1 5501/5502/5503 Overview

SiS5501 PCI/ISA Cache Memory Controller (**PCMC**)
SiS5502 PCI Local Data Buffer (**PLDB**)
SiS5503 PCI System I/O (**PSIO**)

A whole set of the SiS5501, 5502, and 5503 provides fully integrated support for the Pentium/P54C PCI/ISA system. The chipset is developed by using a very high level of function integration and system partitioning. With the SiS5501, SiS5502, and SiS5503 chipset, only 12 TTLs (include 3 DRAM address buffer) are required to implement a low cost, high performance, Pentium/P54C PCI/ISA system. Figure 1 shows the system block diagram.

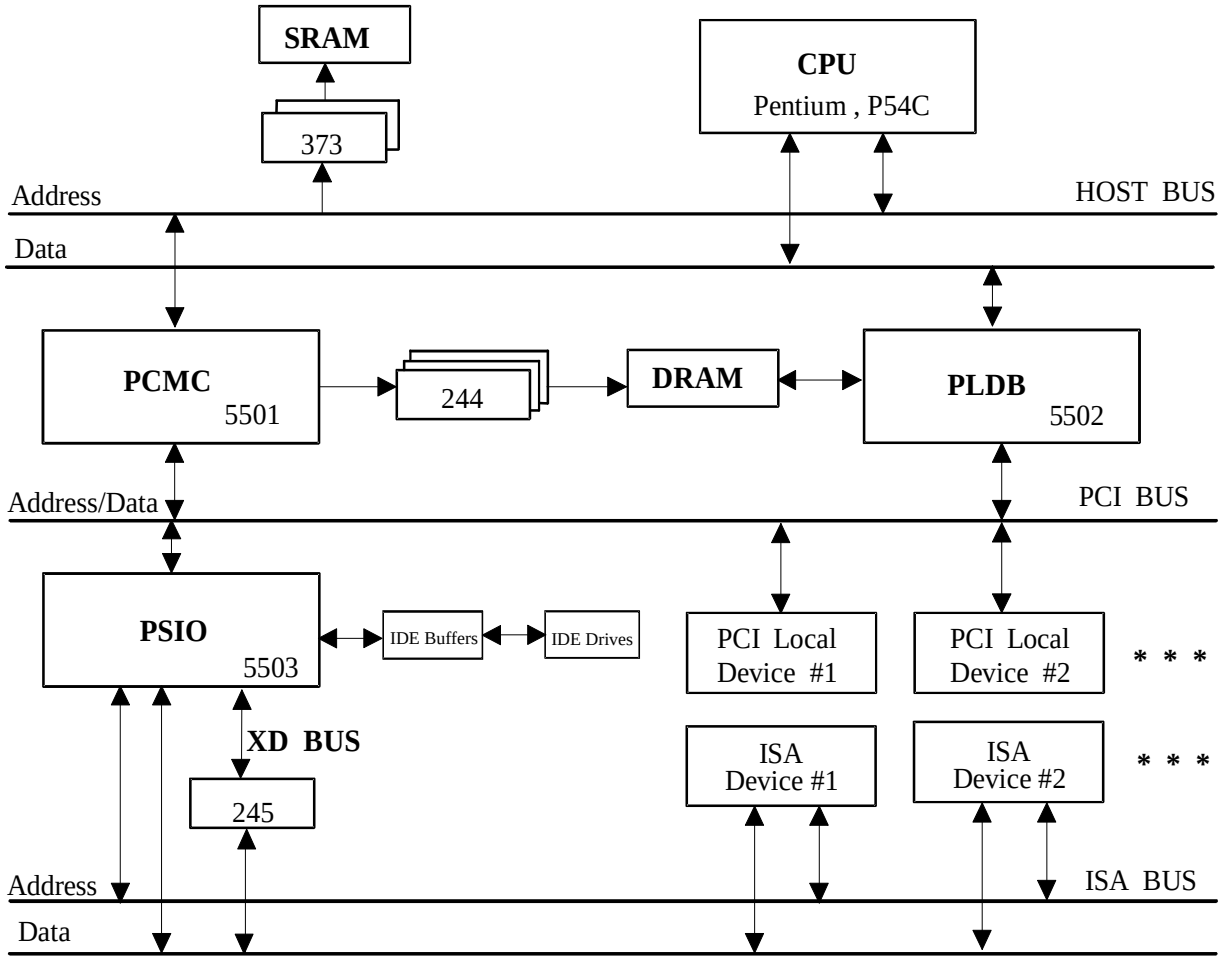


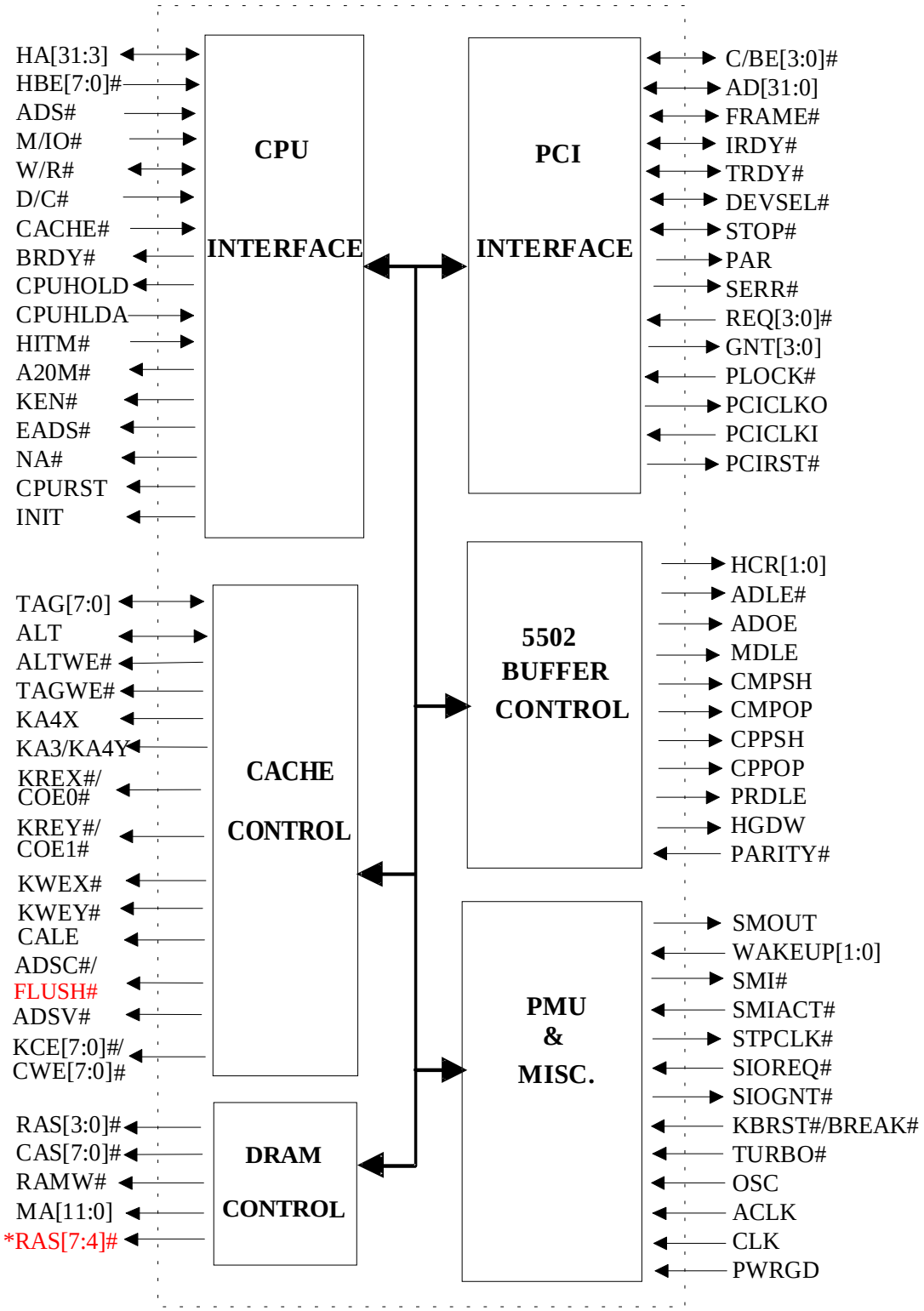
Figure 1.1 System Block Diagram

2. SiS5501

2.1 Features

- **Supports the 510\60, 567\66, 735\90, 815\100 MHz and 75 MHz Pentium Processor**
- **Supports M1 and Other Pentium Compatible CPU**
- **Supports the Pipelined Address Mode of the Pentium or the P54C Processor**
- **Integrated Second Level (L2) Cache Controller**
  - Write Through and Write Back Cache Modes
  - 8 bits or 7 bits Tag with Direct Mapped Organization
  - Supports Standard and Burst SRAMs
  - Supports 64 KBytes to 2 MBytes Cache Sizes
  - 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 SRAMs at 66 MHz
- **Integrated DRAM Controller**
  - Supports 8 Banks of SIMMs up to 512 MBytes of Cacheable Main Memory
  - Supports " Table- Free " DRAM Configuration
  - Concurrent Write Back
  - CAS#-before-RAS# Transparent DRAM Refresh
  - Supports 256K/512K/1M/2M/4M/16M xN 70ns Fast Page Mode and EDO DRAM
  - The Fastest Burst Cycle Speed for FP and EDO are 6-3-3-3 and 6-2-2-2 respectively
  - Programmable CAS# driving Current
  - Programmable DRAM Speed
- **Two Programmable Non-Cacheable Regions**
- **Option to Disable Local Memory in Non-Cacheable Regions**
- **Shadow RAM in Increments of 16 KBytes**
- **Supports Pentium/P54C SMM Mode**
- **Supports CPU Stop Clock**
- **Provides High Performance PCI Arbiter**
  - Supports Four PCI Masters
  - Supports Rotating Priority Mechanism
  - Hidden Arbitration Scheme Minimizes Arbitration Overhead
- **Integrated PCI Bridge**
  - 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
  - PCI Burst Write in the Pace of X-2-2-2-....
  - PCI Burst Read L2 Cache in X-2-2-2-....
  - PCI Burst Read DRAM in X-3-2-3-2-....
  - Cache Snoop Filters Ensure Data Coherency and Minimize Snoop Frequency
  - Meet PCI Specification Buffer Strength
- **208-Pin PQFP Package**
- **0.6µm CMOS Technology**

2.2 Functional Block Diagram



SiS5501 Functional Block Diagram

* : Multi-function pin

2.3 General Description

The SiS5501(PCMC) bridges between the host bus and the PCI local bus. The SiS5501 (PCMC) monitors each cycle initiated by the CPU, and forwards it to the PCI bus if the CPU cycle does not target the local memory. For the CPU or the PCI bus to the local memory cycles, the built-in Cache and DRAM Controller assume control to the secondary cache, DRAMs, and the SiS5502 (PLDB). The SiS5501 (PCMC) also guides the SiS5502 (PLDB) for correct data flow. All of the Green PC functions are provided.

2.4 CPU Interface

The SiS5501 is designed to support Pentium/P54C CPU host interface at 66.667/60/50 MHz. The host data bus and the DRAM bus are 64-bit wide.

The SiS5501 supports the pipelined addressing mode of the Pentium/P54C CPU by issuing the next address signal, NA#. NA# is only generated in two cases:
a) burst read L2 cache or DRAM, and
b) single read DRAM.

The PCMC supports the CPU L1 write back(WB) or write through(WT) cache and the PCMC L2 WB or WT cache. 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/P54C CPU in response to the assertion of PCI master requests(REQ[3:0]#, and SIOREQ#). Upon receiving the CPUHLDA from the CPU, it does not immediately assert GNT[3:0]# or SIOGNT# 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.

2.5 Cache Controller

The built-in L2 Cache Controller uses a direct-mapped, bank-interleaved/non-interleaved scheme, which can be configured as either in the write through or write back mode. Both standard and burst SRAMs are supported.

Table 1 shows the cache sizes that are supported by the SiS5501, with the corresponding TAG RAM sizes, data RAM sizes, and cacheable memory sizes. Tables 2 and 3 summarize the performance and options when either the standard SRAMs or the Burst SRAMs are used.

Table 1

Cache Size	Data RAM	Tag RAM	Alter RAM	Cacheable Size	Interleaved
64K	8Kx8x8	2Kx8	2Kx1	16M	No
128K	8Kx8x16	4Kx8	4Kx1	32M	Yes
256K	32Kx8x8	8Kx8	8Kx1	64M	No
512K	32Kx8x16	16Kx8	16Kx1	128M	Yes
512K	64Kx8x8	16Kx8	16Kx1	128M	No
1M	128Kx8x8	32Kx8	32Kx1	256M	No
1M	64Kx8x16	32Kx8	32Kx1	256M	Yes
2M	128Kx8x16	64Kx8	64Kx1	512M	Yes

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) Non-cacheable Area defined in register 57h, 58h, 59h and 5Ah and
3) C, D, E, F Segment Cachability defined in register 53h, 54h, 55h, and 56h.

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 Burst SRAM Speed Setting

Cycle type	66,60 MHz	50MHz
Burst read	3/**4**-1-1-1 3/4-2-2-2	3/**4**-1-1-1 3/4-2-2-2
Burst write	3/**4**-1-1-1 3/4-2-2-2	3/**4**-1-1-1 3/4-2-2-2
Single read	3/**4**	3/**4**
Single write	3/**4**	3/**4**

Note :

1: The burst SRAM speed for 66/60 MHz is 9 ns. For 50MHz, it is 12 ns.
2: **X**-Y-Y-Y is the recommended setting.

Table 3 Asynchronous SRAM speed setting (apply to read and write cycle)

	66 MHz		60 MHz		50MHz	
cache configuration	Tag	Data	Tag	Data	Tag	Data
3-1-1-1 interleave	---	---	---	---	15ns	15ns
3-1-1-1 non-interleave	---	---	---	---	---	---
3-2-2-2 interleave	15ns	15ns	15ns	15ns	20ns	20ns
3-2-2-2 non-interleave	15ns	15ns	15ns	15ns	20ns	20ns
3-3-3-3 interleave	15ns	15ns	15ns	15ns	20ns	20ns
3-3-3-3 non-interleave	15ns	15ns	15ns	15ns	20ns	20ns
4-1-1-1 interleave	15ns	12ns	15ns	12ns	20ns	15ns
4-1-1-1 non-interleave	---	---	---	---	---	---
4-2-2-2 interleave	15ns	15ns	20ns	20ns	20ns	20ns
4-2-2-2 non-interleave	15ns	15ns	20ns	20ns	20ns	20ns
4-3-3-3 interleave	15ns	20ns	20ns	20ns	20ns	20ns
4-3-3-3 non-interleave	15ns	20ns	20ns	20ns	20ns	20ns
5-1-1-1 interleave	20ns	12ns	20ns	12ns	20ns	15ns
5-1-1-1 non-interleave	---	---	---	---	---	---
5-2-2-2 interleave	20ns	15ns	20ns	20ns	20ns	20ns
5-2-2-2 non-interleave	20ns	15ns	20ns	20ns	20ns	20ns
5-3-3-3 interleave	20ns	20ns	20ns	20ns	20ns	20ns
5-3-3-3 non-interleave	20ns	20ns	20ns	20ns	20ns	20ns

2.6 DRAM Controller

The 5501 can support 8 rows of DRAM, and memory size from 2 MBytes up to 512 MBytes. Each populated bank could be single or double sided 64 bits FP DRAM or EDO (Extended Data Output) DRAM. It is also permissible to mix FP DRAM bank and EDO DRAM bank without any order. The installed DRAM type can be 256K x 36, 512K x 36, 1M x 36, 2M x 36, 4M x 36 or 16M x 36 SIMMs. However, since RAS 5 shares the same signal of MA 11, Bank 5 should be excluded if 16 M x 36 DRAM is used.

DBR 8~0 (DRAM Boundary Register, register 79h and 77h~70h) are used to configure the total amount of memory. In DBR 7~0, bit 7~0 corresponds to host address 28~21 and DBR 8 bit 7~0 is used to compare against the host address 29 of bank 7~0. Contents in these registers reflect the boundary address, that means the value programmed to the last DBR will be the DRAM size in the system.

DBR0 (Reg. 70) = Total amount of memory in Bank 0
DBR1 (Reg. 71) = Total amount of memory in Bank 0 + Bank 1
DBR2 (Reg. 72) = Total amount of memory in Bank 0 + Bank 1 + Bank 2
.....
DBR7 (Reg. 77) = Total amount of memory in Bank 0 + ...+ Bank 7

The following 2 examples show how the DBR registers be used to determine the memory size.

Example 1:

The system memory is populated as 2 banks of single-sided 1M x 36 DRAM, which are located at Bank 1 and 3. This yields 16 M Bytes DRAM totally. The DBR registers are programmed as follows:

```
DBR0 = 00h   DBR 8 bit 0 = 0    ; empty                    ; 0 M Byte totally
DBR1 = 04h   DBR 8 bit 1 = 0    ; 8 M Bytes for Bank 1     ; 8 M Bytes totally
DBR2 = 04h   DBR 8 bit 2 = 0    ; empty                    ; 8 M Bytes totally
DBR3 = 08h   DBR 8 bit 3 = 0    ; 8 M Bytes for Bank 3     ; 16 M Bytes totally
DBR4 = 08h   DBR 8 bit 4 = 0    ; empty                    ; 16 M Bytes totally
DBR5 = 08h   DBR 8 bit 5 = 0    ; empty                    ; 16 M Bytes totally
DBR6 = 08h   DBR 8 bit 6 = 0    ; empty                    ; 16 M Bytes totally
DBR7 = 08h   DBR 8 bit 7 = 0    ; empty                    ; 16 M Bytes totally
```

Example 2:

The system memory is populated as 4 banks of single-sided 16 M x 36 DRAM, which are located from Bank 0 to 3. This yields 512 M Byte DRAM totally. The DBR registers are programmed as follows:

```
DBR0 = 40h   DBR 8 bit 0 = 0    ; 128 M Bytes for Bank 0   ; 128 M Bytes totally
DBR1 = 80h   DBR 8 bit 1 = 0    ; 128 M Bytes for Bank 1   ; 256 M Bytes totally
DBR2 = C0h   DBR 8 bit 2 = 0    ; 128 M Bytes for Bank 2   ; 384 M Bytes totally
DBR3 = 00h   DBR 8 bit 3 = 1    ; 128 M Bytes for Bank 3   ; 512 M Bytes totally
DBR4 = 00h   DBR 8 bit 4 = 1    ; empty                    ; 512 M Bytes totally
DBR5 = 00h   DBR 8 bit 5 = 1    ; empty                    ; 512 M Bytes totally
DBR6 = 00h   DBR 8 bit 6 = 1    ; empty                    ; 512 M Bytes totally
DBR7 = 00h   DBR 8 bit 7 = 1    ; empty                    ; 512 M Bytes totally
```

The 12-bit multiplexed row/column address MA[11:0] allows the PCMC to support 256K, 1M, 4M, and 16M 70ns fast page mode DRAMs.

Table 4. shows the corresponding request address bits used in column address and row address for the DRAM.

Table 4. MA Generation Table

Body Type	256k		512k		1M		2M		4M		16M	
MA	CAS	RAS	CAS	RAS	CAS	RAS	CAS	RAS	CAS	RAS	CAS	RAS
MA0	A3	A12	A3	A12	A3	A22	A3	A22	A3	A22	A3	A22
MA1	A4	A13	A4	A13	A4	A13	A4	A13	A4	A24	A4	A24
MA2	A5	A14	A5	A14	A5	A14	A5	A14	A5	A14	A5	A26
MA3	A6	A15	A6	A15	A6	A15	A6	A15	A6	A15	A6	A15
MA4	A7	A16	A7	A16	A7	A16	A7	A16	A7	A16	A7	A16
MA5	A8	A17	A8	A17	A8	A17	A8	A17	A8	A17	A8	A17
MA6	A9	A18	A9	A18	A9	A18	A9	A18	A9	A18	A9	A18
MA7	A10	A19	A10	A19	A10	A19	A10	A19	A10	A19	A10	A19
MA8	A11	A20	A11	A20	A11	A20	A11	A20	A11	A20	A11	A20
MA9	NA	NA	NA	A21	A12	A21	A12	A21	A12	A21	A12	A21
MA1	NA	NA	NA	NA	NA	NA	NA	A23	A13	A23	A13	A23
0	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	A14	A25
MA1												
1												

To improve the CPU write DRAMs performance, there is a one level built-in CPU-to-Memory posted write buffer with 4 QWs deep (CTMPB). All the single writes and the burst writes are buffered. In the CPU read miss/line fill cycle, the write-back data from the L2 cache are also buffered into the CTMPB. At the same time, the PCMC starts reading from the DRAMs. The buffered data are written to the DRAMs when the read cycle completes. With this concurrent write back policy, many wait states are eliminated. However, any other cycle targeting the DRAMs will be suspended until the CTMPB is empty.

Table 5 outlines the read and write DRAM cycle performance based on 70ns DRAMs.

Table 5 DRAM Performance

Cycle type	66,60 MHz	50MHz	DRAM Type
read (page hit/row miss /page miss)	**6/9/12-3-3-3** 7/10/13-4-4-4 6/9/12-2-2-2 7/10/13-2-2-2	**6/9/12-3-3-3** 7/10/13-4-4-4 6/9/12-2-2-2 7/10/13-2-2-2	standard page mode standard page mode EDO EDO
posted write (CPU --> Buffer)	**3/4/5-1-1-1** 3/4/5-2-2-2 3/4/5-3-3-3	**3/4/5-1-1-1** 3/4/5-2-2-2 3/4/5-3-3-3	standard page mode, EDO standard page mode, EDO standard page mode, EDO
write retire rate (Buffer --> DRAM)	**3**/4/5 2	**3**/4/5 2	standard page mode EDO

Note:
1: **X-Y-Y-Y** is the recommended setting.

Table 6 DRAM speed setting based on 70ns DRAMs (apply to read and write cycle)

	Register	66MHz	60MHz	50MHz
read CAS pulse width	50h bit 7-6	2T	2T	2T
write CAS pulse width	50h bit 5	2T	2T	2T
CAS precharge time [1]	53h bit 7	1T/2T	1T/2T	1T
RAS precharge time	53h bit 1	4T	4T	4T
RAS to CAS delay time	53h bit 2	3T	3T	3T
refresh RAS active time	52h bit 0	5T	5T	4T
DRAM write push to CAS delay	5Bh bit 3	1T	1T	1T
EDO DRAM CAS pulse width [2]	7Ch bit 1	1T/2T	1T/2T	1T
EDO DRAM CAS precharge time [2]	7Ch bit 0	1T/2T	1T/2T	1T

Note:

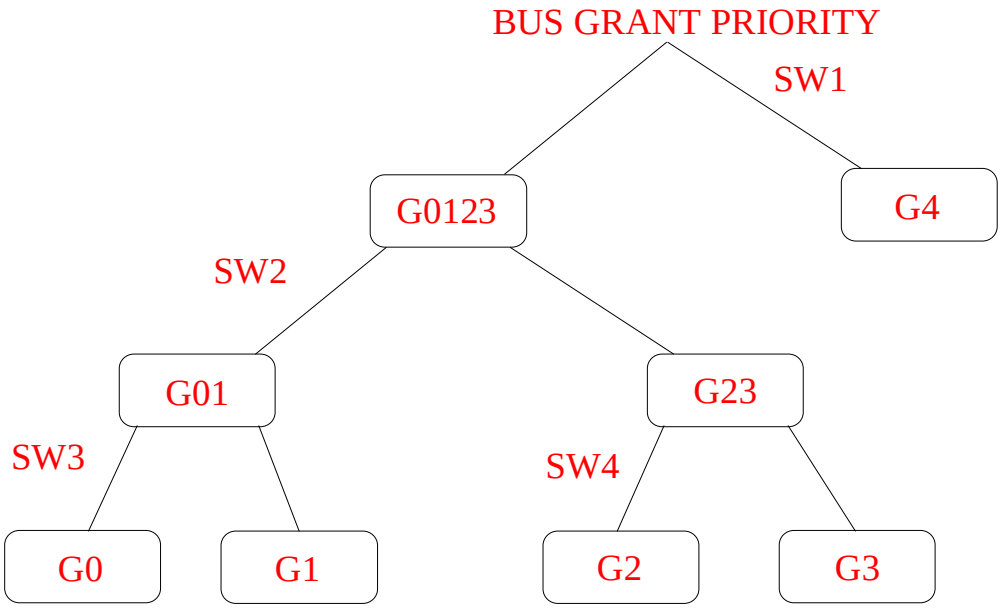
1. The burst DRAM read hit cycle is 6-3-3-3 when the CAS precharge time is 1T. If the CAS precharge time is 2T, the burst DRAM read hit cycle is increased to 7-4-4-4.
2. When EDO type DRAMs are installed and register 7Ch bits[1:0] are set to "11" (1T), the burst DRAM read hit cycle is 6-2-2-2. The standard Fast Page mode DRAM timing is applied, if register 7Ch bits[1:0] are set to "00". In fact, 5501 can detect the EDO type DRAM and applies optimal timing automatically.

