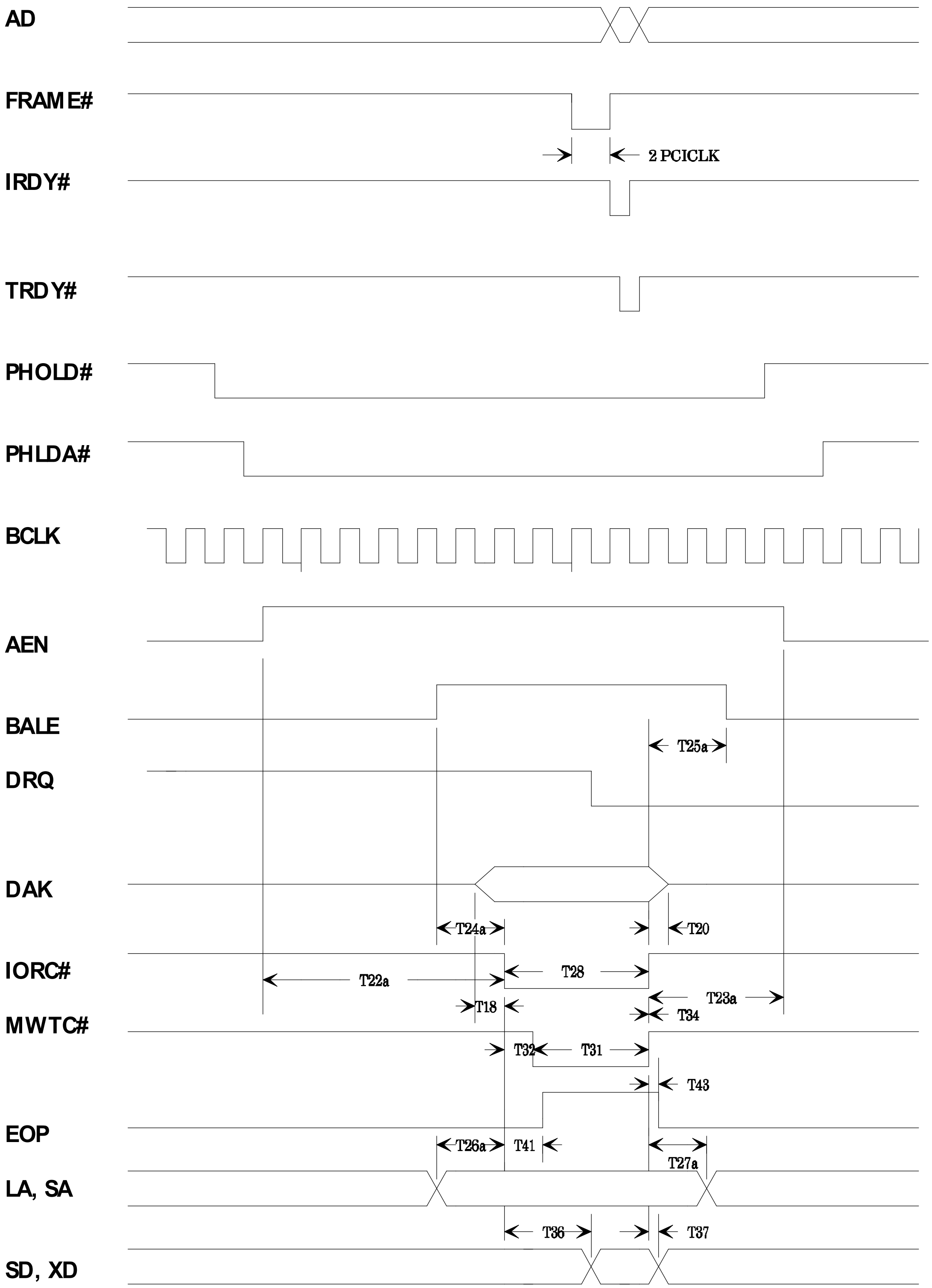