2.7 PCI Arbiter

The SiS5501 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 SiS5501 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) access the bus with minimal latency. The PSIO is given a high level of priority to assure compatibility with traditional ISA expansion boards that require short bus latency. 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:



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
   SIOGNT# is asserted
   SW1 is toggled to G0123 (since it has been utilized)
   Priority change to G0, G1, G2, G3, G4

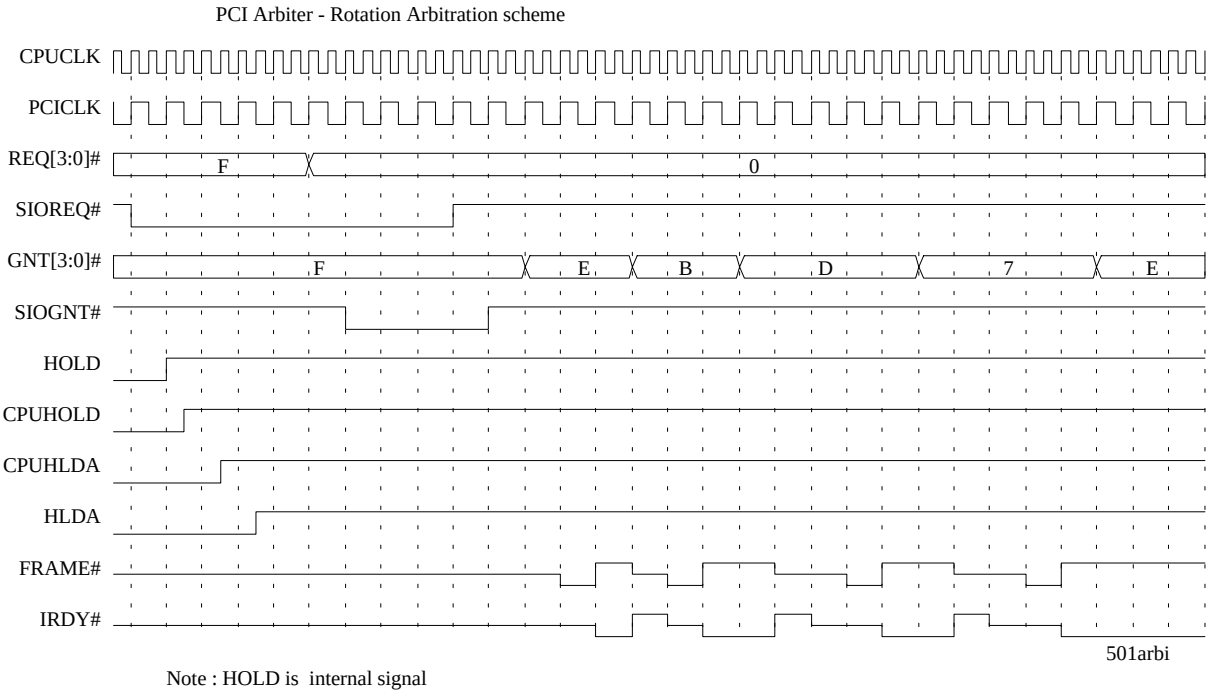
2. PSIO, REQ3, REQ2, REQ1, REQ0 are requesting bus
   GNT0# is asserted
   SW1, SW2 and SW3 are toggled to G4, G23 and G1 respectively (since they have been utilized)
   Priority change to G4, G2, G3, G1, G0

3. REQ3, REQ2, REQ1, REQ0 are active
   GNT2# is asserted
   SW2, 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, 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, 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.



Note : HOLD is internal signal

A PCI master can burst so long as the 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 SiS5501, 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 accesses non-local memory target, the master and target should together have the responsibility of maintaining reasonable latency, but not the system arbiter does.

The PCI arbiter asserts only one GNT# at any time. The 5501 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 desserts GNT#. When this happens, all PCI devices start arbitration again. Note that PSIO is free to this constraint.

The 5501 PCI master will also mask the PSIO request to the arbiter if the PCI LOCK# is asserted to keep ISA master or DMA channels target latency within specification. The 5501 PCI arbiter is also allowed to force system back to CPU each time after SIOREQ# is serviced. This function is disabled by default, and can be enabled by set bit 7 of register 6F in the PCMC Configuration space.

2.8 PCI Bridge

2.8.1 PCI Master Controller

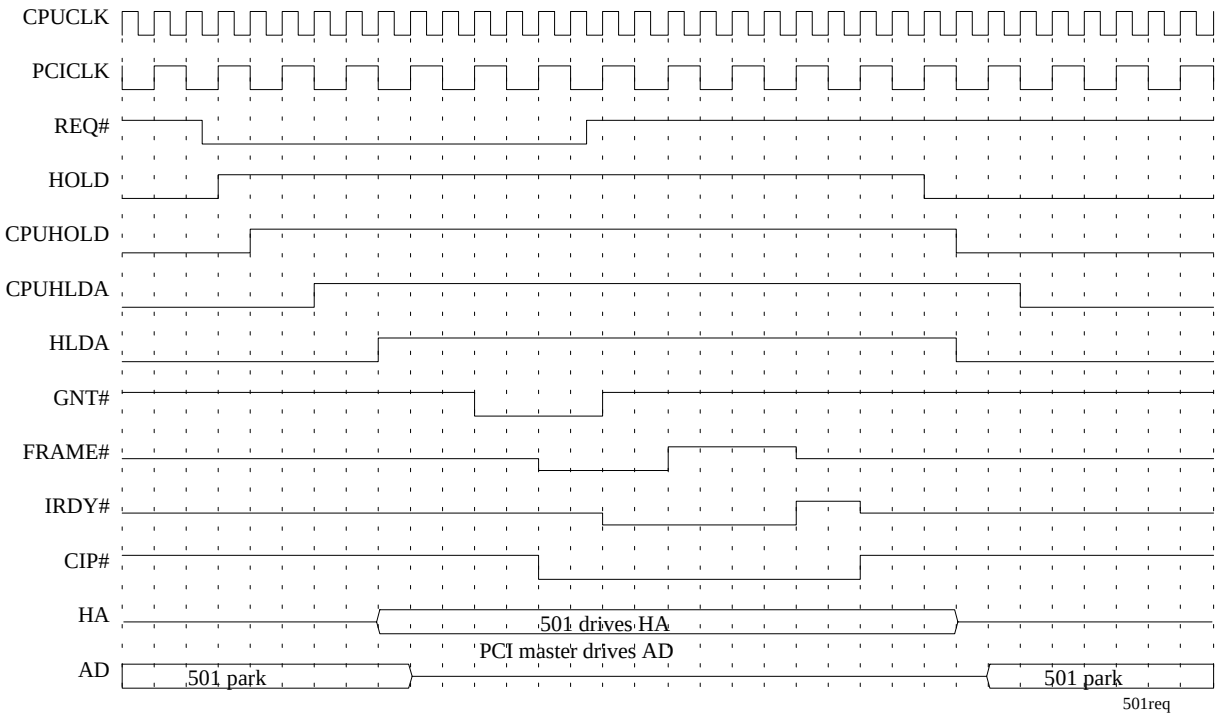
The PCI Master Controller forwards the CPU cycles not targeting the local memory to the PCI bus. In the case of a 64-bit CPU request or a misaligned 32-bit CPU request, the PCMC assumes the read assembly and write disassembly control. A 4 level posted write buffer (CTPPB) is implemented to improve the 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 CTPPB is empty. For PCI bus memory write cycles, the CPU data are pushed into the CTPPB if it is not full. 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 burst transfer rate is always X-2-2-2-... until 128 DWs are exhausted.

The PCI master interface can read data from or write data to the PCI bus at the utmost speed of 1 wait state. This is due to the fact that the PCMC drives the PCI bus address and the PLDB drives the PCI bus data. That necessitates a turn around cycle between the address and the data phases.

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.

2.8.2 PCI Slave Controller

The SiS5501 operates as a slave on the PCI bus whenever a PCI master requests an access to the SiS5501 resource such as Cache, DRAM and the SiS5501 Internal registers. Note that the internal registers can only be accessed by the SiS5501 itself when in CPU cycle. In the SiS550x 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.



Note : HOLD,CIP# (current in progress) are internal signal

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 8 outlines.

Table 7

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)	Data transfer from L2 to PCI
HitM	Miss	Data is first written back from L1 to DRAM. Then, PCI master gets data from DRAM.
HitM	Hit (Dirty or !Dirty)	Data is first written back from L1 to L2. Then, PCI master gets data from L2. The line is marked dirty in the L2.
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)	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.
HitM	Hit (Dirty or !Dirty)	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.

A snoop filter is implemented to prevent the need of multiple inquires to the same line if the line was inquired previously. To support snoop filter, a Snoop Address Latch (SAL) and a Line Comparator are implemented. The line comparator is used to determine if the New Address (NA) is the same as the content of the SAL. If it is not, the NA is loaded into the SAL, and a snoop cycle is issued. In addition, a Valid bit in association with the SAL is used to ensure the snoop filtering is effective only when HLDA is asserted. The simplified filter algorithm is:

1) Write Back Mode

a) if NA=SAL in a PCI master write cycle, the PCMC only issues EADS#. It does not wait for the status of HITM#.
b) if NA=SAL in a PCI master read cycle, no snoop cycle nor EADS# is issued.
c) if NA≠SAL in a PCI master cycle, the PCMC issues a snoop cycle by EADS#, and then monitors the status of HITM#.
d) During a burst transaction, the PCMC automatically generates a snoop cycle when the address advances across a new line.

2) Write Through Mode

In the following two cases, the PCMC only generates EADS#. It ignores the logic of HITM#.
a) if NA=SAL in a PCI master write cycle, and
b) During a burst transaction, the address advances across a new line.

In the SiS550x, the INV signal of P54C should be connected to W/R# that is driven by the SiS5501 in the PCI master cycle. In this way, the SiS5501 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. A burst transfer will be disconnected (retry) if the transfer goes across the 512 bytes(or 1 KBytes selected by Register 5Dh, bit 5) address boundary. This is due to the fact that the address generator, to support the burst transfer, can only address 512 or 1K bytes. In this way, at most 32 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 .

The PCI master writes are buffered in the one QW deep PCI to Memory posted write buffer (PTMPB). The PCMC always packs an aligned QW PCI write data into the write buffer, and then retires it into the DRAM array or the L2 cache. The PCI master write performance, to the utmost, is X-2-2-2- ...

The PCI master reads are through a QW read buffer with which the burst transfers can perform in the pace of X-2-2-2-... (from the L2 cache), or X-3-2-3-2-... (from the DRAMs).

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 SiS550x system, the refresh may be postponed for no more than 24 us in the worst case when a PCI master is reading the whole 32 lines through one burst transaction.

2.8.3 PCI Bus Speed Setting

The following settings apply to all system environment, even though the system is running at 66MHz while the PCI bus is running at 33MHz.

Table 8 PCI bus setting

	Register	Setting	Unit
latency from ADS# to monitor local memory status	5Ch bit 7	2T	CPUCLK
CAS# pulse width in PCI master write cycle	5Ch bit 4	1T	PCICLK
latency from the disarming of "full" to the assertion of BRDY# for the pending CPU to PCI write cycle	5Ch bit 3	1T	CPUCLK
latency from reading L2/DRAM to the assertion of TRDY# in PCI master read cycles	5Dh bit 4	1T	PCICLK
latency from packing one Qword into PTMPB to the assertion of CAS#(or KWE#)	5Dh bit 3	1T	PCICLK
latency from TRDY# to BRDY# in CPU read/write PCI slave cycles	5Dh bit 2	2T	CPUCLK

2.9 Green PC Function

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

2.9.1 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.

2.9.2 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.

2.9.3 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.

2.9.4 Shadow Register

In order to support "suspend to HDD" function, all necessary shadow registers are implemented into 5503. For more detailed information, please refer to "5503 Register Description"

2.10 Configuration Registers

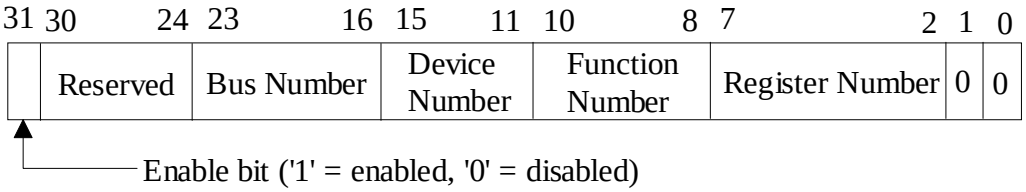
There are two sets of registers in the PCMC, I/O mapped registers and the PCI configuration space registers.

2.10.1 I/O Mapped Registers

The SiS5501 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



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 SiS5501 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 SiS5501, because 5501 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 5501 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 **CPU BIST Enable.**

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 **Select Turbo /DeTurbo Mode**

There are two ways to enter Deturbo mode. One is through software; another is hardware.

- Software Deturbo: Set Reg. 5Bh bit 1 to 1, Reg. 65h bit 3 to 1, Reg. 78h bit 2 to 0 and pull GNT#3 high, then set Reg. CF9h bit 0 to 1.
- Hardware Deturbo: Set Reg. 5Bh bit 1 to 1, Reg. 65h bit 3 to 1, Reg. 78h bit 2 to 0 and pull GNT#3 high, then press deturbo switch.

2.10.2 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 **06h**

Register 03h **Device ID - high byte**

Bits 7:0 **04h**

Register 04h **Command - low byte**

Bit 7 **Reserved**

Bit 6 **Respond to parity.**

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

Bits 5:4 **Reserved**

Bit 3 **Enable special cycle.**

This bit is always 0 since the PCMC does not issue special cycle.

Bit 2 **Enable bus master.**

This bit is always 1, allowing the PCMC to serve as a PCI bus master.

Bit 1 **Enable response to memory access.**

0: Disables PCI master's accesses to local memory

1: Enables PCI master's accesses to local memory

Bit 0 **Enable response to I/O access.**

This bit is always 0 since the PCMC does not respond to any PCI I/O cycles. The PCMC only responds to CPU initiated I/O cycles.

Register 05h **Command - high byte**

Bits 7:0 **Reserved**

Register 06h **Status - low byte**

Bits 7:0 **Reserved**

Register 07h **Status - high byte**

Bit 7 **Detected parity error.**

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

Bit 6 **Signaled system error.**

This bit is set when the PCMC asserts SERR#. This bit is cleared by writing a 1 to it.

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 when a CPU to PCI transaction is terminated 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 0B~09h **Class Code**

Bits 23:0 **060000h**

Register 0Eh **Header Type**

Bits 7:0 **00h**

Register 50h

Bits 7:6 **DRAM Read CAS Pulse Width**

00 : 4T

01 : 3T

10 : 2T

11 : Reserved

Bit 5 **DRAM Write CAS Pulse Width**

0 : 3T

1 : 2T

Bit 4 **MA Timing Setting**

0: Normal operation. (MD is changed on the CAS# rising edge.)

1: Advance 1T than normal operation.

When using EDO DRAM, this bit must set to 0.

Bit 3 **Reserved**

Bit 2 **Cache Toggle /Linear burst mode selection**

0: Toggle mode

1: Linear burst mode

Bits 1:0 **DRAM type selection**

00: 256k x N

01: 1M x N

10: 4M x N

11: 16M x N

Register 51h

Bit 7 **L2 Cache Exist or not**

0 : Not Exist

1 : Exist

Bit 6 **L2 Cache Enable**

0 : Disable

1 : Enable

Bit 5 **SRAM type (Standard or Burst)**

0 : Standard SRAM

1 : Burst SRAM

Bit 4 **L2 Cache WT/WB Policy**

0 : Write-Through mode

1 : Write-Back mode

Bits 3:1 **L2 Cache Size**

000 : 64KB

001 : 128KB

010 : 256KB

011 : 512KB

100 : 1MB

101 : 2MB

11x : Reserved

Bit 0 **CPU L1 Cache Write-Back Enable**

0 : Disable

1 : Enable

Register 52h

Bits 7:6 **Standard SRAM Cache speed (Read/Write)**

00 : 5-x-x-x Slower

01 : 4-x-x-x Faster

10 : 3-x-x-x Fastest

11 : Reserved

Bits 5:4 **Standard/Burst SRAM x Setting (Burst Read/Write cycle)**

00 : 3T

x1 : 1T

10 : 2T

Bit 3 **Cache Interleave Enable**

0 : Disable

1 : Enable

Bit 2 **Burst SRAM Cache Burst Cycle**

0 : 4-x-x-x

1 : 3-x-x-x

Bit 1 **Cache Sizing Enable**

0: Normal Operation

1: Always Cache hit to enable Cache Sizing for BIOS

Bit 0 **Refresh RAS Active time**

0 : 6T

1 : 5T

Register 53h

Bit 7 **DRAM CAS precharge time**

0 : 2T

1 : 1T

Bit 6 **Shadow RAM Read Enable**

0 : Disable

1 : Enable

When this bit is enabled, the F segment is shadowed by default.

Before shadowing, BIOS should not turn on the bit so that reading F segment is always forwarded to PCI bus.

Bit 5 **Shadow RAM Write Protection Enable**

0 : Disable

1 : Enable

After porting the shadowed segment into DRAM, this bit can be set so that the corresponding shadowed segment is not writable. Under such circumstances, the cycle which intends to write the segment is treated as non-local memory cycle, and is forwarded to PCI bus.

Bit 4 **Shadow RAM Enable for PCI Master Accesses**

0 : Disable

1 : Enable

Bit 3 **F0000h - FFFFFh Shadow RAM Cacheable**

0 : Non-Cacheable

1 : Cacheable

Note that only code is cacheable to L2/L1 when this bit is set.

Bit 2 **RAS to CAS delay time**

0 : 4T

1 : 3T

Bit 1 **RAS precharge time**

0 : 5T

1 : 4T

Bit 0 **Enable host to CTMPB push rate to be X-1-1-1**

0 : Enable

1 : Disable.

When this bit is disabled, the push rate is defined by bit [5:4] of register 52h.

Register 54h **E Segment Setting**

Bit 7 **E0000h - E3FFFh Shadow RAM Enable**

Bit 6 **E4000h - E7FFFh Shadow RAM Enable**

Bit 5 **E8000h - EBFFFh Shadow RAM Enable**

Bit 4 **EC000h - EFFFFh Shadow RAM Enable**

Bit 3 **E0000h - E3FFFh Shadow RAM Cacheable**

Bit 2 **E4000h - E7FFFh Shadow RAM Cacheable**

Bit 1 **E8000h - EBFFFh Shadow RAM Cacheable**

Bit 0 **EC000h - EFFFFh Shadow RAM Cacheable**

Register 55h **D Segment Setting**

Bit 7 **D0000h - D3FFFh Shadow RAM Enable**

Bit 6 **D4000h - D7FFFh Shadow RAM Enable**

Bit 5 **D8000h - DBFFFh Shadow RAM Enable**

Bit 4 **DC000h - DFFFFh Shadow RAM Enable**

Bit 3 **D0000h - D3FFFh Shadow RAM Cacheable**

Bit 2 **D4000h - D7FFFh Shadow RAM Cacheable**

Bit 1 **D8000h - DBFFFh Shadow RAM Cacheable**

Bit 0 **DC000h - DFFFFh Shadow RAM Cacheable**

Register 56h C Segment Setting

Bit 7 **C0000h - C3FFFh Shadow RAM Enable**

Bit 6 **C4000h - C7FFFh Shadow RAM Enable**

Bit 5 **C8000h - CBFFFh Shadow RAM Enable**

Bit 4 **CC000h - CFFFFh Shadow RAM Enable**

Bit 3 **C0000h - C3FFFh Shadow RAM Cacheable**

Bit 2 **C4000h - C7FFFh Shadow RAM Cacheable**

Bit 1 **C8000h - CBFFFh Shadow RAM Cacheable**

Bit 0 **CC000h - CFFFFh Shadow RAM Cacheable**

Register 57h

Bit 7 **Allocation of Non-cacheable Area #1**

0 : Local DRAM

1 : AT Bus. The local DRAM is disabled.

Bit 6 **Non-cacheable Area #1 Enable**

0 : Disable

1 : Enable

Bits 5:3 **Size of Non-Cacheable Area #1 (within 128 MBytes)**

000 : 64KB

001 : 128KB

010 : 256KB

011 : 512KB

100 : 1MB

101 : 2MB

110 : 4MB

111 : 8MB

Bits 2:0 **A26 ~ A24 of Non-Cacheable Area #1 (within 128 MBytes)**

Register 58h

Bits 7~0 **A23 ~ A16 of Non-Cacheable Area #1 (within 128 MBytes)**

Register 59h

Bit 7 **Allocation of Non-cacheable Area #2**

0 : Local DRAM

1 : AT Bus. The local DRAM is disabled.

Bit 6 **Non-cacheable Area #2 Enable**

0 : Disable

1 : Enable

Bits 5:3 **Size of Non-Cacheable Area #2 (within 128 MBytes)**

000 : 64KB

001 : 128KB

010 : 256KB

011 : 512KB

100 : 1MB

101 : 2MB

110 : 4MB

111 : 8MB

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

Register 5Ah

Bits 7:0 **A23 ~ A16 of Non-Cacheable Area #2 (within 128 MBytes)**

Register 5Bh

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 SiS5501 responds the cycle by asserting DEVSEL# in slowest timing. Otherwise, the cycle is subtractively decoded by SiS 5503, 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 **Slow Refresh Enable (1:4)**

0 : Normal Refresh

1 : Slow Refresh

Bit 3 **DRAM Write Push to CAS delay**

0 : 2T

1 : 1T

Bit 2 **De-turbo Hold time**

0 : Hold 4 us

1 : Hold 8 us (Every 12 us)

Bit 1 **De-turbo Switch Enable**

0 : Always turbo, ignore the status of De-turbo Switch

1 : De-turbo Switch Enable

Bit 0 **CAS Driving Current Control Bit 0 (Please refer to Reg. 5Eh Bit 0 for details)**

Register 5Ch

Bit 7 **Latency from ADS# to Monitor Local Memory Status**

0 : 3T

1 : 2T

Depending on the setting of this bit, the PCI master bridge in the SiS5501 may monitor the local memory status from the inside local memory decoder either by the end of T2 or T3. If the CPU initiates a PCI cycle, it is determined to be converted to PCI side from this point. Specifically, BRDY# is always returned to CPU one CPUCLK later if the CTPPB is not full, for post memory write cycles. Thus, this bit also affects the CPU to PCI Post write speed. When it is set to 0, the Post write rate is 5T for each double word. When it is set to 1, the rate is 4T per double word. For a Qword PCI memory write, the post write rate is 7T(bit7=1), or 8T(bit7=0).

Bit 6 **Enable Refresh Cycle when CPU is hold**

0 : Disable

1 : Enable

Bit 5 **Enable Snoop Filter**

0 : Disable

1 : Enable

Bit 4 **CAS# Pulse Width in PCI master write cycle**

0 : 1T

1 : 2T

Bit 3 **Latency from the disarming of "Full" to the assertion of BRDY# for the pending CPU to PCI write cycle**

0 : 1T

1 : 2T

Bit 2 **Selection of KWE# synchronization**

0 : KWE# is synchronized with ACLK (Recommended)

1 : KWE# is synchronized with CPUCLK

Bit 1 **L2 Tag Length**

0 : 8 bits

1 : 7 bits

Bit 0 **Memory Parity Enable/Disable**

0: Enable parity error detection (default value)

1: Disable parity error detection

Register 5Dh **PCI Control Register**

Bits 7:6 **PCI Clock Frequency Selection**

00 : PCICLK=CPUCLK/2

01 : PCICLK=CPUCLK/1.5

10 : Reserved

11 : PCICLK=14MHz

Bit 5 **Maximum Burstable Address Range in PCI master cycles**

0 : 512 Bytes

1 : 1 KBytes

This bit defines the maximum bursting length for each FRAME# asserting.

Bit 4 **Latency from Reading L2/DRAM to the assertion of TRDY# in PCI master read cycles**

0 : 1T

1 : 2T

Bit 3 **Latency from Packing one Qword into PTMPB to the assertion of CAS#(or KWE#)**

0 : 1T

1 : 2T

This latency is reserved for the Post write data propagating onto MD bus, and also for the parity generation so that minimum set up time for MD data to CAS# will not be violated.

Bit 2 **Latency from TRDY# to BRDY# in CPU read/write PCI slave cycles**

0 : 2 CPUCLKs

1 : 3 CPUCLKs

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 5Eh

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**

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**

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**

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**

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**

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**

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 **CAS Driving Current Control Bit 1**

Register 5B bit 0 and 5E bit 0 are used to control CAS driving current.

Register 5B bit 0	Register 5E bit 0	Minimum Current
0	0	8mA (default)
1	0	4mA
0	1	12mA
1	1	8mA

Register 5Fh

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

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 (pin 138)**

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 61h **STPCLK# Assertion Timer**

Bits 7:0

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 62h **STPCLK# De-assertion Timer**

Bits 7:0

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

Register 63h **System Standby Timer**

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 **SMRAM mapping address.**

Bits 7:0 **Correspond to Host address A[27:20].**

This register together with register 65h define SMRAM location. SMRAM location can either be set to a non-shadow, non-cacheable location by selecting E segment as defined in register 65h or be implemented through logical address remap scheme. Logical address remap is done through comparing the upper 11 bits of access address with the address bits defined in register 64h and 65h. If addresses are compared equal and SRAM area selection has been set to either A or B segment, then access is remapped into an A or B segment access. The SMRAM mapping address should be set up by BIOS during the POST process and the SMI service routine is also moved into the SMRAM area during this process. When the system is in the SMM mode or the SMRAM access control bit is enabled, any access to SMRAM area will be redirected as defined by these two registers.

Note: The SMRAM mapping address defines 1MB granularity and the logical address must not set to the first 1MB memory area.

Register 65h

Bits 7:5 **SMRAM area selection**

000 : E0000h-E7FFFh

100 : A0000h-A7FFFh

010 : A0000h-AFFFFh

110 : B0000h-B7FFFh

001 : B0000h-BFFFFh

others : reserved

The SMRAM area is non-cacheable, and non-shadowed.

E0000h-E7FFFh is a physical and logical address space. The other selections can be used to relocate the SMRAM from the pre-defined area (as defined in registers 64h and 65h) during SMM.

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.

Bit 3 **FLUSH# (De-turbo mode), ADSC# selection (pin 13)**

0: ADSC#

1: FLUSH# (De-turbo mode)

Bits 2:0 **Bits 2-0 correspond to Host Address A[30:28].**

Register 66h

Bit 7 **Reserved (must be '0')**

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

Bits 7:0

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

Register 68h

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

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

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**

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**

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**

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**

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

Register 6Fh

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 **Return Bus to CPU after SIOREQ# is Serviced**

0 : Disable

1 : Enable

Bit 6 **SMOUT**

It is reserved for the application circuit.

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 ~ 77h DRAM Boundary

Each register records the accumulated DRAM size including the present and previous banks.

Bits 7:0 DRAM Bank Boundary Address A[28:21]

00h: 0Mbyte

01h: 2Mbyte

02h: 4Mbyte

04h: 8Mbyte

:

Note: Please refer to "2.6 DRAM Controller" for detailed information.

Register 78h

Bits 7:6 EDO BRDY# Timing Selection

00,10: no EDO DRAM

01: BRDY# type 1 timing (6-2-2-2)

11: BRDY# type 2 timing (7-2-2-2)

Bits 5:4 EDO MDLE to 5502 Timing Selection

00,10: no EDO DRAM

01: MDLE type 1 timing (6-2-2-2)

11: MDLE type 2 timing (7-2-2-2)

Bit 3 ADSV#, RAS6# Selection (pin 12)

0: Select ADSV#

1: Select RAS6#

Bit 2 ADSC#/FLUSH#, RAS7# Selection (pin 13)

0: Select ADSC#/FLUSH#

1: Select RAS7#

Bit 1 NA#, RAS4# selection (pin 193)

0: Select NA#

1: Select RAS4#

Bit 0 MA11, RAS5# selection (pin 56)

0: Select MA11

1: Select RAS5#

Note: The function of pin 12, pin 13, pin 56, and pin 193 can be chosen by Register 78 bits 0~3 or hardware trap.

Register 79h DRAM Bank Boundary Address A29

Bits 7:0 Corresponds to A29 of Bank 7~0

Register 7Ah

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 Reserved

Bit 1 Control Register 7Dh

0: Disable register 7Dh.

1: Enable register 7Dh.

When the bit 0 of register 5Ch is set to zero, the parity checking function of 5501 is enabled. When register 5Ch bit 0 and 7Ah bit 1 are set to "1", the parity check of each bank is controlled by register 7Dh.

Bit 0 Reserved

Register 7Bh

Bit 7 AD[31:0] output current selection

0: 50mA/2.2V (default value)

1: 95mA/2.2V

Bit 6 **FRAME#, IRDY#, TRDY#, DEVSEL#, C/BE[3:0]# output current selection**

0: 50mA/2.2V (default value)

1: 95mA/2.2V

Bit 5 **GNT[3:0]#, PAR, SERR# output current selection**

0: 50mA/2.2V (default value)

1: 95mA/2.2V

Bits 4:0 **Reserved**

Register 7Ch

Bit 7 **Set CMPOP Synchronous to CAS#**

0: CMPOP is active after CAS# is active for 1T.

1: CMPOP and CAS# are active at the same time.

Bit 6 **EDO DRAM Write CAS# Pulse Width Control Bit**

0: Disable register 7Ch bit 4.

1: Enable register 7Ch bit 4.

Bit 5 **MDLE Control in EDO Write Access**

0: According to bits 5 and 4 of register 78h.

1: MDLE and CAS# are active at the same time.

Bit 4 **EDO DRAM Write CAS# pulse Width**

0: 1T

1: 2T

In data sheet Rev. 1.0, register 7Ch bit 1 is used to set the CAS# pulse width of EDO DRAM read and write cycle. From 5501 REV. 1C, register 7Ch bit 1 is only used to define the CAS# pulse width in read cycle while bit 4 of register 7Ch is used to define the CAS# pulse width in write cycle.

Bit 3 **Set MDLE Always High**

This bit is only for testing purpose.

0: Disable

1: Enable

Bit 2 **Standard SRAM First Cache Read/Write Cycle Setting**

0: read 3T and write 3T

1: read 3T and write 4T

In the original design, bits 7 and 6 of register 52h are used to define the first cycle time of cache burst read and write cycle when using standard SRAM. When bits 7 and 6 of register 52h are set to "10", the first cycle time is 3T for both read and write cycle. When bit 2 of register 7Ch is set to "1", the first cycle time is redefined to 3T for read cycle and 4T for write cycle. This bit is only valid when the first cycle time is set to 3T defined in bit 7 and 6 oh register 52h.

Bit 1 **Access EDO DRAM CAS# pulse width**

0: 2T

1: 1T

Bit 0 **Access EDO DRAM CAS# pre-charge time**

0: 2T

1: 1T

Note: It is recommended that set the CAS# pulse width and pre-charge time to 1T when the EDO DRAM is used.

Register 7Dh Banks 7~0 Parity Disable Control

This register is valid when bit 1 of register 7A is set.

Bits 7:0 **Bank Parity Disable Control Bit**

0: Enable parity check.

1: Disable parity check.

Register 7Eh

Bit 7 **Setting Bank 7 Standard/EDO type DRAM**

0: Standard DRAM

1: EDO type DRAM

Bit 6 **Setting Bank 6 Standard/EDO type DRAM**

0: Standard DRAM

1: EDO type DRAM

Bit 5 **Setting Bank 5 Standard/EDO type DRAM**

0: Standard DRAM

1: EDO type DRAM

Bit 4 **Setting Bank 4 Standard/EDO type DRAM**

0: Standard DRAM

1: EDO type DRAM

Bit 3 **Setting Bank 3 Standard/EDO type DRAM**

0: Standard DRAM

1: EDO type DRAM

Bit 2 **Setting Bank 2 Standard/EDO type DRAM**

0: Standard DRAM

1: EDO type DRAM

Bit 1 **Setting Bank 1 Standard/EDO type DRAM**

0: Standard DRAM

1: EDO type DRAM

Bit 0 **Setting Bank 0 Standard/EDO type DRAM**

0: Standard DRAM

1: EDO type DRAM

2.11 Pin Assignment and Description

2.11.1 Hardware Trap

5501 will strobe the status of GNT[3:0]# on the rising edge of PWRGD to determine the function of pin 12, 13, 56, and 193. The definition is described below:

Pin No.	Function	condition
12	RAS6#	GNT2# pulled low via 10K ohms resistor
13	RAS7#	GNT3# pulled low via 10K ohms resistor
56	RAS5#	GNT1# pulled low via 10K ohms resistor
193	RAS4#	GNT0# pulled low via 10K ohms resistor
12	ADSV#	GNT2# pulled high via 10K ohms resistor
13	ADSC#/FLUSH#	GNT3# pulled high via 10K ohms resistor
56	MA11	GNT1# pulled high via 10K ohms resistor
193	NA#	GNT0# pulled high via 10K ohms resistor

Due to pin restriction, pin 13 of 5501 is shared by ADSC#, FLUSH#, and RAS7#. The hardware trap can only distinguish ADSC#/FLUSH# and RAS7#. In order to distinguish ADSC# and FLUSH#, bit 3 of register 65h is implemented. The definition of register 65h bit 3 is described below:

Register 65h bit 3	
"0"	ADSC#
"1"	FLUSH#

Beside the hardware method, software method is also provided to define the multi-function pins. The following is the method to use hardware trap or register to define the multi-function pins.

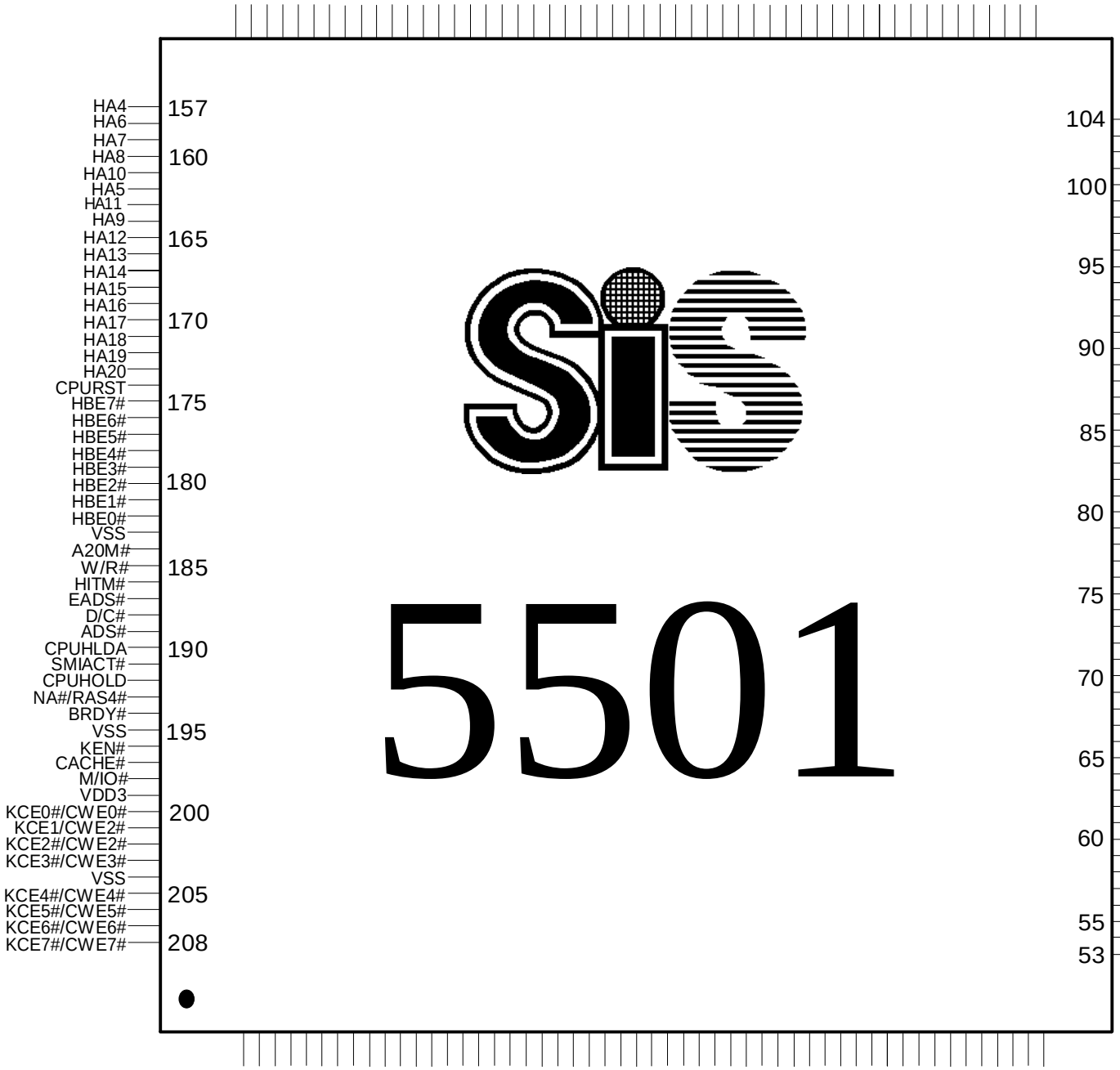
By Hardware trap:

- Set bits[3:0] of register 78h to "0"
- Select function through GNT[3:0]#

By Software:

- Pull GNT[3:0]# to logic "high"
- Select function through bits[3:0] of register 78h

2.11.2 Pin Assignment



2.11.3 Pin Listing (# means active low)

1=CALE	5V/3.3V	48=MA3	5V	95=AD20	5V
2=KA3/KA4Y	5V/3.3V	49=MA4	5V	96=AD21	5V
3=KA4X	5V/3.3V	50=MA5	5V	97=AD22	5V
4=KWY1#	5V/3.3V	51=MA6	5V	98=AD23	5V
5=KWY0#	5V/3.3V	52=MA7	5V	99=AD24	5V
6=VSS		53=MA8	5V	100=AD25	5V
7=KWX1#	5V/3.3V	54=MA9	5V	101=AD26	5V
8=KWX0#	5V/3.3V	55=MA10	5V	102=AD27	5V
9=KREY#/COE1#	5V/3.3V	56=MA11/RAS5#	5V	103=AD28	5V
10=KREX#/COE0#	5V/3.3V	57=HGDW	5V	104=AD29	5V
11=VSS		58=ADLE#	5V	105=AD30	5V
12=ADSV#/RAS6#	5V/3.3V	59=CPPOP	5V	106=AD31	5V
13=ADSC#/FLUSH #/RAS7#	5V/3.3V	60=CPPSH	5V	107=C/BE0#	5V
14=VDD3	5V/3.3V	61=CMPOP	5V	108=C/BE1#	5V
15=TA0	5V	62=CMPSH	5V	109=VSS	
16=TA1	5V	63=MDLE	5V	110=C/BE2#	5V
17=TA2	5V	64=PRDLE	5V	111=C/BE3#	5V
18=TA3	5V	65=ADOE	5V	112=REQ0#	5V
19=TA4	5V	66=PARITY#	5V	113=REQ1#	5V
20=TA5	5V	67=HCR0	5V	114=REQ2#	5V
21=TA6	5V	68=VSS		115=REQ3#	5V
22=TA7	5V	69=HCR1	5V	116=GNT0#	5V
23=ALTWE#	5V	70=HLDA	5V	117=GNT1#	5V
24=ALT	5V	71=PCICLKO	5V	118=GNT2#	5V
25=TAGWE#	5V	72=AD0	5V	119=GNT3#	5V
26=CAS0#	5V	73=AD1	5V	120=STOP#	5V
27=CAS1#	5V	74=AD2	5V	121=DEVSEL#	5V
28=CAS2#	5V	75=AD3	5V	122=TRDY#	5V
29=CAS3#	5V	76=AD4	5V	123=IRDY#	5V
30=CPUCLK	5V	77=AD5	5V	124=FRAME#	5V
31=VSS		78=AD6	5V	125=PLOCK#	5V
32=CAS4#	5V	79=AD7	5V	126=PAR	5V
33=CAS5#	5V	80=VDD	5V	127=SERR#	5V
34=CAS6#	5V	81=AD8	5V	128=VSS	
35=CAS7#	5V	82=AD9	5V	129=PCICLKI	5V
36=ACLK	5V	83=PWRGD	5V	130=SIOGNT#	5V
37=VSS		84=VSS		131=SIOREQ#	5V
38=RAS0#	5V	85=AD10	5V	132=PCIRST#	5V
39=RAS1#	5V	86=AD11	5V	133=SMOUT	5V
40=VDD	5V	87=AD12	5V	134=WAKEUP1	5V
41=RAS2#	5V	88=AD13	5V	135=WAKEUP0	5V
42=RAS3#	5V	89=AD14	5V	136=VDD	5V
43=VSS		90=AD15	5V	137=TURBO	5V
44=RAMW#	5V	91=AD16	5V	138=KBRST#/BREAK#	5V
45=MA0	5V	92=AD17	5V	139=VSS	
46=MA1	5V	93=AD18	5V	140=OSC	5V
47=MA2	5V	94=AD19	5V	141=VDD3	5V/3.3V

142=STPCLK#	5V/3.3V	176=HBE6#	5V/3.3V
143=INIT	5V/3.3V	177=HBE5#	5V/3.3V
144=SMI#	5V/3.3V	178=HBE4#	5V/3.3V
145=HA23	5V/3.3V	179=HBE3#	5V/3.3V
146=HA21	5V/3.3V	180=HBE2#	5V/3.3V
147=HA24	5V/3.3V	181=HBE1#	5V/3.3V
148=HA22	5V/3.3V	182=HBE0#	5V/3.3V
149=HA27	5V/3.3V	183=VSS	
150=HA26	5V/3.3V	184=A20M#	5V/3.3V
151=HA25	5V/3.3V	185=W/R#	5V/3.3V
152=HA28	5V/3.3V	186=HITM#	5V/3.3V
153=HA31	5V/3.3V	187=EADS#	5V/3.3V
154=HA29	5V/3.3V	188=D/C#	5V/3.3V
155=HA30	5V/3.3V	189=ADS#	5V/3.3V
156=HA3	5V/3.3V	190=CPUHLDA	5V/3.3V
157=HA4	5V/3.3V	191=SMIACT#	5V/3.3V
158=HA6	5V/3.3V	192=CPUHOLD	5V/3.3V
159=HA7	5V/3.3V	193=NA#/RAS4#	5V/3.3V
160=HA8	5V/3.3V	194=BRDY#	5V/3.3V
161=HA10	5V/3.3V	195=VSS	
162=HA5	5V/3.3V	196=KEN#	5V/3.3V
163=HA11	5V/3.3V	197=CACHE#	5V/3.3V
164=HA9	5V/3.3V	198=M/IO#	5V/3.3V
165=HA12	5V/3.3V	199=VDD3	5V/3.3V
166=HA13	5V/3.3V	200=KCE0#/CWE0#	5V/3.3V
167=HA14	5V/3.3V	201=KCE1#/CWE1#	5V/3.3V
168=HA15	5V/3.3V	202=KCE2#/CWE2#	5V/3.3V
169=HA16	5V/3.3V	203=KCE3#/CWE3#	5V/3.3V
170=HA17	5V/3.3V	204=VSS	
171=HA18	5V/3.3V	205=KCE4#/CWE4#	5V/3.3V
172=HA19	5V/3.3V	206=KCE5#/CWE5#	5V/3.3V
173=HA20	5V/3.3V	207=KCE6#/CWE6#	5V/3.3V
174=CPURST	5V/3.3V	208=KCE7#/CWE7#	5V/3.3V
175=HBE7#	5V/3.3V		

2.11.4 Pin Description

Host Interface

Pin No.	Symbol	Type	Function
145-173	HA[31:3]	I/O	The CPU Address is driven by the CPU during CPU bus cycles. The 5501 forwards it to either the DRAM or the PCI bus depending on the address range. The address bus is driven by the 5501 during bus master cycles.
175-182	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.
189	ADS#	I	Address Status is driven by the CPU to indicate the start of a CPU bus cycle.
198	M/IO#	I	Memory I/O definition is an input to indicate an I/O cycle when low, or a memory cycle when high.
185	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.
188	D/C#	I	Data/Code is used to indicate whether the current cycle is a data or code access.
194	BRDY#	O	Burst Ready indicates that data presented are valid during a burst cycle.
192	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.
190	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. CPUHLDA indicates that the CPU has given the bus to another bus master.
186	HITM#	I	Hit Modified indicates the snoop cycle hits a modified line in the L1 cache of the CPU.
184	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.

196	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 5501 during cacheable cycles.
197	CACHE#	I	The Cache pin indicates an 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.
187	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.
174	CPURST	O	Reset CPU is an active high output to reset the CPU.
143	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.
144	SMI#	O	System Management Interrupt is used to indicate the occurrence of system management events. It is connected directly to the CPU SMI# input.
191	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).
142	STPCLK#	O	Stop Clock indicates a stop clock request to the CPU.

Cache & DRAM Interface

Pin No.	Symbol	Type	Function
22-15	TA[7:0]	I/O	TAG RAM data bus lines.
24	ALT	I/O	The ALT bit indicates the particular line in the 2nd level cache contains modified data.
3	KA4X	O	Cache address bit 4 for even bank in an interleaved cache configuration..
2	KA3/KA4Y	O	Cache address bit 4 for odd bank, or Cache address bit 3 in non-interleaved mode.
10	KREX#/COE0#	O	Cache Read Enable for even bank of standard SRAM, or Cache Output Enable for burst SRAM.
9	KREY#/COE1#	O	Cache Read Enable for odd bank of standard SRAM, or Cache Output Enable for burst SRAM. When used as COE1#, it is a copy of COE0# for loading consideration.
8,7	KWX0/1#	O	Cache Write Enable for standard SRAM, even bank.
5,4	KWY0/1#	O	Cache Write Enable for standard SRAM, odd bank.
23	ALTWE#	O	The ALTWE# is the write strobe to the ALT RAM. This signal is active low when cache read miss or cache write hit occurs. It is used to update the ALT bit.
25	TAGWE#	O	TAG RAM write enable output.
208-205 203-200	KCE[7:0]# / CWE[7:0]#	O	Cache Enable pins for standard SRAM indicate that the corresponding byte is accessed. Cache Write Enable pins for burst SRAM to allow cache data RAM update on a byte-by-byte basis.
1	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.
42,41 39,38	RAS[3:0]#	O	The RAS[3:0]# are used to latch the row address on the MA bus. Each RAS[3:0]# corresponds to one DRAM row.