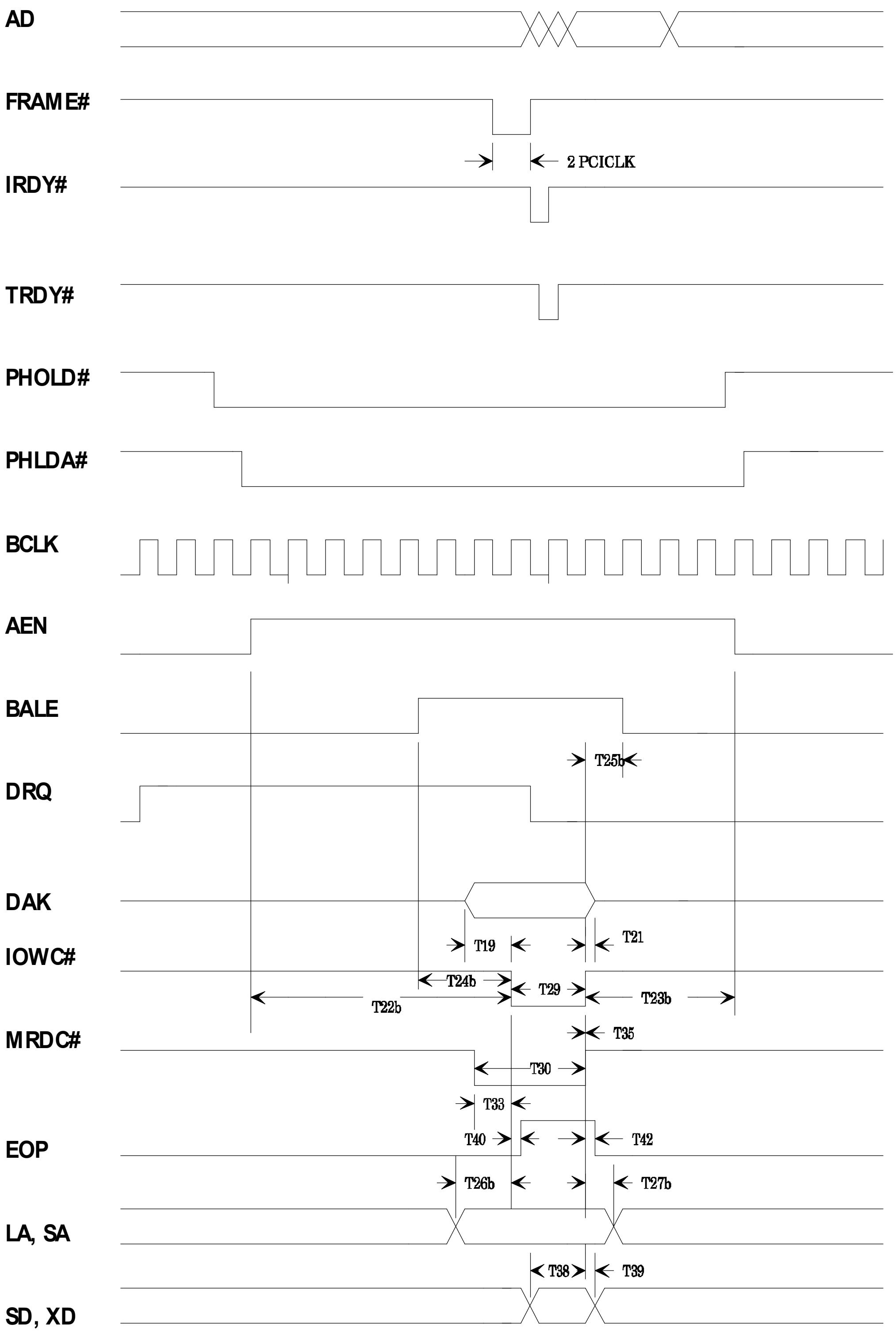