35-32 29-26	CAS[7:0]#	O	The CAS[7:0]# are used to latch the column address on the MA bus. Each CAS[7:0]# corresponds to one byte of the eight-byte wide array.
44	RAMW#	O	RAM Write is an active low output signal to enable local DRAM writes.
55-45	MA[10:0]	O	The MA[10:0] provide the row and column address to the DRAM.

PCI Interface

Pin No.	Symbol	Type	Function
111,110 108,107	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 5501 is a PCI bus master and inputs when it is a PCI slave.
106-85 82,81 79-72	AD[31:0]	I/O	PCI Address /Data Bus *In address phase:* 1. When the 5501 is a PCI bus master, AD[31:0] are output signals. 2. When the 5501 is a PCI target, AD[31:0] are input signals. *In data phase:* 1. When the 5501 is a bus master of a memory read/write cycle, AD[31:0] are floating. 2. When the 5501 is a bus master of a configuration or an I/O cycle, AD[31:0] are input signals in a read cycle, and output signals in a write cycle. 3. When the 5501 is a target of a memory read/write cycle, AD[31:0] are floating. 4. When the 5501 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.
124	FRAME#	I/O	FRAME# is an output when the 5501 is a PCI bus master. The 5501 drives FRAME# to indicate the beginning and duration of an access. When the 5501 is a PCI slave, FRAME# is an input signal.

123	IRDY#	I/O	IRDY# is an output when the 5501 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 5501 is a PCI slave, IRDY# is an input.
122	TRDY#	I/O	TRDY# is an output when the 5501 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 5501 is a PCI master, it is an input.
121	DEVSEL#	I/O	The 5501 drives DEVSEL# based on the DRAM address range being accessed by a PCI bus master or if the current configuration cycle is to the 5501. As an input it indicates if any device has responded to current PCI bus cycle initiated by the 5501.
120	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.
126	PAR	O	Parity is an even parity generated across AD[31:0] and C/BE[3:0]#.
127	SERR#	O	System error is an open drain output for reporting errors.
115-112	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.
119-116	GNT[3:0]#	O	PCI Bus Grant indicates to an agent that access to the PCI bus has been granted.

125	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 5501 considers itself a locked resource and remains in the locked state until PLOCK# is sampled negated on a new PCI cycle.
71	PCICLKO	O	The PCICLKO provides the clock for the 5501/5502/5503 and PCI devices of the system.
129	PCICLKI	I	The PCICLKI input provides the fundamental timing and the internal operating frequency for the 5501. It runs at the same frequency and skew of the PCI local bus. It should be generated from the PCICLKO signal through a clock distribution buffer.
132	PCIRST#	O	The PCI Reset forces the PCI devices to a known state.

Data Buffer Control Interface

Pin No.	Symbol	Type	Function
69,67	HCR[1:0]	O	Host Data Bus Controls. These signals are driven by the 5501 and are used to control the 5502 HD[63:0] bus. They are defined as: 00: 5502 floats HD bus 01: 5502 drives FFFFFFFF to HD bus 10: 5502 drives data from AD bus to HD bus 11: 5502 drives data from MD bus to HD bus
58	ADLE#	O	AD Bus Data Latch Enable. This signal has the following functions: 1. Latch HD or MD data into the PCI read buffer (PRMB) 2. Latch AD data into CPU read PCI buffer on the rising edge of PCICLKI. 3. Latch AD data into PCI posted write buffer (PTMPB) on the rising edge of PCICLKI.
65	ADOE	O	AD Bus Output Enable. This signal is used to enable the 5502 to drive PCI AD bus. It is asserted in CPU writes PCI or PCI master reads local memory cycles.
63	MDLE	O	Memory Data Read Latch Enable. This signal latches the data on the MD bus when negated.

60	CPPSH	O	Push CPU to PCI Posted Write Data into the 5502. The data on the HD bus is latched into the 5502 CPU to PCI Posted Write Buffer on CPPSH rising edge. The edge also increases the write pointer to the next available loading entry in the buffer.
59	CPPOP	O	On the rising edge of CPPOP, the read pointer is changed to address the next available reading location.
62	CMPSH	O	When this signal is asserted, the data on the HD bus is written into the CPU to memory posted write buffer (CTMPB) on the rising edge of CPUCLK, and the write pointer is also changed to address the next available location.
61	CMPOP	O	Pop CPU to Memory Posted Write Buffer Data. When this signal is asserted, the read pointer of the CPU to Memory Posted Write Buffer is increased on the rising edge of CPUCLK.
64	PRDLE	O	This signal latches the current output entry in the CPU to PCI Posted Write Buffer into the prelatch in the 5502. The output of the prelatch is driven onto the PCI AD bus. In a PCI master cycle, PRDLE is asserted when PCI master is reading data from the secondary cache, or when PCI master is writing data to the local memory.
57	HGDW	O	High Double Word Indicator. The signal is driven high when: (1) a high DW from the HD bus is written into CPU to PCI Posted Write Buffer, (2) the CPU reads a high DW from PCI bus, (3) PCI master writes a high DW to local memory, (4) PCI master reads a high DW from local memory.
66	PARITY#	I	Parity Bit, from the 5502.

Multi-function Pins

Pin NO.	Symbol	Type	Function
193	NA#/RAS4#	O	This pin that can be used as NA# or RAS4# depends on the BIOS programming or hardware trap selection. Next Address is driven for one clock to the CPU to indicate that the memory system is ready to accept a new bus cycle. Although the data transfer for the current cycle has not yet completed, the CPU may drive a internally pending cycle out to the address bus two clocks after NA# is asserted. The RAS4# are used to latch the row address on the MA bus.
56	MA11/RAS5#	O	This pin that can be used as MA11 or RAS5# depends on the BIOS programming or hardware trap selection. The MA11 provides the row and column address to the DRAM. The RAS5# are used to latch the row address on the MA bus.
12	ADSV#/RAS6#	O	This pin that can be used as ADSV# or RAS6# depends on the BIOS programming or hardware trap selection. Cache Advance is driven to burst SRAM to advance the internal two-bit address counter to the next address of burst sequence. The RAS6# are used to latch the row address on the MA bus.
13	ADSC#/FLUSH# /RAS7#	O	This pin that can be used as ADSC#, FLUSH#, or RAS7# depends on the BIOS programming or hardware trap selection. Cache Address Strobe Control causes the burst SRAM to latch the cache address. FLUSH# is asserted during deturbo mode. It is used to force CPU to writeback all modified lines in the data cache and invalidate CPU internal cache. The RAS7# are used to latch the row address on the MA bus.

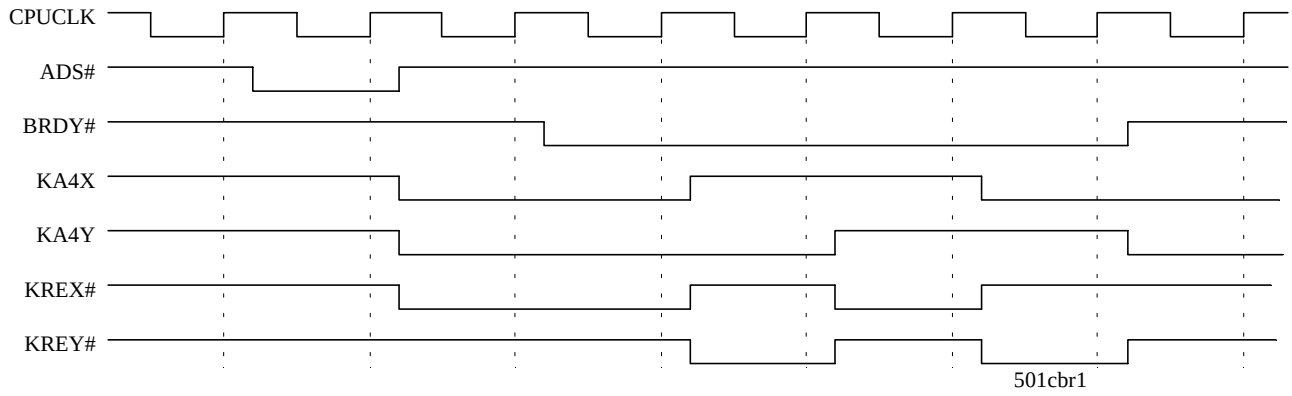
Others

Pin No.	Symbol	Type	Function
70	HLDA	O	Hold Acknowledge.
133	SMOUT	O	System Management Output control pin. It is used to control peripheral's power, clock...etc.
131	SIOREQ#	I	SIO Request from the 5503 to request the PCI bus.
130	SIOGNT#	O	SIO Grant. When asserted, SIOGNT# indicates that the PCI arbiter has granted use of the bus to the 5503.
138	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.
137	TURBO	I	Turbo input pin. The system is in De-turbo mode when this pin is low.
134	WAKEUP1	I	When this input is activated, the 5501 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.
135	WAKEUP0	I	When this input is activated, the 5501 will reload the monitor standby timer. If it is inactive and the monitor standby timer expires, the system will enter monitor standby state. During the monitor standby state, if this input becomes active, the system will wake up from standby state and return back to normal state.
140	OSC	I	OSC is a clock input for the timer and the DMA controller. It is 14.318MHz and is generated by an external oscillator.
36	ACLK	I	Advanced CPU clock should lead the CPUCLK by 3 to 7 ns to provide the clock for the 5501 internal cache control logic.
30	CPUCLK	I	CPU clock input runs at the frequency and skew equal to those of the CPU clock.
83	PWRGD	I	Power Good is a power on reset and push button reset input.
40,80,136	VDD		+5V DC power
14,141 199	VDD3		+3.3V DC power in 3V system +5V DC power in 5V system

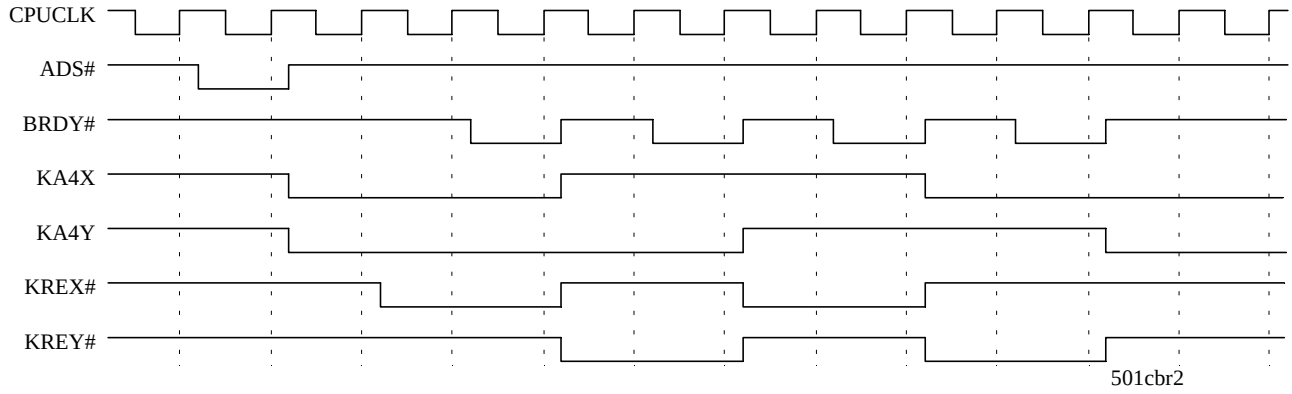
6,11,31,37 43,68,84 109,128 139,183 195,204	VSS		Ground

2.12 Timing Diagram

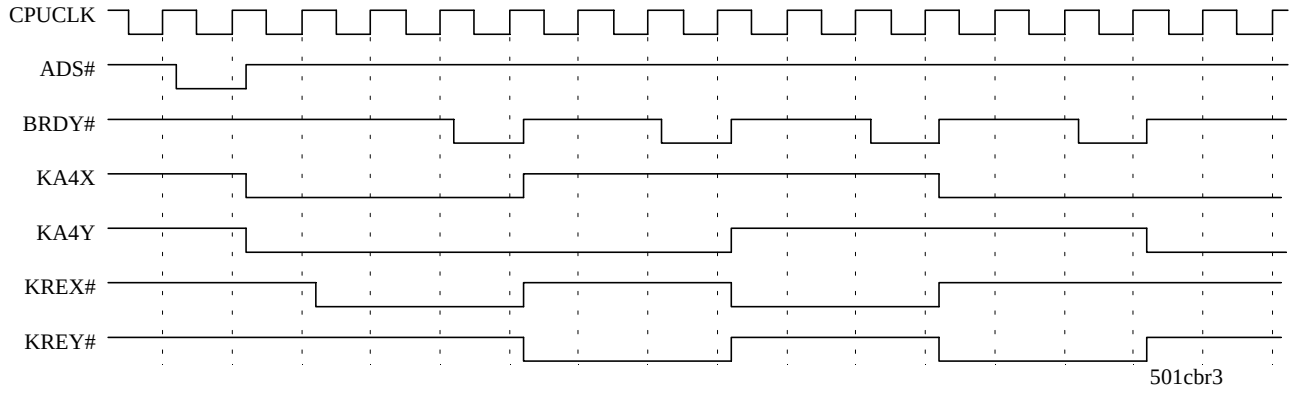
Cache Burst Read Hit Cycle 3-1-1-1



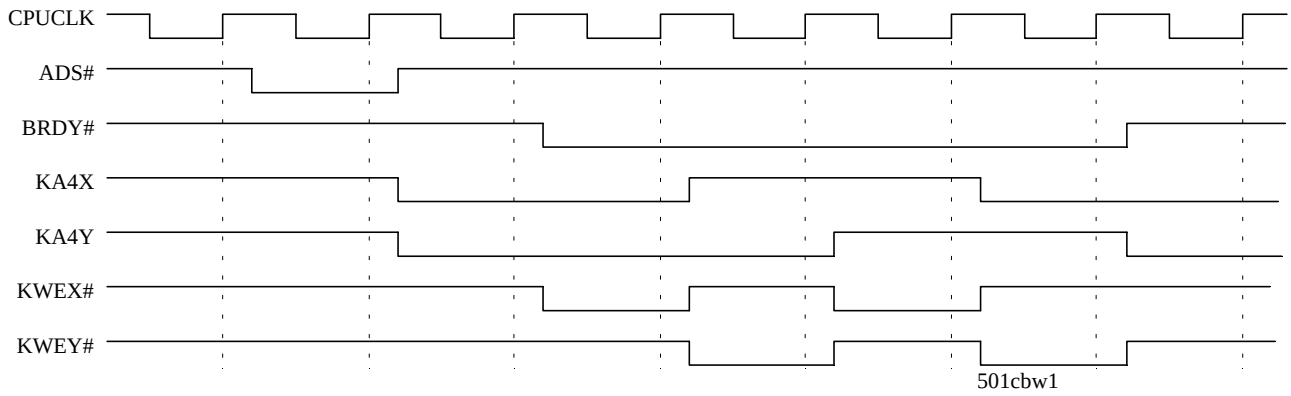
Cache Burst Read Hit Cycle 4-2-2-2



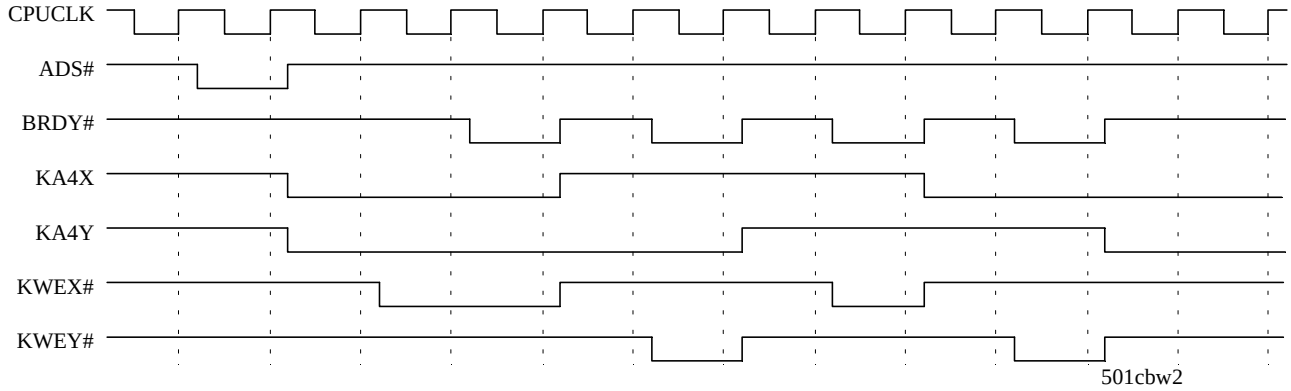
Cache Burst Read Hit Cycle 5-3-3-3



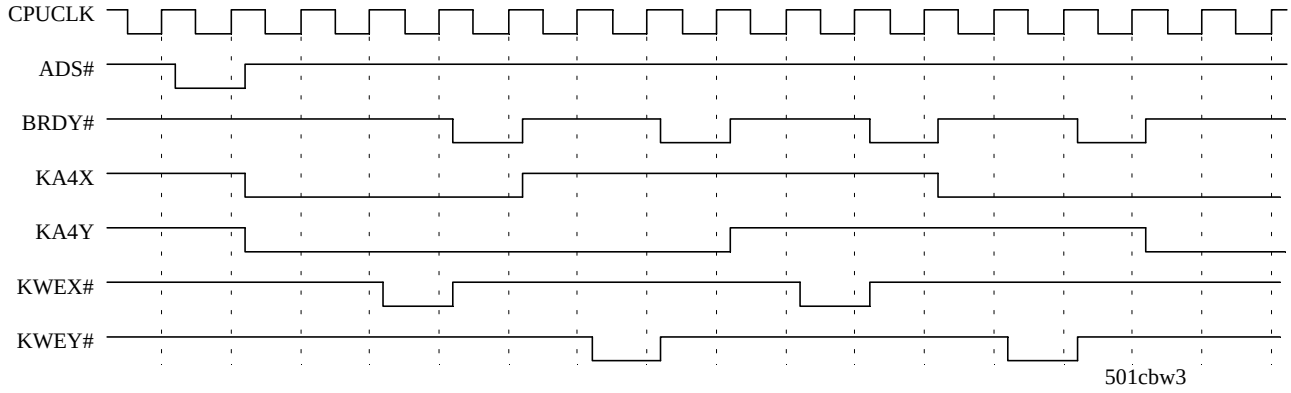
Cache Burst Write Hit Cycle 3-1-1-1



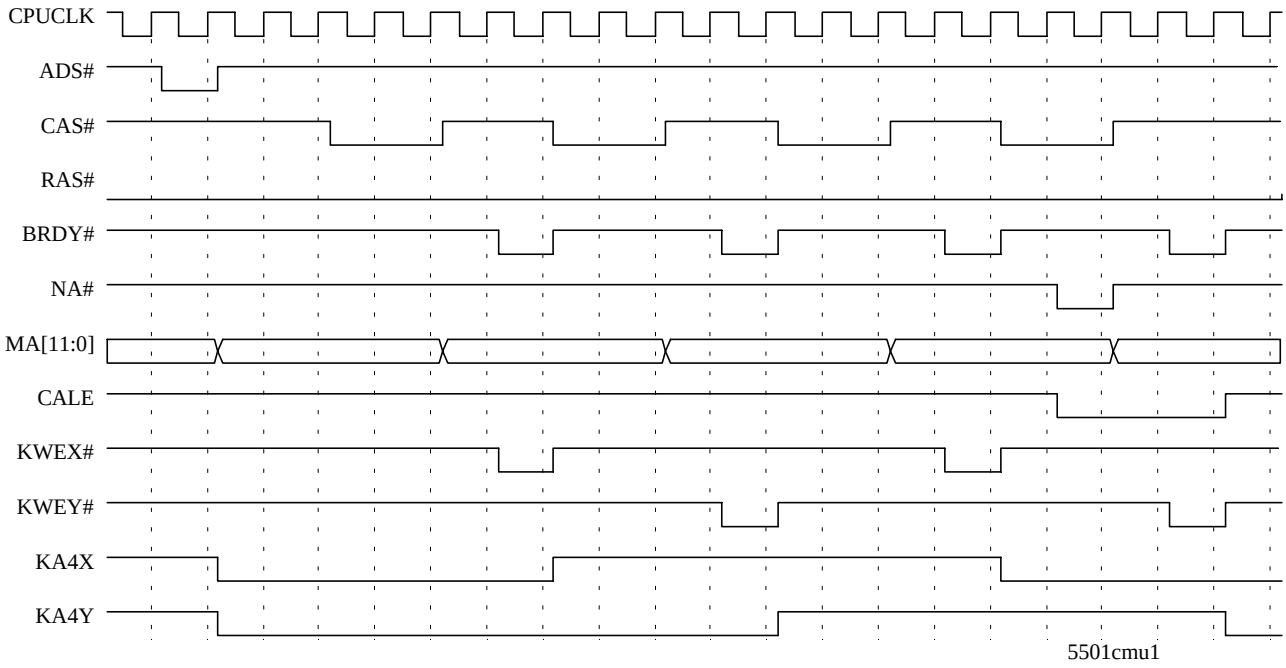
Cache Burst Write Hit Cycle 4-2-2-2



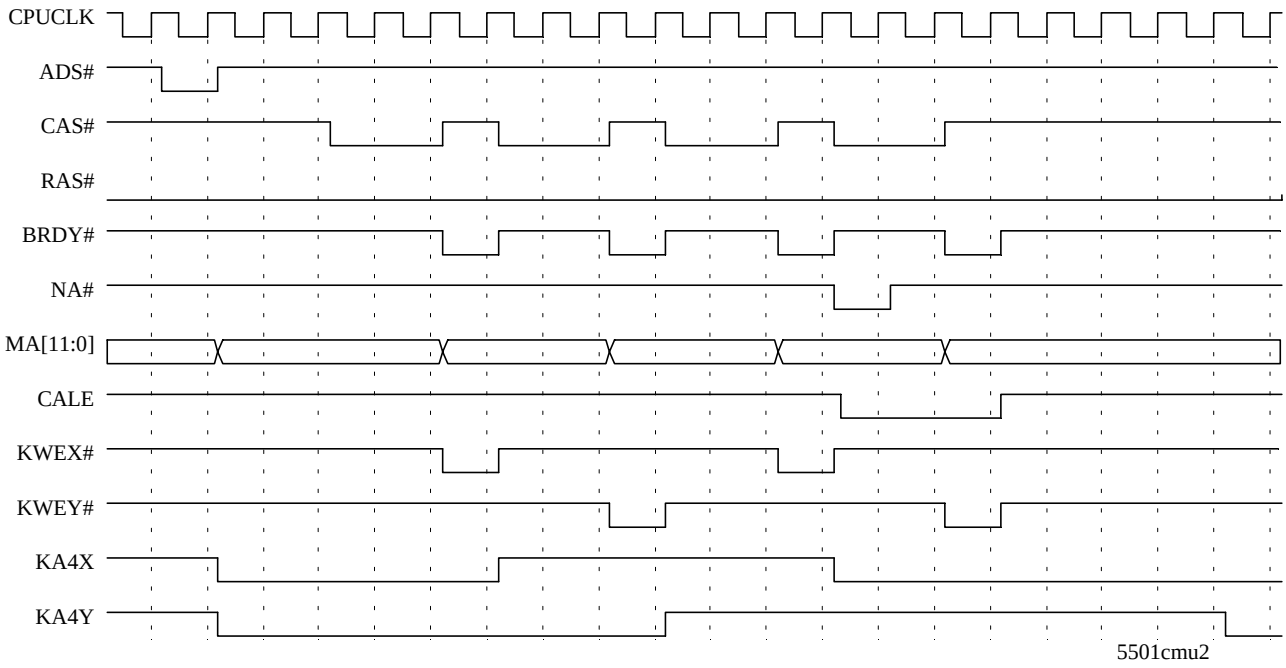
Cache Burst Write Hit Cycle 5-3-3-3



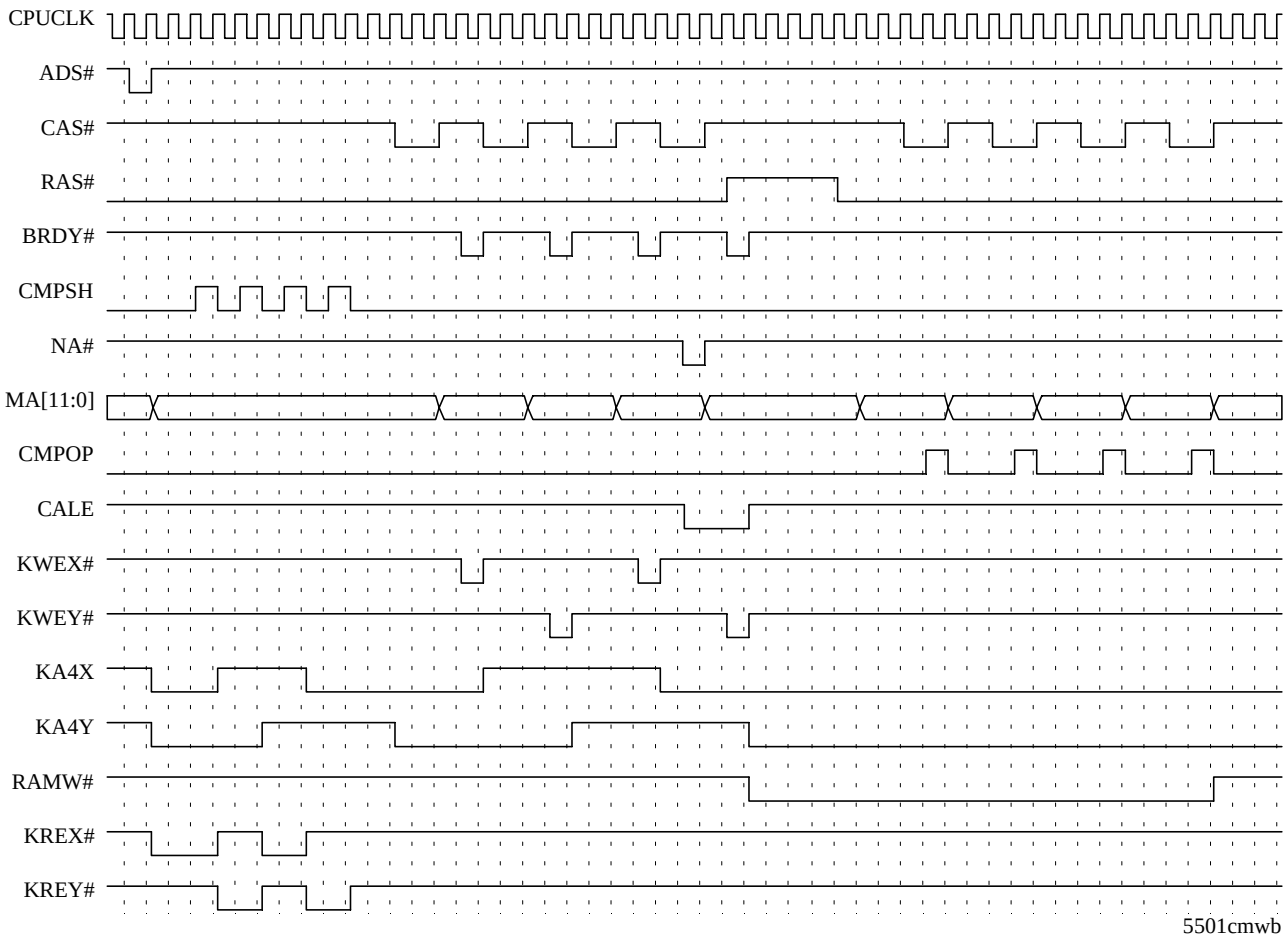
Cache Miss Update Cycle Only (WT or WB , Dirty=0, DRAM:7-4-4-4)



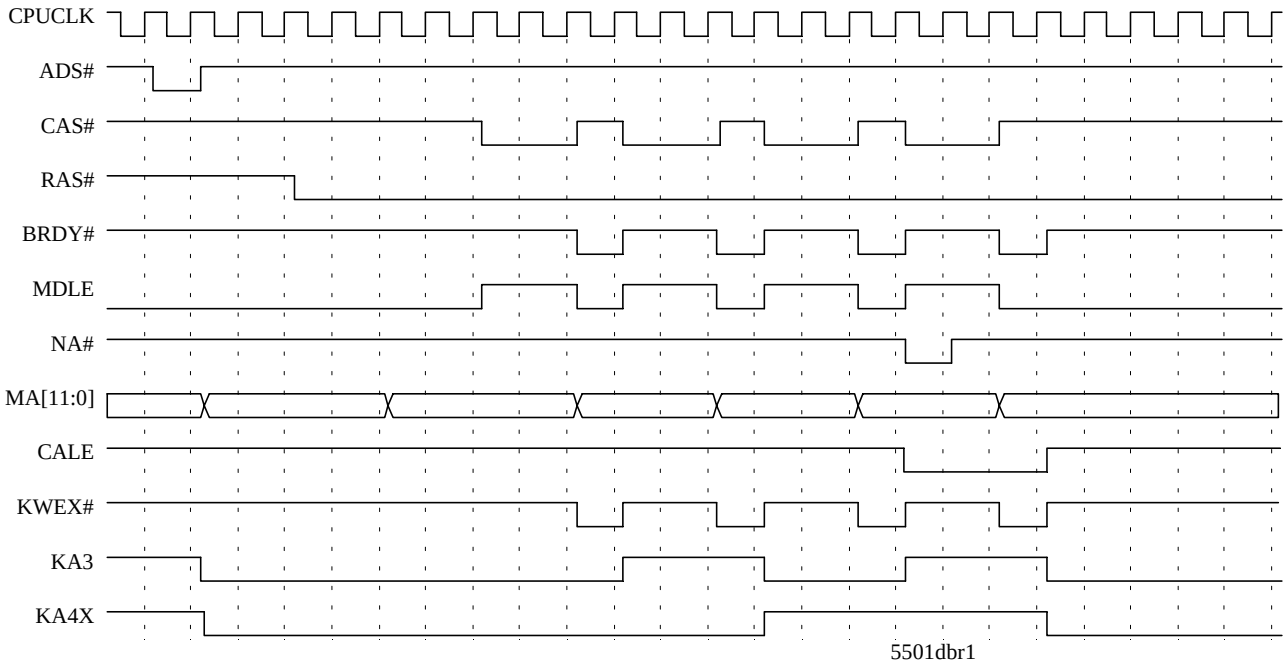
Cache Miss Update Cycle Only (WT or WB , Dirty=0, DRAM:6-3-3-3)



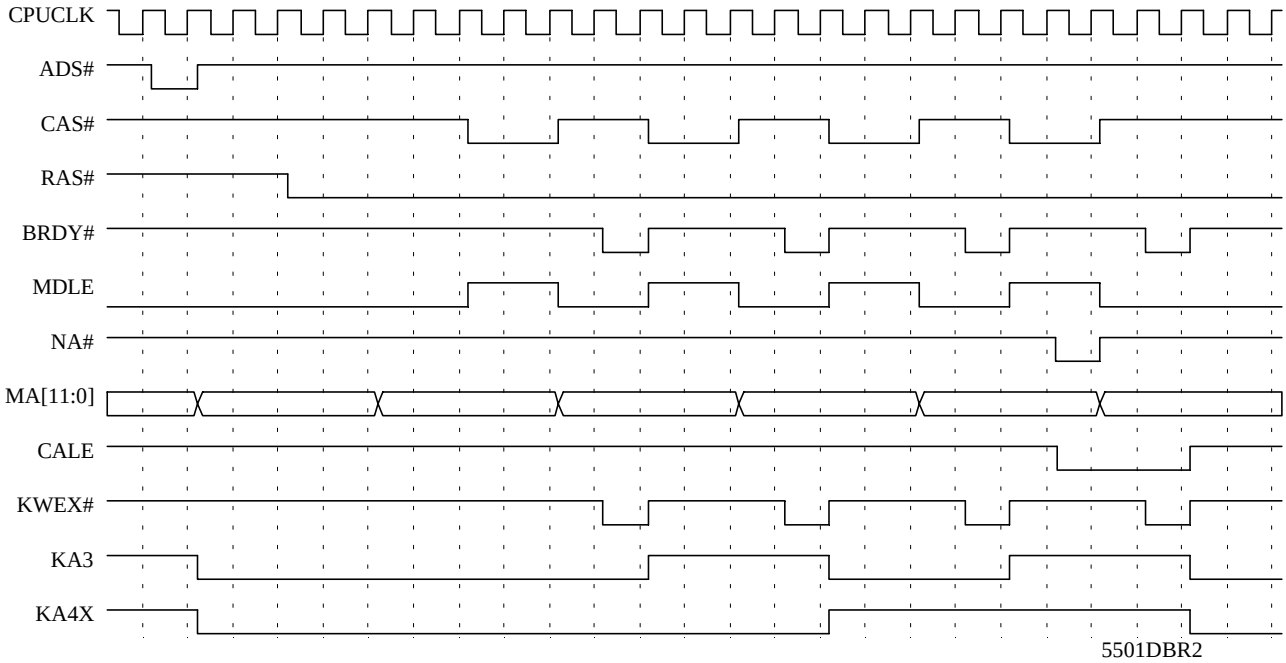
Cache Miss,Concurrent Write Back Cycle (Cache:4-2-2-2 & DRAM:7-4-4-4)



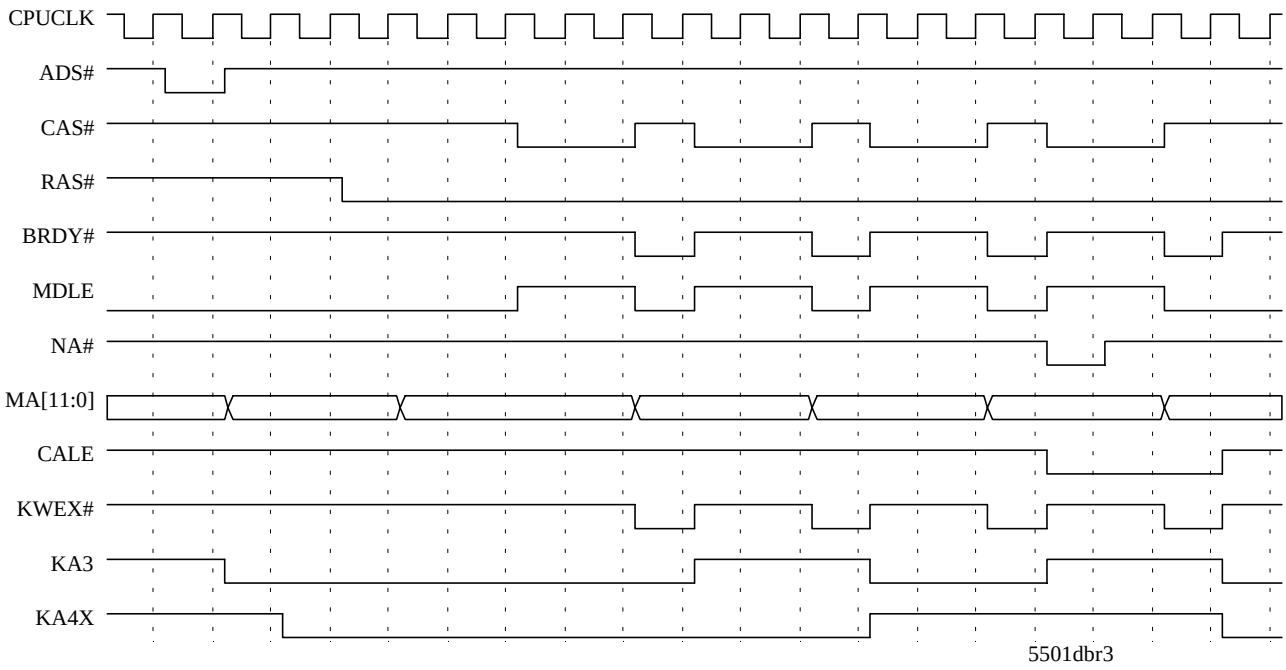
DRAM Burst Read - Row Miss 10-3-3-3 (Cache 3/4-x-x-x)



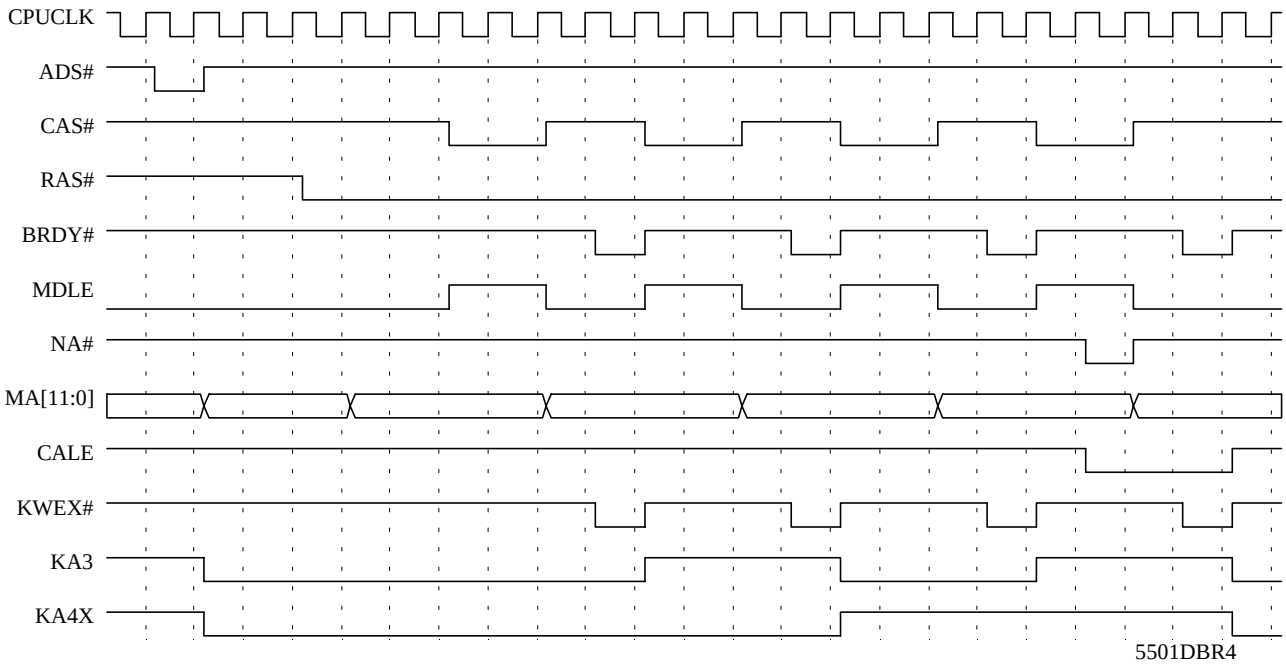
DRAM Burst Read - Row Miss 11-4-4-4 (Cache 3/4-x-x-x)



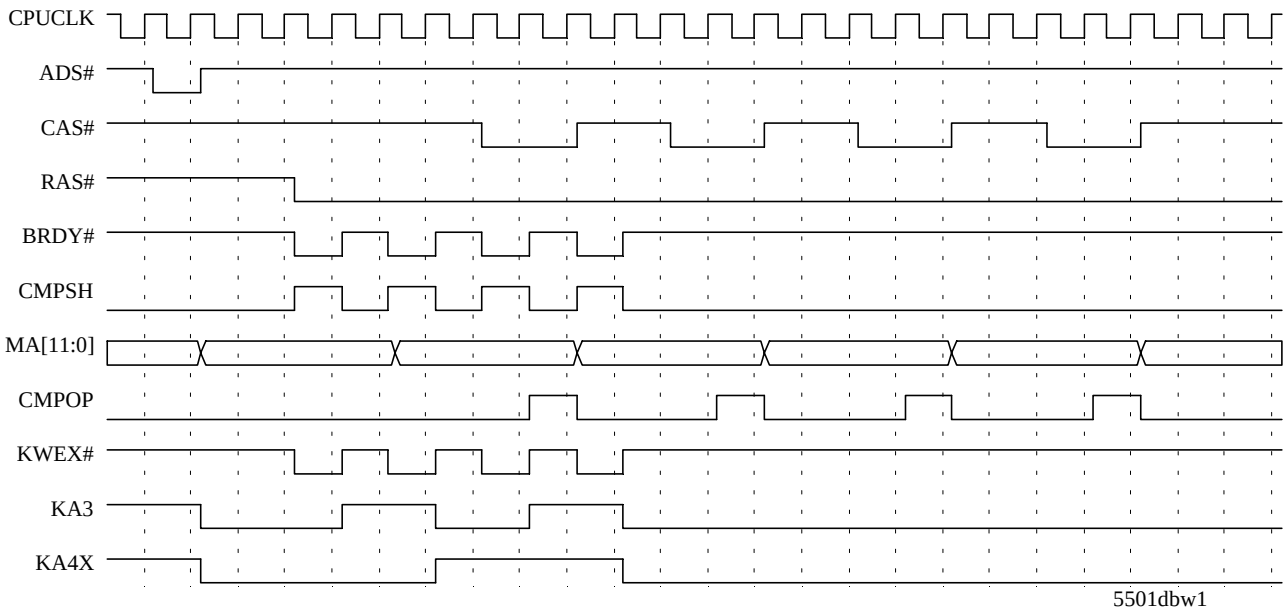
DRAM Burst Read - Row Miss 9-3-3-3 (Cache 3/4-x-x-x)



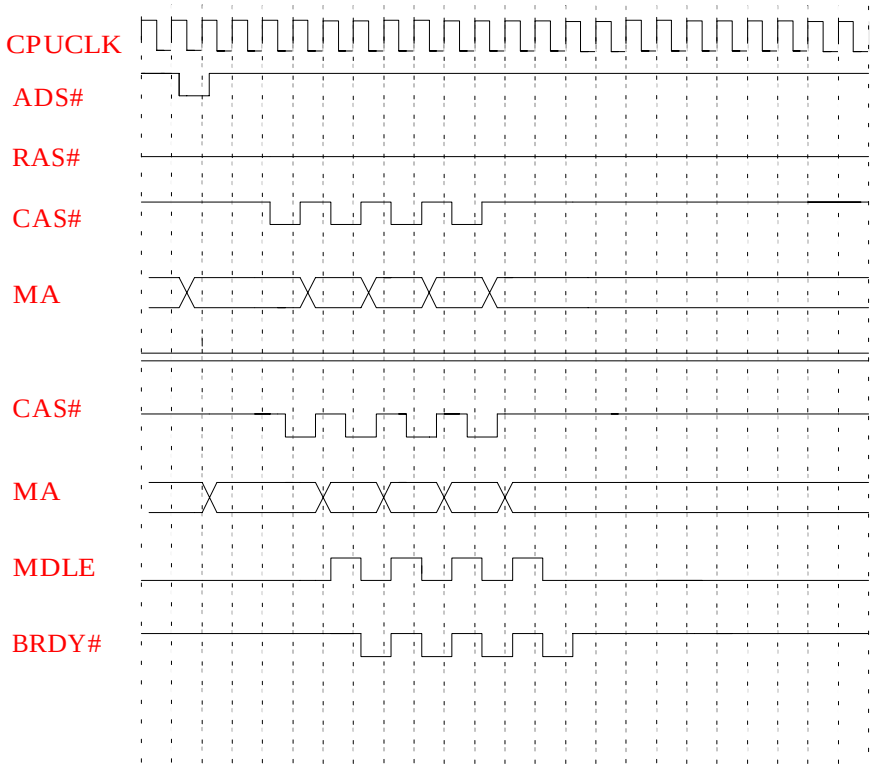
DRAM Burst Read - Row Miss 10-4-4-4 (Cache 3/4-x-x-x)



DRAM Posted Write - 4-2-2-2 , L2 cache Write-Through



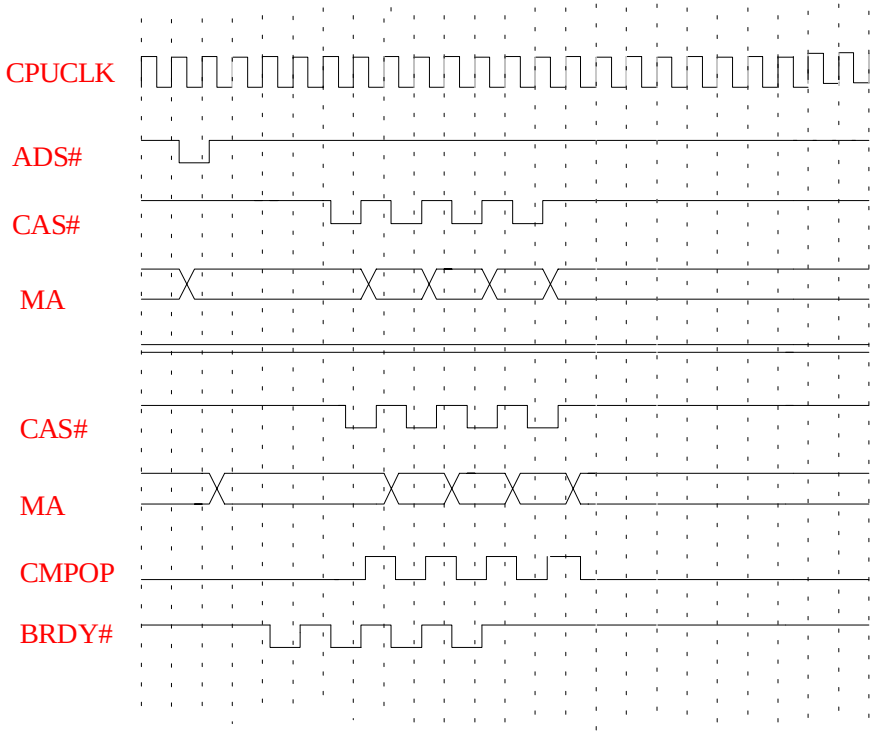
EDO Burst Read (7-2-2-2)