Figure 4.10 PCI to AT Bus Cycle



DMA cycle (IOWC#, MRDC#), DMACLK = BCLK

Figure 4.11 DMA Cycle (IOWC#, MRDC#)



DMA cycle (IORC#, MWTC#), DMACLK = BCLK

Figure 4.12 DMA Cycle (IORC#, MWTC#)

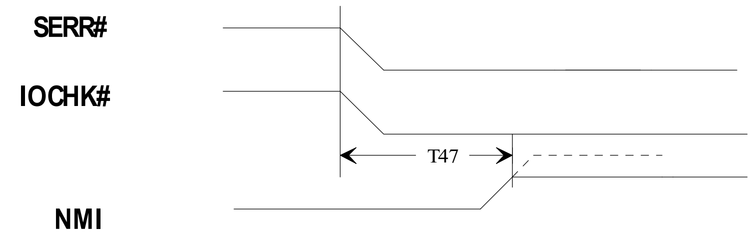


Figure 4.13

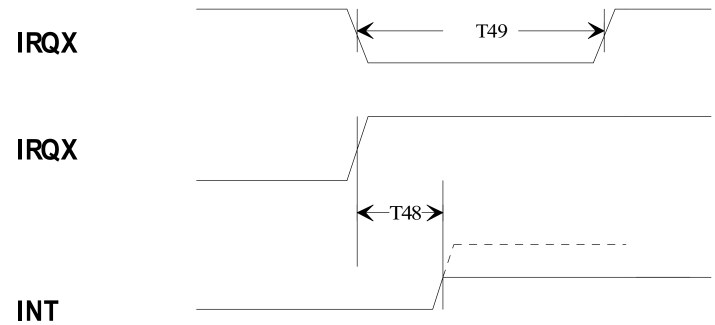


Figure 4.14

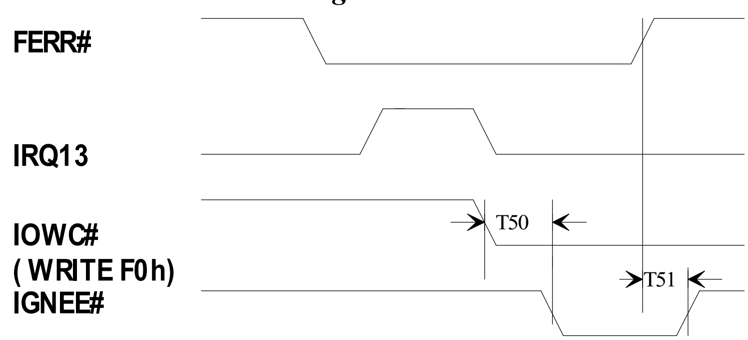


Figure 4.15

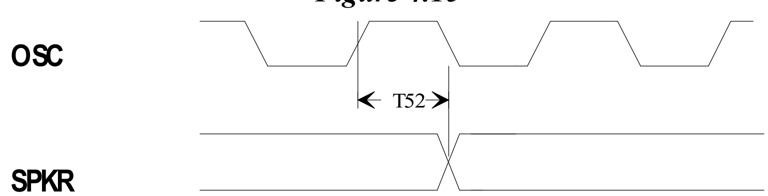


Figure 4.16

Error! Objects cannot be created from editing field codes.

Figure 4.17

Error! Objects cannot be created from editing field codes.