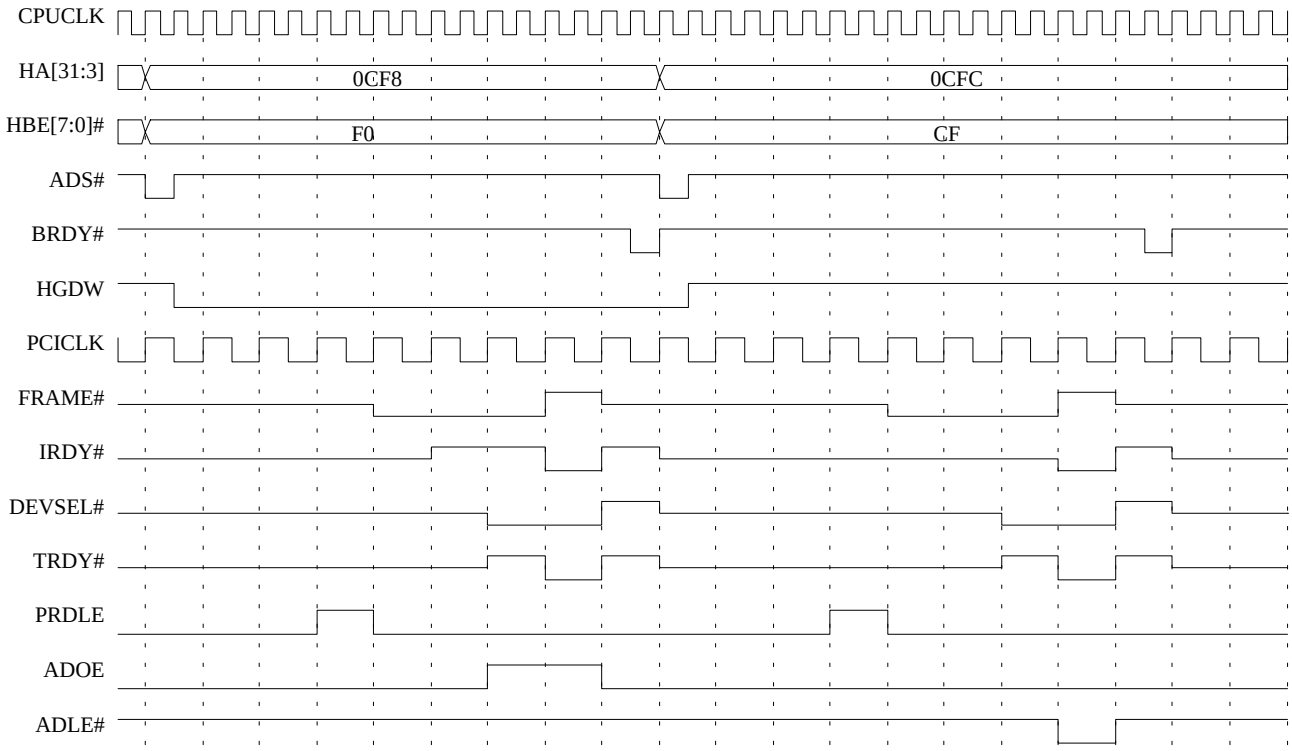
Logic Timing

Physical Timing
Assume CAS delay 10 ns
MA delay 15 ns

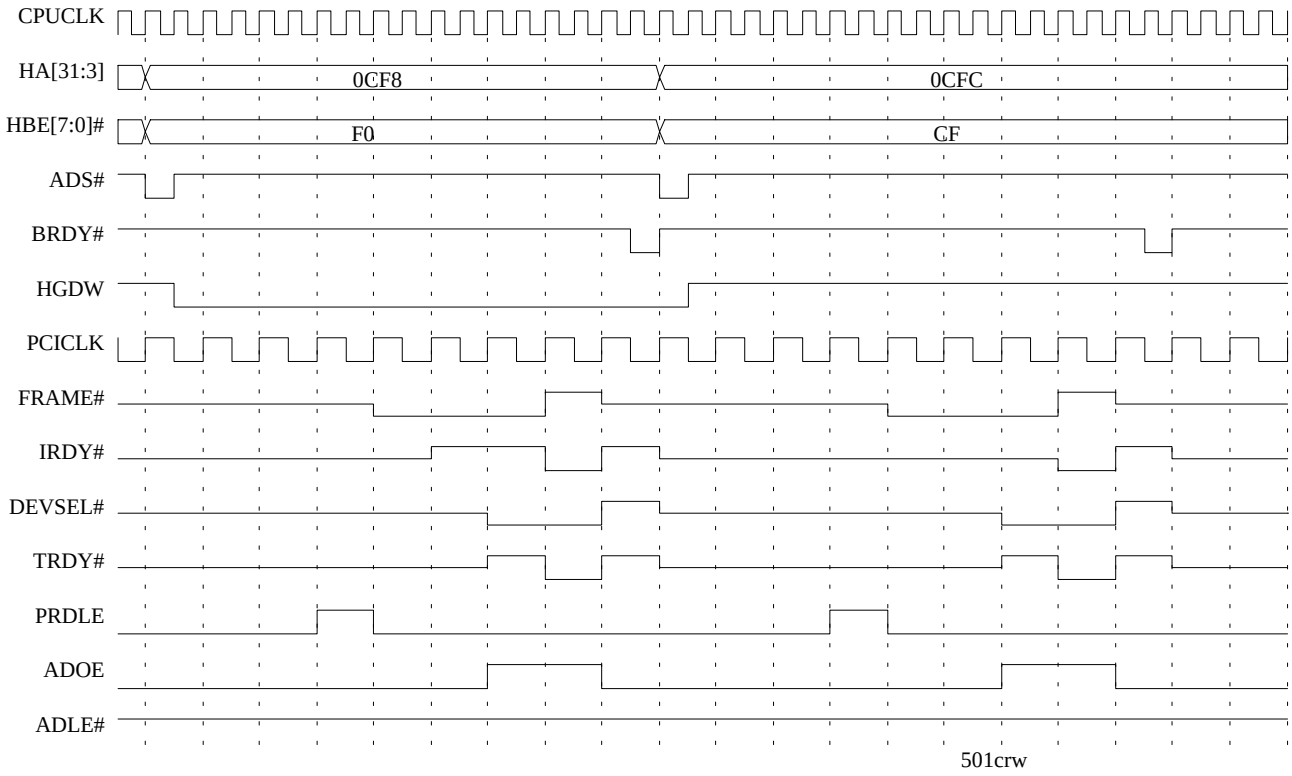
EDO Burst Write (Posted Write 4-2-2-2)



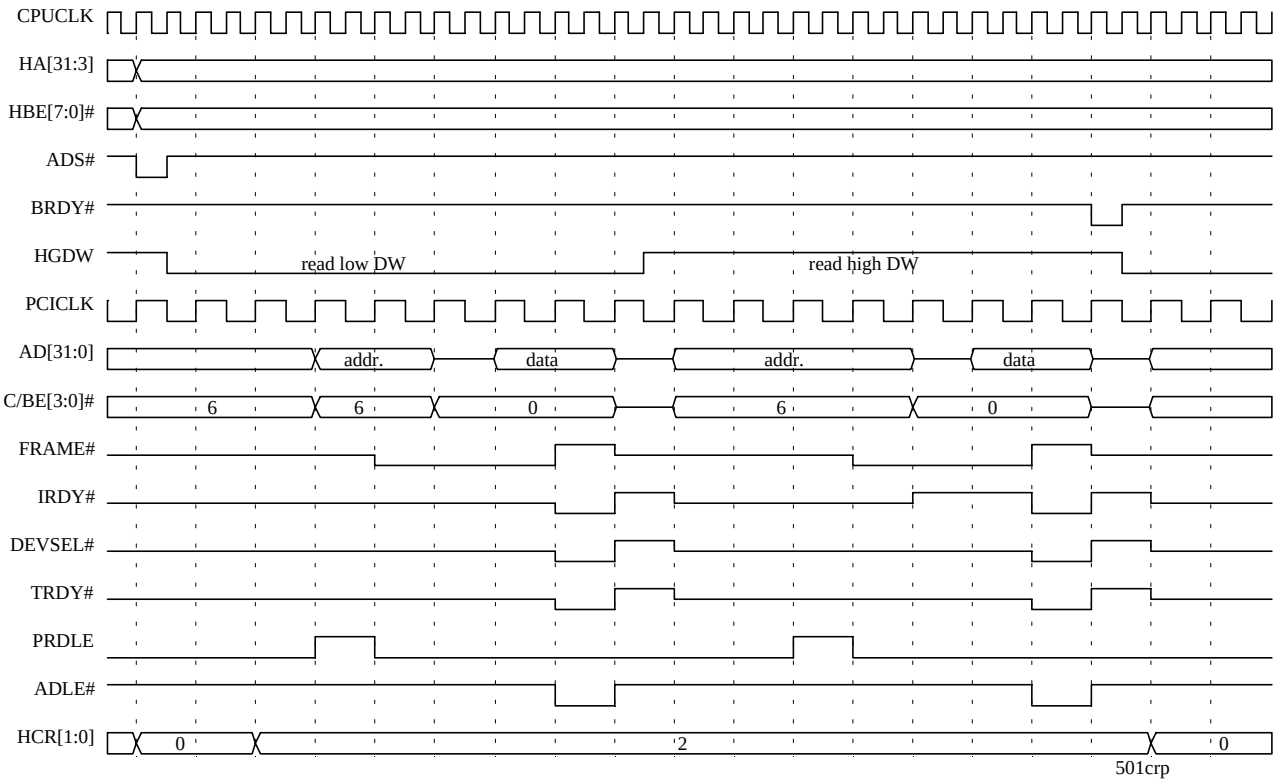
5501 Configuration Register Read Cycle



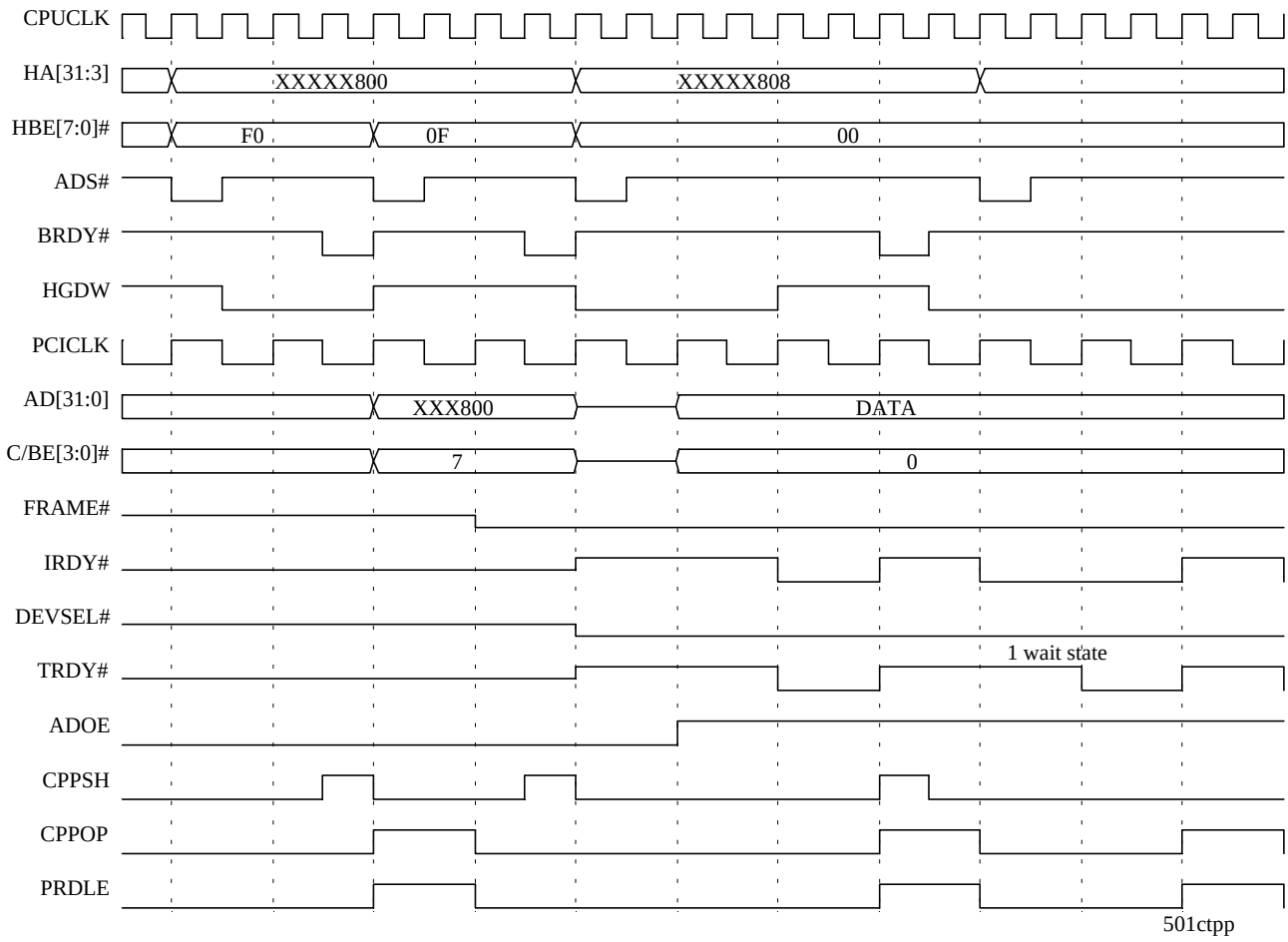
5501 Configuration Register Write Cycle



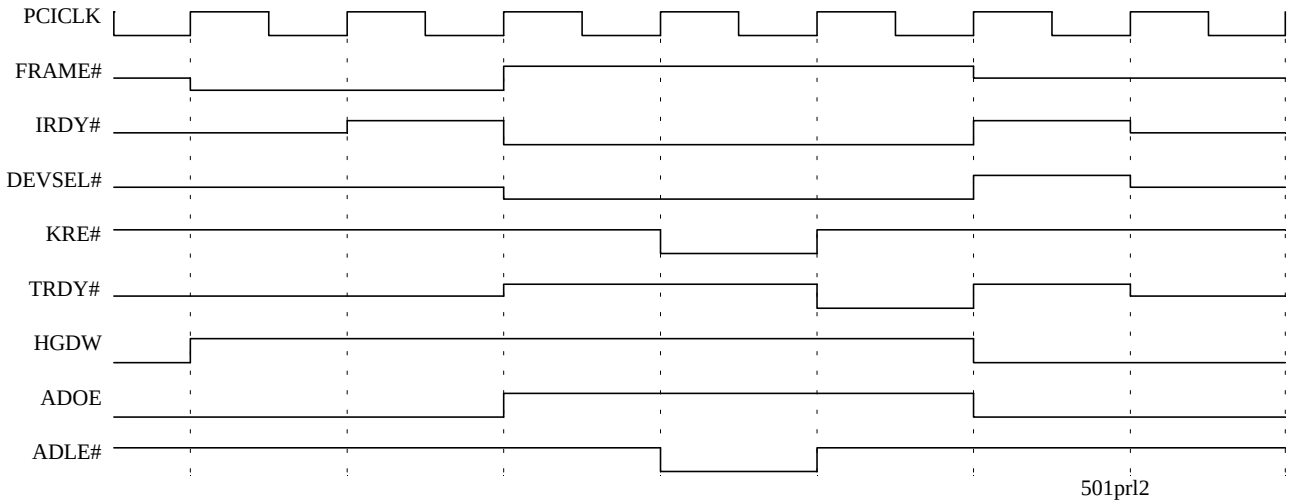
CPU Read PCI Slave



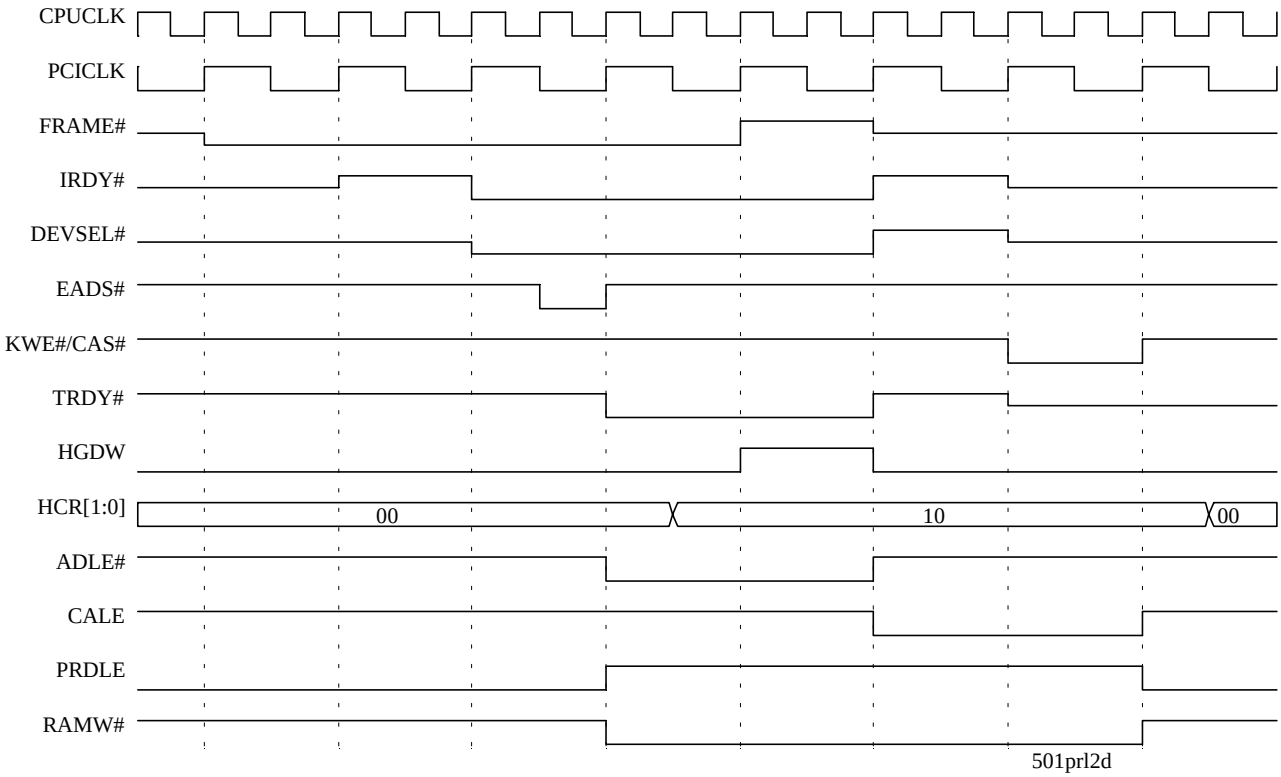
CPU to PCI Psoted Write Cycle



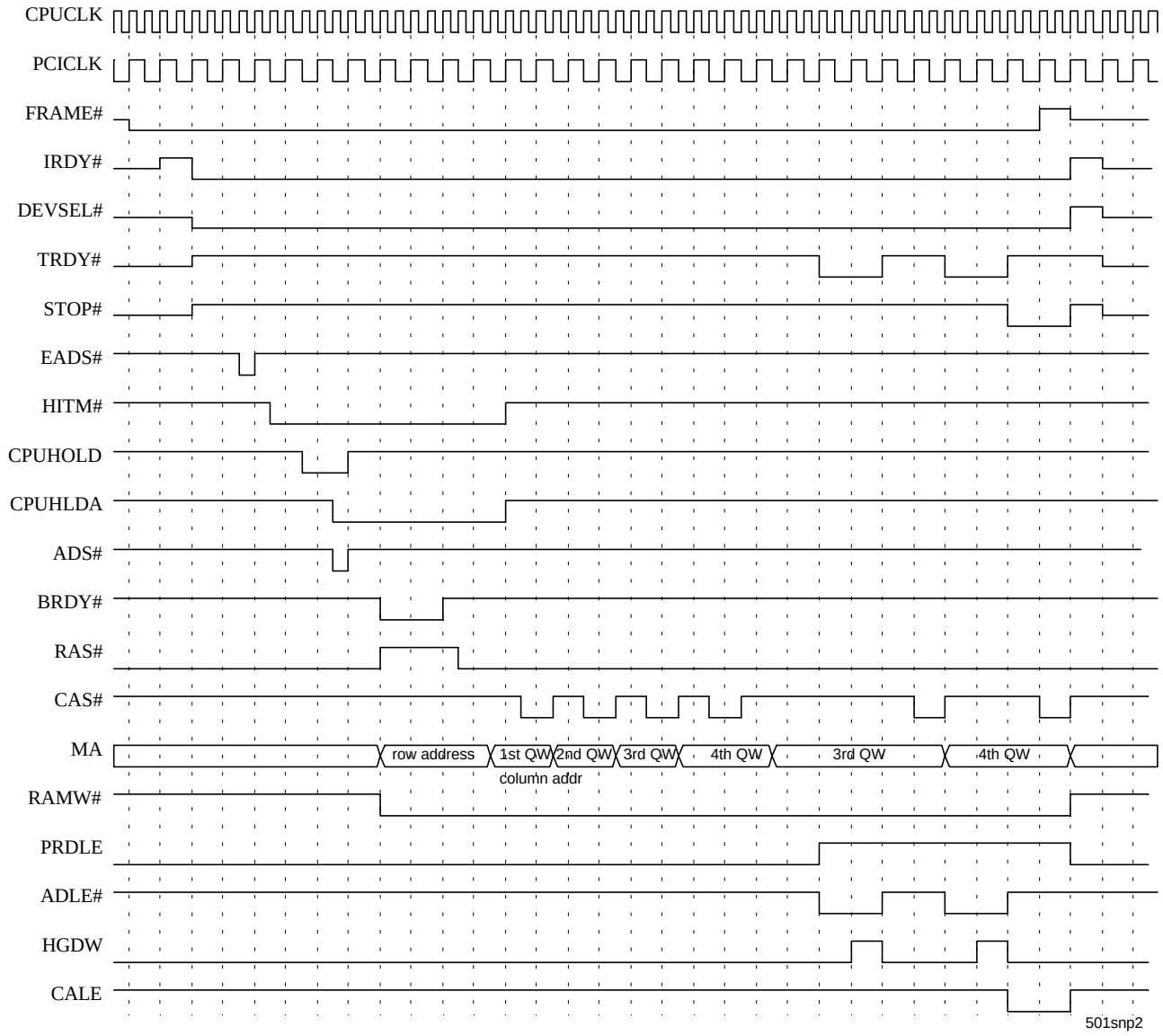
PCI master Reads a High DW from L2,NA=SAL



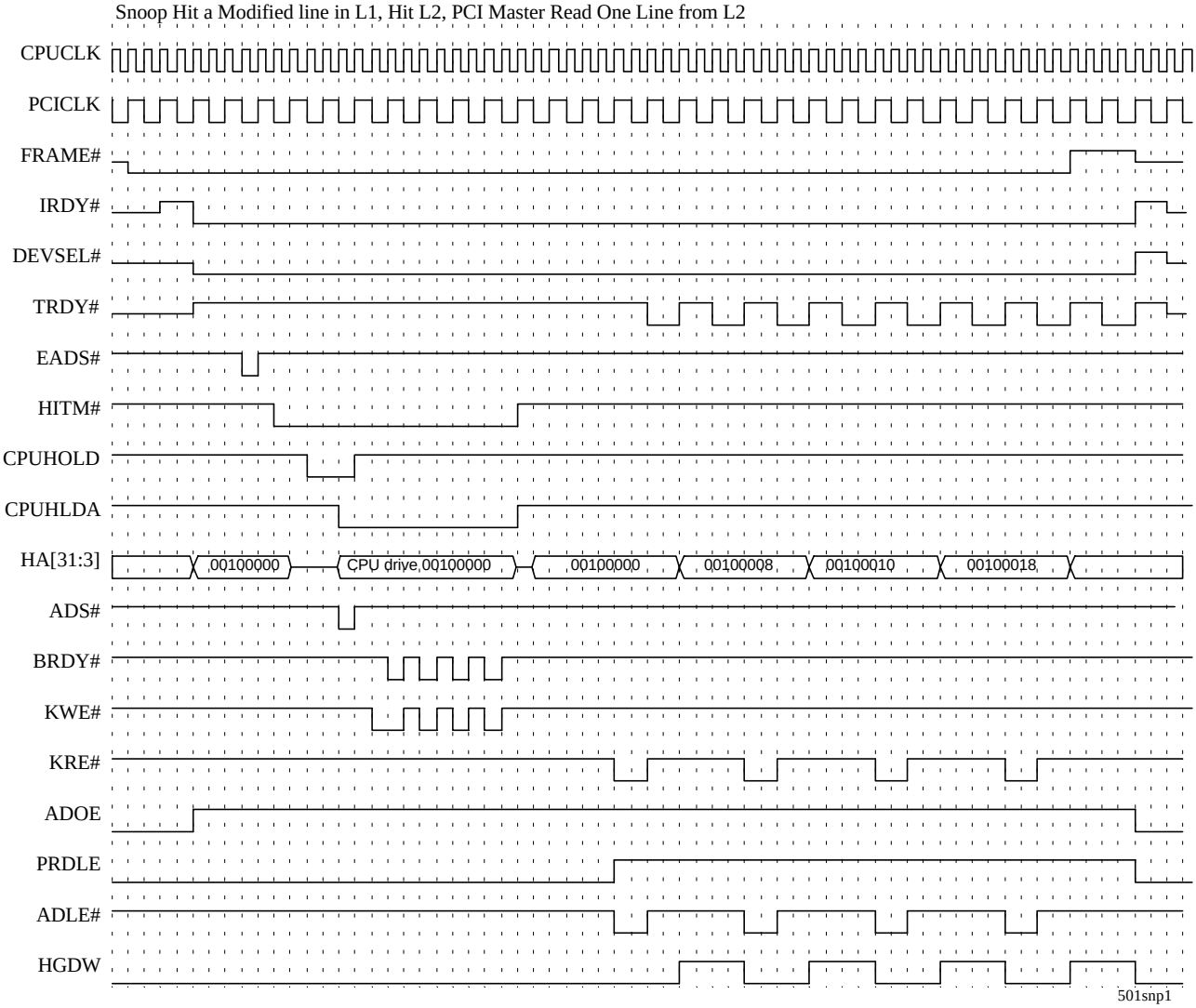
PCI master Writes a QW to L2/DRAM, NA=SAL, Page Hit



Snoop Hit a Modified line in L1, L2 miss
PCI Master Writes the Last Two QW in the 16 Line Boundary, Disconnect



Snoop Hit a Modified line in L1, Hit L2, PCI Master Read One Line from L2



2.13 Electrical Characteristics

2.13.1 Absolute Maximum Ratings

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.

2.13.2 DC Characteristics

TA = 0 - 70 ^{o}C, VSS = 0V , VDD=5V$\pm$5%, VDD3=3.3V$\pm$5%

Symbol	Parameter	Min	Max	Unit	Condition
V_{IL1}	Input low voltage	-0.3	0.8	V	***Note 1***, VDD3=3.3V
V_{IH1}	Input High Voltage	2.2	VDD3+0.3V	V	***Note 1***
V_{IL2}	Input low voltage	-0.3	0.8	V	***Note 2***
V_{IH2}	Input high voltage	2.2	VDD+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	VDD3	V	***Note 5***
I_{OL1}	Output Low Current	4		mA	***Note 6***
I_{OH1}	Output High Current	4		mA	***Note 6***
I_{OL2}	Output Low Current	6		mA	***Note 7***
I_{OH2}	Output High Current	6		mA	***Note 7***
I_{OL3}	Output Low Current	8		mA	***Note 8***
I_{OH3}	Output High Current	8		mA	***Note 8***
I_{OL4}	Output Low Current	16		mA	***Note 9***
I_{OH4}	Output High Current	16		mA	***Note 9***
I_{OL5}	Output Low Current	4		mA	***Note 10***
I_{OH5}	Output High Current	1		mA	***Note 10***, ***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
I_{CC3}	Power Supply Current of VDD3		40	mA	3.3V, 66MHz

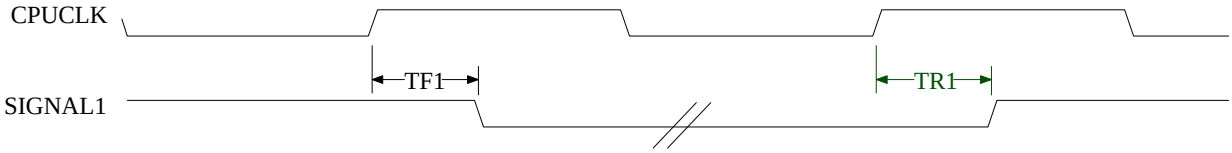
Note:

1. V_{IL1} and V_{IH1} are applicable to HA[31:3], W/R#, HBE[7:0]#, HITM#, D/C#, ADS#, CPUHLDA, SMIACT#, CACHE#, M/IO#
2. V_{IL2} and V_{IH2} are applicable to TA[7:0], ALT, CPUCLK, ACLK, PARITY#, AD[31:0], C/BE[3:0]#, REQ[3:0]#, STOP#, DEVSEL#, TRDY#, IRDY#, FRAME#, LOCK#, PCICLKI, SIOGNT#, SIOREQ#, WAKEUP[1:0], TURBO, KBRST#, OSC
3. V_{T1-},V_{T1+} and V_{H1} are applicable to PWRGD
4. V_{OL1} and V_{OH1} are applicable to TA[7:0], ALTWE#, ALT, TAGWE#, CAS[7:0]#, RAS[3:0]#, RAMW#, MA11/RAS5#, MA[10:0], HGDW, ADLE#, CPPOP, CPPSH, CMPOP, CMPSH, MDLE, PRDLE, ADOE, HCR[1:0], HLDA, PCICLKO, AD[31:0], GNT[3:0]#, STOP#, DEVSEL#, TRDY#, FRAME#, PAR, SERR#, PCIRST#, SMOUT
5. V_{OL2} and V_{OH2} are applicable to CALE, KA4Y, KA4X, KWY[1:0]#, KWX[1:0]#, KREX#, KREY#, ADSC#/FLUSH#/RAS7#, ADSV#/RAS6#, STPCLK#, INIT, SMI#, HA[31:3], CPURST, W/R#, A20M#, EADS#, CPUHOLD, NA#/RAS4#, BRDY#, KEN#, KCE[7:0]#
6. I_{OL1} and I_{OH1} are applicable to TA[7:0], ALTWE#, ALT, TAGWE#, RAMW#, MA11/RAS5#, MA[10:0], HGDW, ADLE#, CPPOP, CPPSH, CMPOP, CMPSH, MDLE, PRDLE, ADOE, HCR[1:0], HLDA, PCICLKO, AD[31:0], C/BE[3:0]#, GNT[3:0]#, PAR, SERR#, PCIRST#, SMOUT, WAKEUP[1:0]
7. I_{OL2} and I_{OH2} are applicable to FRAME#, IRDY#, TRDY#, DEVSEL#, STOP#.
8. I_{OL3} and I_{OH3} are applicable to CAS[7:0]#
9. I_{OL4} and I_{OH4} are applicable to KA4X, KA4Y, KWY[1:0]#, KWX[1:0]#, ADSV#/RAS6#, ADSC#/FLUSH#/RAS7#, RAS[3:0]#
10. I_{OL5} and I_{OH5} are applicable to CALE, KREY#, KREX#, STPCLK#, INIT, SMI#, HA[31:3], W/R#, EADS#, CPUHOLD, NA#/RAS4#, BRDY#, KEN#, KCE[7:0]#, CPURST, A20M#
11. I_{OH5} is 1mA in 3.3 system, when in 5V system, the I_{OH5} is 4mA.
12. The driving current of CAS# and some PCI bus signals are programmable. Please refer to Register Description.

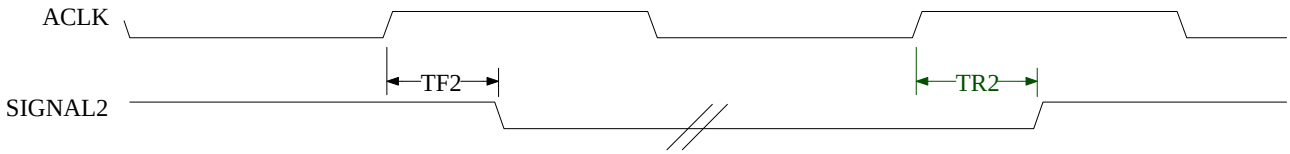
2.13.3 AC Characteristics

Symbol	Parameter	Typ	Max	Unit	CL
T1	BRDY# Active delay from CPUCLK	8	12	ns	35pf
T2	BRDY# Inactive delay from CPUCLK	6	9	ns	35pf
T3	KEN# Active delay from CPUCLK	7	11	ns	35pf
T4	KEN# Inactive delay from CPUCLK	5	8	ns	35pf
T5	NA# Active delay from CPUCLK	8	12	ns	35pf
T6	NA# Inactive delay from CPUCLK	6	9	ns	35pf
T7	EADS# Active delay from CPUCLK	8	12	ns	35pf
T8	EADS# Inactive delay from CPUCLK	6	9	ns	35pf
T9	CPUHOLD Active delay from CPUCLK	8	12	ns	35pf
T10	CPUHOLD Inactive delay from CPUCLK	6	9	ns	35pf
T11	CPURST Inactive delay from CPUCLK	7	11	ns	35pf
T12	CPURST High Pulse Width	25		cpuclk	35pf
T13	KREX#,KREY# Active delay from ACLK	9	13	ns	100pf
T14	KREX#,KREY# Inactive delay from ACLK	6	9	ns	100pf
T15	KWX[0:1]#,KWY[0:1]# Active delay from ACLK	8	12	ns	100pf
T16	KWX[0:1]#,KWY[0:1]# Inactive delay from ACLK	6	9	ns	100pf
T17	KWX[0:1]#,KWY[0:1]# Active delay from CPUCLK	8	12	ns	100pf
T18	KWX[0:1]#,KWY[0:1]# Inactive delay from CPUCLK	6	9	ns	100pf
T19	KCE[7:0]# Active delay from ADS# falling edge	7	12	ns	35pf
T20	KCE[7:0]# Inactive delay from CPUCLK	7	12	ns	35pf
T21	MDLE High Active delay from CPUCLK	5	8	ns	35pf
T22	MDLE High Inactive delay from CPUCLK	7	11	ns	35pf
T23	KA4X,KA4Y Low Valid delay from ACLK	7	11	ns	100pf
T24	KA4X,KA4Y High Valid delay from ACLK	6	9	ns	100pf
T25	KA4X,KA4Y Low Valid delay from CPUCLK In Update Cycle & Write cycle	7	11	ns	100pf
T26	KA4X,KA4Y High Valid delay from CPUCLK In Update Cycle & Write cycle	6	9	ns	100pf
T27	Tag Output Valid delay from CPUCLK In Update Cycle	14	23	ns	35pf
T28	RAS[7:0]# Active delay from CPUCLK	12	18	ns	250pf
T29	RAS[7:0]# Inactive delay from CPUCLK	9	14	ns	250pf
T30	CAS[7:0]# Active delay from CPUCLK	11	16	ns	120pf
T31	CAS[7:0]# Inactive delay from CPUCLK	8	12	ns	120pf
T32	MA[11:0] Low Valid delay from CPUCLK	12	18	ns	35pf
T33	MA[11:0] High Valid delay from CPUCLK	11	16	ns	35pf
T34	MA[11:0] Propagation delay from A[27:3]	8	12	ns	35pf
T35	ALT Output Valid delay from CPUCLK	10	15	ns	35pf
T36	ALTWE#,TAGWE# Active delay from CPUCLK	9	14	ns	35pf
T37	ALTWE#,TAGWE# Inactive delay from CPUCLK	9	14	ns	35pf
T38	A20M# Active delay from CPUCLK	9	14	ns	35pf
T39	A20M# Inactive delay from CPUCLK	8	12	ns	35pf
T40	AD[31:0],C/BE[3:0]# Output valid delay from PCICLKI	10	15	ns	50pf
T41	PRDLE Active delay from PCICLKI	9	14	ns	35pf
T42	DEVSEL#,FRAME#,IRDY#,STOP#,TRDY# Active delay from PCICLKI	10	15	ns	50pf
T43	DEVSEL#,FRAME#,IRDY#,STOP#,TRDY# Inactive delay from PCICLKI	9	14	ns	50pf

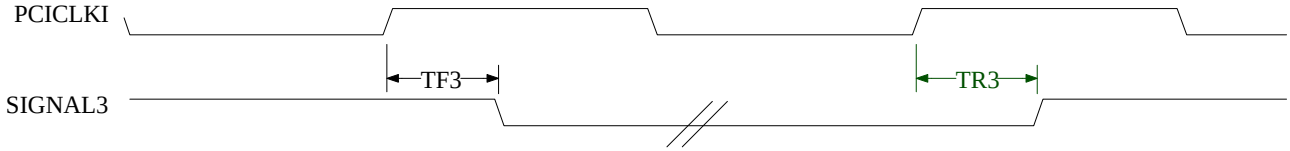
T44	GNT[3:0]#,PAR,SERR#,SIOGNT#,STPCLK# Active delay from PCICLKI	10	15	ns	50pf
T45	GNT[3:0]#,PAR,SERR#,SIOGNT#,STPCLK# Inactive delay from PCICLKI	10	15	ns	50pf
T46	HA[31:3] Drive Output Valid delay from PCICLKI	12	18	ns	50pf
T47	HCR[1:0],HGDW Active delay from CPUCLK	7	11	ns	35pf
T48	HLDA Active delay from CPUCLK	8	12	ns	35pf
T49	HLDA Inactive delay from CPUCLK	7	11	ns	35pf
T50	INIT# Active delay from CPUCLK	7	11	ns	35pf
T51	INIT# Inactive delay from CPUCLK	6	9	ns	35pf
T52	MDLE Active delay from CPUCLK	7	11	ns	35pf
T53	MDLE Inactive delay from CPUCLK	6	9	ns	35pf
T54	PCICLKO,PCIRST Active delay from CPUCLK	8	12	ns	50pf
T55	RAMW# Active delay from CPUCLK	11	16	ns	35pf
T56	RAMW# Inactive delay from CPUCLK	8	12	ns	35pf
T57	SMOUT Active delay from CPUCLK	10	15	ns	50pf
T58	ADSC# Active delay from CPUCLK	7	11	ns	90pf
T59	ADSC# Inactive delay from CPUCLK	6	9	ns	90pf
T60	ADSV# Active delay from CPUCLK	7	11	ns	150pf
T61	ADSV# Inactive delay from CPUCLK	6	9	ns	150pf
T62	CPPSH Active delay from CPUCLK	4	6	ns	35pf
T63	CPPOP Active delay from PCICLK	8	12	ns	35pf
T64	CPPSH Inactive delay from CPUCLK	7	11	ns	35pf
T65	CPPOP Inactive delay from PCICLK	10	15	ns	35pf
T66	ADOE Active delay from PCICLK	6	9	ns	35pf
T67	ADOE Inactive delay from PCICLK	6	9	ns	35pf
T68	ADLE# Active delay from PCICLK	6	9	ns	35pf
T69	ADLE# Inactive delay from PCICLK	6	9	ns	35pf
T70	PCICLKO high time (Divided by 2)	15.2		ns	50pf
T71	PCICLKO low time (Divided by 2)	12.6		ns	50pf
T72	PCICLKO high time (Divided by 1.5)	12.5		ns	50pf
T73	PCICLKO low time (Divided by 1.5)	15.8		ns	50pf
T74	PCICLKO rise time (Divided by 2)	1.16		ns	50pf
T75	PCICLKO fall time (Divided by 2)	0.66		ns	50pf
T76	PCICLKO rise time (Divided by 1.5)	1.06		ns	50pf
T77	PCICLKO fall time (Divided by 1.5)	0.9		ns	50pf
T78	HCR[1:0] fall time to CPUCLK rising	4.5		ns	35pf
T79	HCR[1:0] rise time to CPUCLK rising	3.7		ns	35pf
T80	CALE# Active delay from CPUCLK	8	12	ns	35pf
T81	CALE# Inactive delay from CPUCLK	6	9	ns	35pf
T82	SMI# rise time to CPUCLK rising	7.8	10	ns	35pf
T83	SMI# fall time to CPUCLK rising	7.8	10	ns	35pf



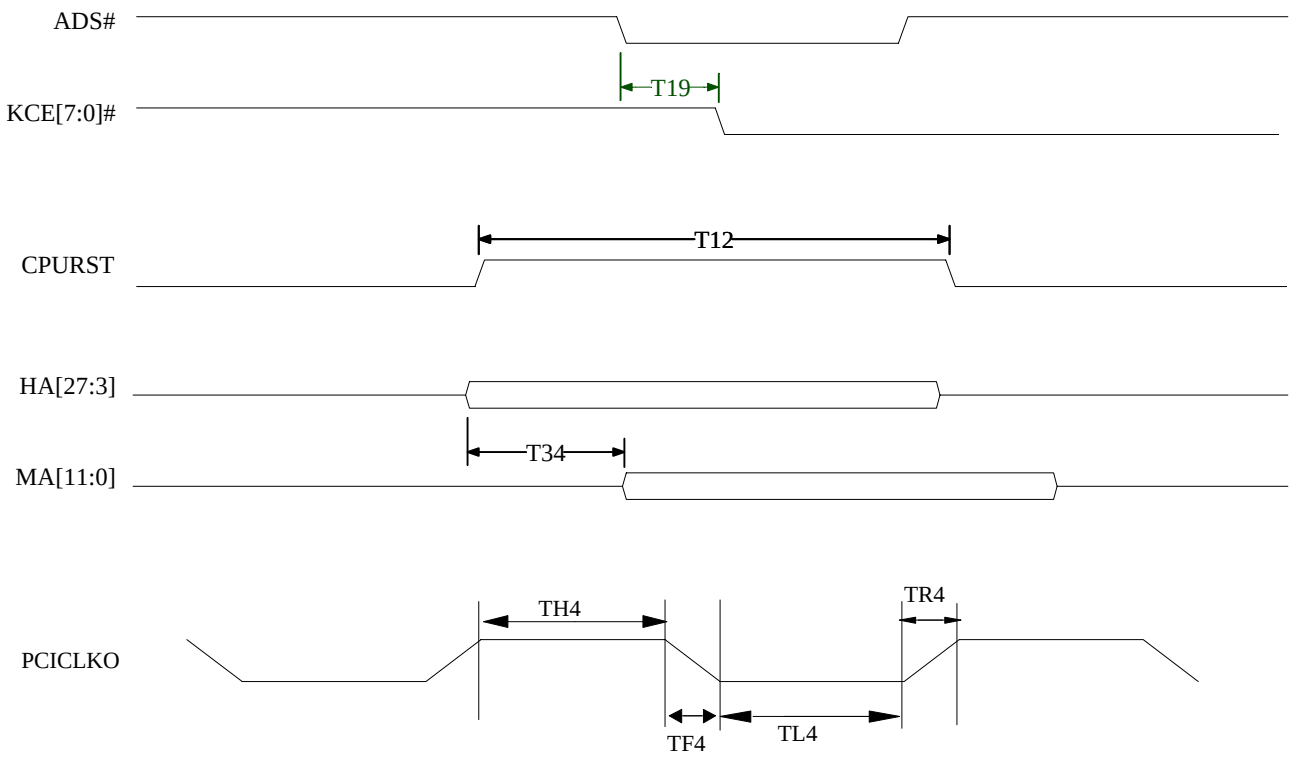
TF1 = T1, T3, T5, T7, T10, T11, T17, T22, T23, T25, T27, T28, T30, T32, T36, T38, T49, T50, T53, T55, T58, T60, T64, T78, T80, T83
TR1 = T2, T4, T6, T8, T9, T18, T20, T21, T24, T26, T27, T29, T31, T33, T35, T37, T39, T47, T48, T51, T52, T54, T56, T57, T59, T61, T62, T79, T81, T82
SIGNAL1 = BRDY#, KEN#, NA#, EADS#, CPUHOLD, CPURST, KWX[0:1]#, KWY[0:1]#, KCE[7:0]#, MDLE, CALE, KA4X, KA4Y, TA[7:0], RAS[7:0]#, CAS[7:0]#, MA[11:0], ALT, ALTWE#, TAGWE#, A20M#, HLDA, INIT#, PCICLKO, PCIRST, RAMW#, SMOUT, ADSC#, ADSV#, GNT[3:0]#, PAR, SERR#, SIOGNT#, STPCLK#, CPPSH



TF2 = T13, T15, T23
TR2 = T14, T16, T24
SIGNAL2 = KREX#, KREY#, KWX[0:1]#, KWY[0:1]#, KA4X, KA4Y



TF3 = T40, T41,T42, T46, T44, T65, T67, T68
TR3 = T40, T41, T43, T46, T45, T63, T66, T69
SIGNAL3 = AD[31:0], C/BE[3:0], ADLE#, ADOE, PRDLE, DEVSEL#, FRAME#, IRDY#,STOP#, TRDY#, HA[31:3], CPPOP, ADOE, ADLE



TH4 = T70,T72
TL4 = T71, T73
TR4 = T74, T76
TF4 = T75, T77

3 SiS5502

3.1 Features

- **Supports the Full 64-bit Pentium Processor Data Bus**
- **Provides a 64-Bit Interface to DRAM Memory**
- **Provides a 32-bit Interface to PCI**
- **Three Integrated Posted Write Buffers and Two Read Buffers Increase System Performance**
  - 1 level CPU-to-Memory Posted Write Buffer (CTMPB) with 4 QuadWords (QWs) Deep
  - 4 level CPU-to-PCI Posted Write Buffer (CTPPB) with 4 DoubleWords (DWs) Deep
  - 1 level PCI-to-Memory Posted Write Buffer (PTMPB) with 1 QW Deep
  - 1 level Memory-to-CPU Read Buffer (CRMB) with 1 QW Deep
  - 1 level Memory-to-PCI Read Buffer (PRMB) with 1 QW Deep
- **Near Zero Wait State Performance on CPU-to-Memory and CPU-to-PCI writes**
- **Operates Synchronously to the 66.7 MHz CPU and 33.3 MHz PCI Clocks**
- **Provides Parity Generation for Memory Writes**
- **208-Pin PQFP**
- **0.6 um CMOS Technology**

3.2 General Description

The SiS5502 PCI Local Data Buffer(PLDB) provides a bi-directional data buffering among the 64-bit Host Data Bus, the 64-bit Memory Data Bus, and the 32-bit PCI Address/Data Bus. The PLDB incorporates three Posted Write Buffers and two Read Buffers along 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 (CTPPB) provides buffering on CPU write to PCI bus. A one level/4QWs deep write buffer (CTMPB) is used for buffering write data from the CPU to Memory. A one level/1QW deep write buffer (PTMPB) is used to buffer PCI write to Memory data. A one QW Read Buffer (CRMB) is used to latch CPU read Memory data and a one QW Read Buffer (PRMB) is used to latch data in a PCI master read from L2 Cache or DRAM cycle. 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. Figure 3.1 shows the PLDB block diagram.

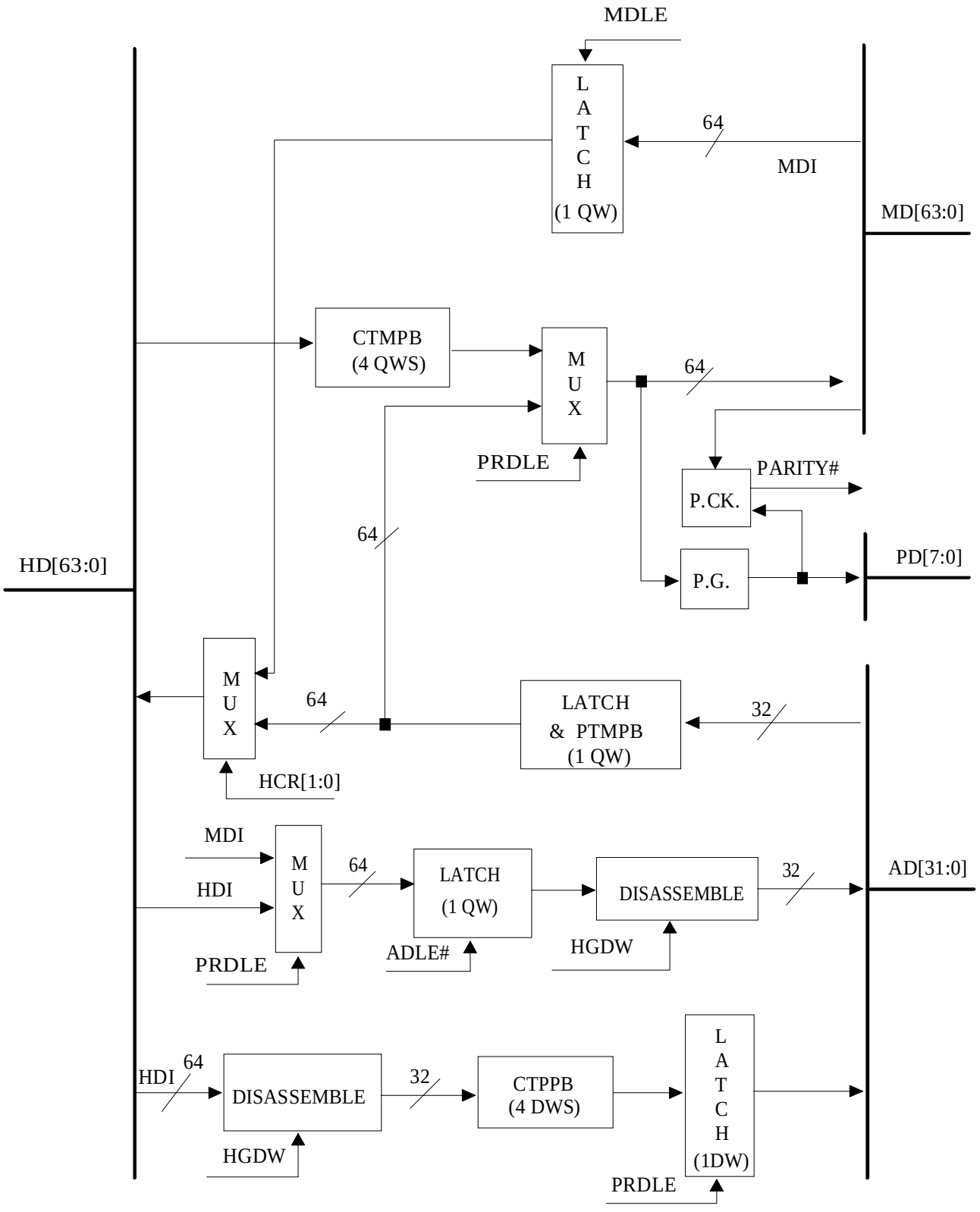


Figure 3.1 PLDB Block Diagram

3.3 Functional Description

3.3.1 Data Flow Between HD Bus and MD Bus

- **HD Bus to MD Bus**

Data flows from HD bus to MD bus when the CPU writes to local memory or the PCMC writes back a dirty line from the L2 cache to local memory in a CPU read miss/line fill cycle. All the data written to memory are first pushed into the CPMPB. The data are then popped from the buffer and written to Memory.