Figure 4.18

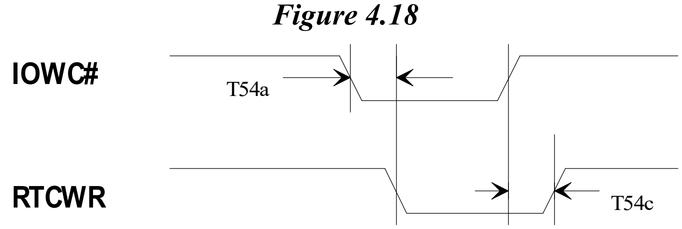


Figure 4.19

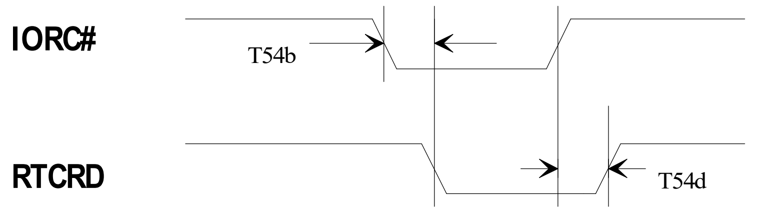


Figure 4.20

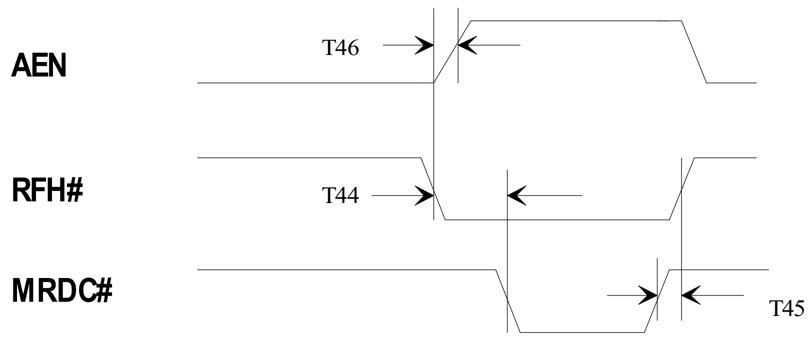


Figure 4.21

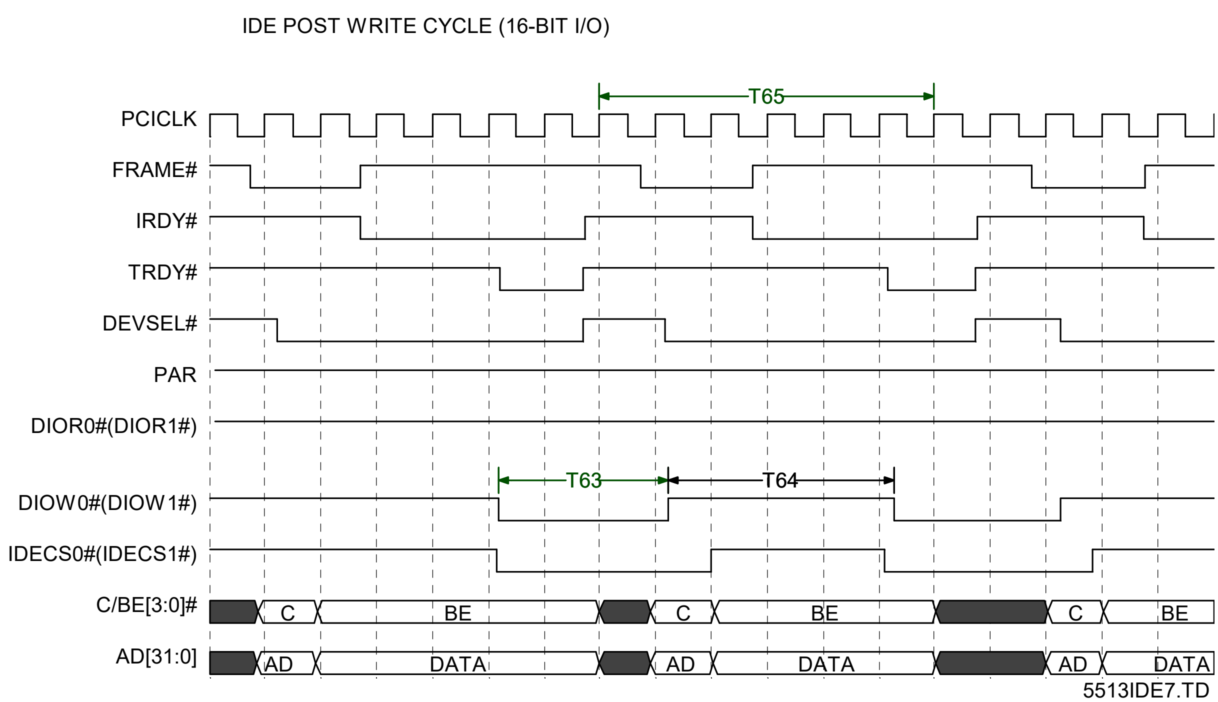


Figure 4.22 IDE POST Write Cycle (16-Bit I/O)

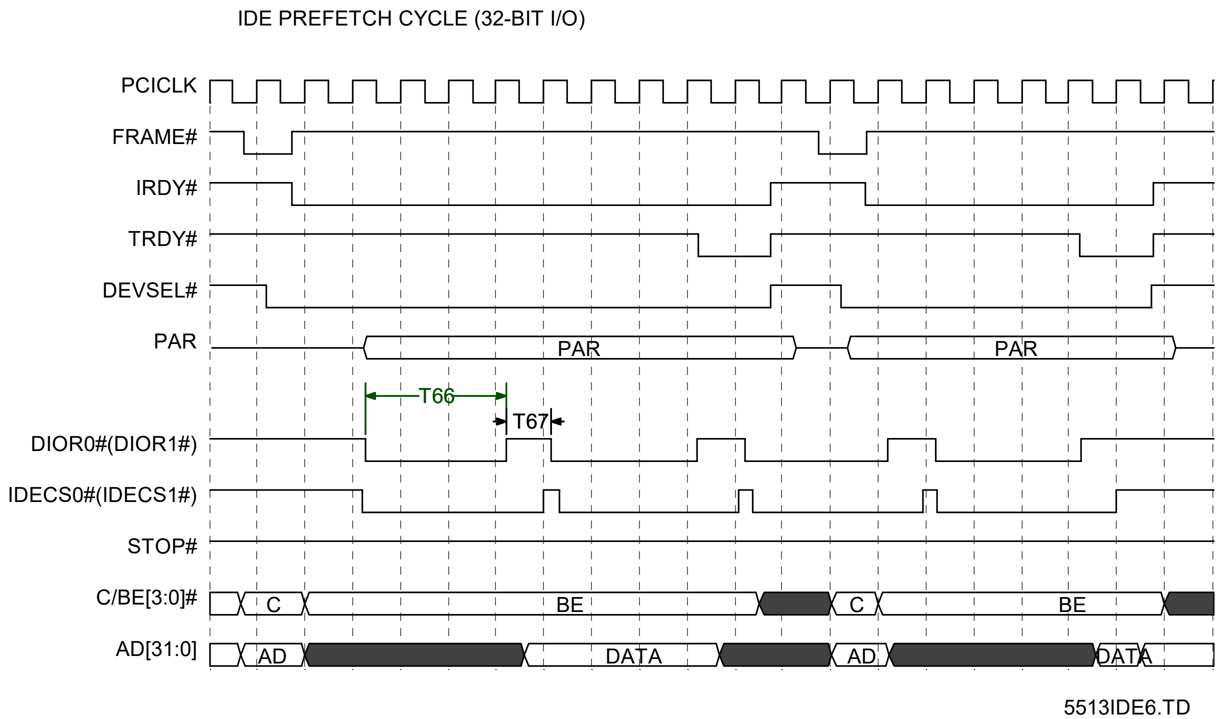


Figure 4.23 IDE Prefetch Cycle (32-Bit I/O)

5. Mechanical Dimension

5.1 SiS5511, SiS5512, SiS5513 (208 pins)

QFP208-P (208-Pin Plastic Flat Package) Unit:mm

Error! Objects cannot be created from editing field codes.

Figure 4.24

COPYRIGHT NOTICE

`

TRADEMARKS

SiS is a registered trademark of Silicon Integrated Systems Corp.

All brands or product names mentioned are trademarks or registered trademarks of their respective holders.

DISCLAIMER

Silicon Integrated Systems Corp. makes no representations or warranties regarding the contents of this manual. We reserve the right to revise the manual or make changes in the specifications of the product described within it at any time without notice and without obligation the notify any person of such revision or change.

The information contained in this manual is provided for general use by our customers. Our customers should be aware that the personal computer field ins the subject of many patents. Our customers should ensure that they take appropriate action so that their use of our products does not infringe upon any patents. It is the policy of Silicon Integrated Systems Corp. To respect the valid patent rights of third parties and not to infringe upon or assist others to infringe upon such rights.

RESTRICTED RIGHTS LEGEND

Use, duplication, or disclosure by the Government is subject to restrictions set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at 252.277-7013.