- **MD Bus to HD Bus**

During a CPU read local memory cycle, the data read are first latched in the 64-bit read buffer (CRMB) in order to provide enough hold time for the CPU and the L2 cache. The PLDB also checks the parity on the read data.

3.3.2 Data Flow Between HD Bus and AD Bus

- **HD Bus to AD Bus**

This path is used in the following two cases. The first case is in a CPU writes PCI slave cycle. The second case is in a PCI master read cycle that hits modified data in local cache which is implemented using write-back policy. All the CPU data sent to PCI memory slave is first pushed into the CTPPB. The data are then popped onto the PCI AD bus at later time when the PCI bus is not busy. Any further write to the PCI bus is suspended if the CTPPB is full. The I/O writes are not posted, but still exploit the CTPPB write buffer.

The path for PCI master read from the L2 cache is implemented through the PRMB, a built-in 64-bit PCI read memory buffer. Since one QW is read each time, the PCMC always sustains 1 wait state for reading the second DW.

- **AD Bus to HD Bus**

This path is exercised in two cases. The first case is during CPU reads PCI slave and the second case is during PCI master write cycles. All the CPU reads PCI cycle is stalled until the CTPPB is empty. When the CPU reads PCI slave, the data are latched and assembled in the PTMPB before they are transmitted to HD bus. During PCI master writes to local memory, the PCI data are first posted in the PTMPB. They are then transferred to local memory and host bus if the PCI master write also hits L2 cache.

3.3.3 Data Flow Between AD Bus and MD Bus

- **AD Bus to MD Bus**

Write data from PCI master is buffered in PTMPB before transferred to local memory. Parity is generated for memory write data.

- **MD Bus to AD Bus**

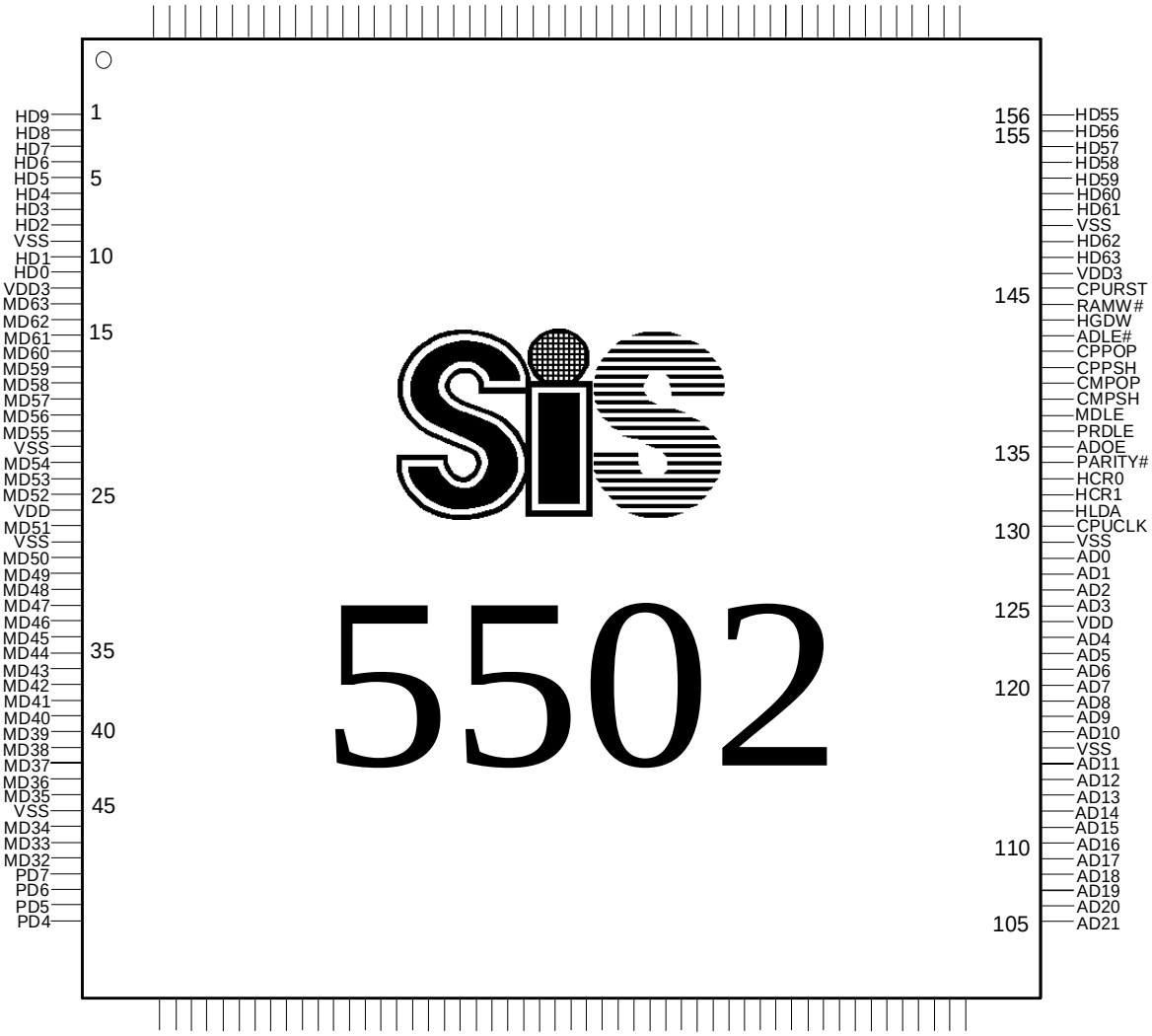
PCI masters receive data from local memory through this path. The PRMB, a 64-bit PCI read Memory buffer is implemented on this path. The read parity is ignored inside the PLDB.

3.3.4 Address Flow and Data Flow of Basic Cycles

Cycles	Address Flow	Data Flow
1. CPU/R/PCI	HA➜5501➜AD	AD➜5502➜HD
2. CPU/W/PCI	HA➜5501➜AD	HD➜5502➜AD
3. CPU/R/ISA	HA➜5501➜AD➜5503➜ LA,SA	SD➜5503➜AD➜5502➜ HD
4. CPU/W/ISA	HA➜5501➜AD➜5503➜ LA,SA	HD➜5502➜AD➜5503➜ SD
5. CPU/R/DRAM	HA➜5501➜MA	MD➜5502➜HD
6. CPU/W/DRAM	HA➜5501➜MA	HD➜5502➜MD
7. CPU/R/L2	Independent	Independent
8. CPU/W/L2	Independent	Independent
9. CPU/R/PCI(master abort)	HA➜5501➜AD	5502➜HD
10. PCI/R/L2	AD➜5501➜HA	HD➜5502➜AD
11. PCI/W/L2	AD➜5501➜HA	AD➜5502➜HD
12. PCI/R/DRAM	AD➜5501➜HA	MD➜5502➜AD
13. PCI/W/DRAM	AD➜5501➜HA	AD➜5502➜MD
14. ISA/R/L2	LA,SA➜5503➜AD➜5501 ➜HA	HD➜5502➜AD➜5503➜ SD
15. ISA/W/L2	LA,SA➜5503➜AD➜5501 ➜HA	SD➜5503➜AD➜5502➜ HD
16. DMA/R/L2	5503➜AD➜5501➜HA, 5503➜LA,SA	HD➜5502➜AD➜5503➜ SD
17. DMA/W/L2	5503➜AD➜5501➜HA, 5503➜LA,SA	SD➜5503➜AD➜5502➜ HD
18. ISA/R/DRAM	LA,SA➜5503➜AD➜5501 ➜MA	MD➜5502➜AD➜5503➜ SD
19. ISA/W/DRAM	LA,SA➜5503➜AD➜5501 ➜MA	SD➜5503➜AD➜5502➜ MD
20. DMA/R/DRAM	5503➜AD➜5501➜MA, 5503➜LA,SA	MD➜5502➜AD➜5503➜ SD
21. DMA/W/DRAM	5503➜AD➜5501➜MA, 5503➜LA,SA	SD➜5503➜AD➜5502➜ MD
22. ISA Refresh	5503➜SA	

3.4 Pin Assignment and Description

3.4.1 Pin Assignment



3.4.2 Pin Listing (# means active low)

1=HD9	5V/3.3V	48=MD32	5V	95=AD29	5V
2=HD8	5V/3.3V	49=PD7	5V	96=AD28	5V
3=HD7	5V/3.3V	50=PD6	5V	97=AD27	5V
4=HD6	5V/3.3V	51=PD5	5V	98=PCICLK	5V
5=HD5	5V/3.3V	52=PD4	5V	99=VSS	
6=HD4	5V/3.3V	53=MD31	5V	100=AD26	5V
7=HD3	5V/3.3V	54=MD30	5V	101=AD25	5V
8=HD2	5V/3.3V	55=MD29	5V	102=AD24	5V
9=VSS		56=MD28	5V	103=AD23	5V
10=HD1	5V/3.3V	57=MD27	5V	104=AD22	5V
11=HD0	5V/3.3V	58=MD26	5V	105=AD21	5V
12=VDD3	5V/3.3V	59=VDD	5V	106=AD21	5V
13=MD63	5V	60=MD25	5V	107=AD19	5V
14=MD62	5V	61=MD24	5V	108=AD18	5V
15=MD61	5V	62=MD23	5V	109=AD17	5V
16=MD60	5V	63=VSS		110=AD16	5V
17=MD59	5V	64=MD22	5V	111=AD15	5V
18=MD58	5V	65=MD21	5V	112=AD14	5V
19=MD57	5V	66=MD20	5V	113=AD13	5V
20=MD56	5V	67=MD19	5V	114=AD12	5V
21=MD55	5V	68=MD18	5V	115=AD11	5V
22=VSS		69=MD17	5V	116=VSS	
23=MD54	5V	70=MD16	5V	117=AD10	5V
24=MD53	5V	71=MD15	5V	118=AD9	5V
25=MD52	5V	72=MD14	5V	119=AD8	5V
26=VDD	5V	73=MD13	5V	120=AD7	5V
27=MD51	5V	74=MD12	5V	121=AD6	5V
28=VSS		75=MD11	5V	122=AD5	5V
29=MD50	5V	76=MD10	5V	123=AD4	5V
30=MD49	5V	77=MD9	5V	124=VDD	5V
31=MD48	5V	78=MD8	5V	125=AD3	5V
32=MD47	5V	79=MD7	5V	126=AD2	5V
33=MD46	5V	80=VSS		127=AD1	5V
34=MD45	5V	81=MD6	5V	128=AD0	5V
35=MD44	5V	82=MD5	5V	129=VSS	
36=MD43	5V	83=MD4	5V	130=CPUCLK	5V
37=MD42	5V	84=MD3	5V	131=HLDA	5V
38=MD41	5V	85=MD2	5V	132=HCR1	5V
39=MD40	5V	86=MD1	5V	133=HCR0	5V
40=MD39	5V	87=MD0	5V	134=PARITY#	5V
41=MD38	5V	88=PD3	5V	135=ADOE	5V
42=MD37	5V	89=PD2	5V	136=PRDLE	5V
43=MD36	5V	90=VDD	5V	137=MDLE	5V
44=MD35	5V	91=PD1	5V	138=CMPSH	5V
45=VSS		92=PD0	5V	139=CMPOP	5V
46=MD34	5V	93=AD31	5V	140=CPPSH	5V
47=MD33	5V	94=AD30	5V	141=CPPOP	5V

142=ADLE#	5V	176=HD38	5V/3.3V
143=HGDW	5V	177=HD37	5V/3.3V
144=RAMW#	5V	178=HD36	5V/3.3V
145=CPURST	5V	179=HD35	5V/3.3V
146=VDD3	5V/3.3V	180=HD34	5V/3.3V
147=HD63	5V/3.3V	181=HD33	5V/3.3V
148=HD62	5V/3.3V	182=HD32	5V/3.3V
149=VSS		183=VSS	
150=HD61	5V/3.3V	184=HD31	5V/3.3V
151=HD60	5V/3.3V	185=HD30	5V/3.3V
152=HD59	5V/3.3V	186=HD29	5V/3.3V
153=HD58	5V/3.3V	187=HD28	5V/3.3V
154=HD57	5V/3.3V	188=HD27	5V/3.3V
155=HD56	5V/3.3V	189=HD26	5V/3.3V
156=HD55	5V/3.3V	190=HD25	5V/3.3V
157=HD54	5V/3.3V	191=HD24	5V/3.3V
158=HD53	5V/3.3V	192=HD23	5V/3.3V
159=HD52	5V/3.3V	193=HD22	5V/3.3V
160=VSS		194=VSS	
161=HD51	5V/3.3V	195=HD21	5V/3.3V
162=HD50	5V/3.3V	196=HD20	5V/3.3V
163=HD49	5V/3.3V	197=HD19	5V/3.3V
164=HD48	5V/3.3V	198=VDD3	5V/3.3V
165=HD47	5V/3.3V	199=HD18	5V/3.3V
166=HD46	5V/3.3V	200=HD17	5V/3.3V
167=HD45	5V/3.3V	201=HD16	5V/3.3V
168=HD44	5V/3.3V	202=HD15	5V/3.3V
169=VDD3	5V/3.3V	203=HD14	5V/3.3V
170=HD43	5V/3.3V	204=HD13	5V/3.3V
171=HD42	5V/3.3V	205=HD12	5V/3.3V
172=VSS		206=VSS	
173=HD41	5V/3.3V	207=HD11	5V/3.3V
174=HD40	5V/3.3V	208=HD10	5V/3.3V
175=HD39	5V/3.3V		

3.4.3 Pin Description

Pin No.	Symbol	Type	Function
147,148 150-159 161-168 170,171 173-182 184-193 195-197 199-205 207,208 1-8,10,11	HD[63:0]	I/O	CPU data bus.
13-21,23-25, 27,29-44,46-48, 53-58 60-62,64-79, 81-87	MD[63:0]	I/O	Memory data bus.
93-97, 100-115, 117-123 125-128	AD[31:0]	I/O	PCI address/data bus.
91,92,49-52, 88,89	PD[7:0]	I/O	Parity bit bus.
134	PARITY#	O	Parity Error signal.
135	ADOE	I	Drive PCI AD bus. This signal is used to enable the 5502 to drive PCI AD bus. It is asserted in CPU writes PCI or PCI master reads local memory cycles.
131	HLDA	I	Hold Acknowledge is asserted in response to the assertion of CPUHLDA.
137	MDLE	I	Memory data Read Latch Enable. This signal latches the data on the MD bus when negated.
142	ADLE#	I	AD Bus Data Latch Enable. This signal has the following functions: 1. Latch HD or MD data into the PCI read buffer (PRMB) 2. Latch AD data into CPU read PCI buffer on the rising edge of PCICLKI. 3. Latch AD data into PCI posted write buffer (PTMPB) on the rising edge of PCICLKI.

140	CPPSH	I	Push Post Write Data into the CTPPB of the 5502. The data on the HD bus is latched into the CTPPB on CPPSH rising edge. The edge also increases the write pointer to the next available loading entry in the buffer.
141	CPPOP	I	On the rising edge of CPPOP, the read pointer is changed to address the next available reading location.
138	CMPSH	I	When this signal is asserted, the data on the HD bus is written into the CPU to memory posted write buffer (CTMPB) on the rising edge of CPUCLK, and the write pointer is also changed to address the next available location.
139	CMPOP	I	When this signal is asserted, the read pointer of the CTMPB is increased on the rising edge of CPUCLK.
136	PRDLE	I	This signal latches the current output entry of the CTPPB, the CPU to PCI post write buffer, into the prelatch in the 5502. The output of the prelatch is driven to the PCI AD bus. In the PCI master cycles, PRDLE is also asserted when PCI master is reading data from the secondary cache, or when PCI master is writing data to local memory.
143	HGDW	I	High Double Word Indicator. The signal is driven high by the 5501 when : (1) a high DW from HD bus is written into CPU to PCI Posted Write Buffer, (2) the CPU reads a high DW from PCI bus, (3) PCI master writes a high DW to local memory, (4) PCI master reads a high DW from local memory.
132,133	HCR[1:0]	I	Host Data Bus Control. These signals are driven by the 5501 and they are used to control the 5502 HD[63:0] bus. They are defined as: 00: 5502 floats HD bus 01: 5502 drives FFFFFFFF to HD bus 10: 5502 drives data from AD bus to HD bus 11: 5502 drives data from MD bus to HD bus
144	RAMW#	I	DRAM Write Enable.
145	CPURST	I	CPU Reset.
130	CPUCLK	I	CPU Clock.
98	PCICLK	I	PCI Bus Clock.
26,59,90, 124	VDD		+5V DC power

12, 146, 169, 198	VDD3		+3.3V DC power in 3V system +5V DC power in 5V system
9,22,28,45 ,63,80,99, 116,129, 149,160 172,183, 194,206	VSS		Ground

3.5 Electrical Characteristics

3.5.1 Absolute Maximum Ratings

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.

3.5.2 DC Characteristics

TA = 0 - 70 ^{O}C, VSS = 0V , VDD=5V±5%, VDD3=3.3V±5%

Symbol	Parameter	Min	Max	Unit	Condition
V_{IL1}	Input Low Voltage	-0.3	0.8	V	***Note 1***, VDD3=3.3V
V_{IH1}	Input High Voltage	2.2	VDD3+0.3V	V	***Note 1***
V_{IL2}	Input Low Voltage	-0.3	0.8	V	***Note 2***
V_{IH2}	Input High Voltage	2.2	VDD+0.3	V	***Note 2***
V_{OL1}	Output Low Voltage		0.45	V	***Note 3***
V_{OH1}	Output High Voltage	2.4		V	***Note 3***
V_{OL2}	Output Low Voltage		0.4	V	***Note 4***
V_{OH2}	Output High Voltage	2.0	VDD3	V	***Note 4***
I_{OL1}	Output Low Current	4		mA	***Note 5***
I_{OH1}	Output High Current	4		mA	***Note 5***
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
ICC3	Power Supply Current of VDD3		27	mA	3.3V, 66MHz

Note:

1. V_{IL1} and V_{IH1} are applicable to HD[63:0].
2. V_{IL2} and V_{IH2} are applicable to MD[63:0], AD[31:0], CPURST, PD[7:0], CPUCLK, ADOE, HLDA, HCR[1:0], PRDLE, MDLE, CMPSH, CMPOP, CPPSH, CPPOP, ADLE#, HGDW, RAMW#.
3. V_{OL1} and V_{OH1} are applicable to MD[63:0], AD[31:0], PD[7:0], PARITY#.
4. V_{OL2} and V_{OH2} are applicable to HD[31:0].
5. I_{OL1} and I_{OH1} are applicable to HD[63:0], MD[63:0], AD[31:0], PD[7:0], PARITY#.

3.5.3 AC Characteristics

Symbol	Parameter	Min	Typ	Max	Fig
T1	MD Data Setup Time to MDLE falling		6		3.2, 3.9
T2	MD Data Hold Time to MDLE falling	2			3.2, 3.9
T3	HD Data Valid Delay from MD data valid	10	15		3.2
T4	ADLE# Setup Time to PCICLK rising	6			3.3,3.8
T5	ADLE# Hold Time to PCICLK rising	2			3.3,3.8
T6	HGDW Setup Time to PCICLK rising	6			3.3,3.8
T7	HGDW Hold Time to PCICLK rising	2			3.3,3.8
T8	AD Data Setup Time to PCICLK rising	6			3.3,3.8
T9	AD Data Hold Time to PCICLK rising	2			3.3,3.8
T10	HD Data Valid Delay from PCICLK rising	10	15		3.3,3.8
T11	CMPSH Setup Time to CPUCLK rising	6			3.4
T12	CMPSH Hold Time to CPUCLK rising	2			3.4
T13	CMPOP Setup Time to CPUCLK rising	6			3.4
T14	CMPOP Hold Time to CPUCLK rising	2			3.4
T15	HD Data Setup Time to CPPSH rising	6			3.4,3.5
T16	HD Data Hold Time to CPPSH rising	2			3.4,3.5
T17	MD Data Valid Delay from CPUCLK rising	12	20		3.4
T18	PD Data Valid Delay from CPUCLK rising	15	25		3.4
T19	RAMW# Setup Time to MDLE rising				3.7
T20	RAMW# Hold Time to MDLE falling				3.7
T21	PARITY# Active Delay from MDLE falling	6	9		3.7
T22	AD Data Valid from PRDLE rising	7	11		3.6
T23	AD Data Valid from MD data valid	12	18		3.10
T24	HGDW Setup Time to CPPSH rising	6			3.5
T25	HGDW Hold Time to CPPSH rising	2			3.5
T26	MD Data Valid Delay from PCICLK rising	12	18		3.8
T27	PD Data Valid Delay from PCICLK rising	15	23		3.8
T28	HD Data Setup Time to ADLE# falling	6			3.9
T29	HD Data Hold Time to ADLE# falling	2			3.9
T30	AD Data Valid Delay from HD data valid	12	18		3.9
T31	MD Output Delay from RAMW# asserted	5	7.5		3.11

T32	MD Output Float Delay from RAMW# inactive	25	43		3.11
T33	AD Output Delay from ADOE asserted	4	6		3.12
T34	AD Output Float Delay from ADOE inactive			26	3.12
T35	HD Output Delay from HCR asserted	6	9		3.13
T36	HD Output Float Delay from HCR inactive				3.13

Unit :ns

3.5.4 AC Timing Diagram

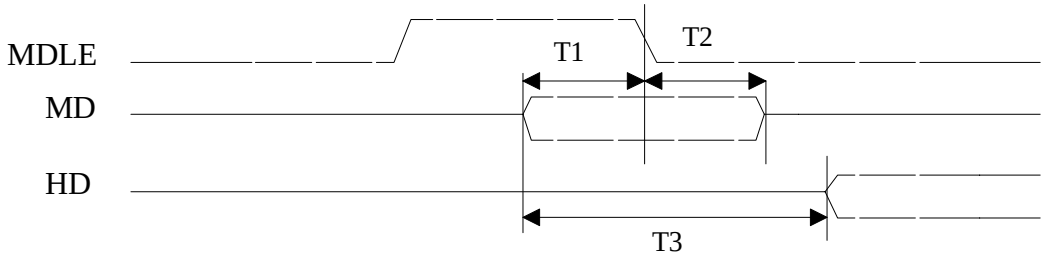


Figure 3.2 CPU Read DRAM Cycle

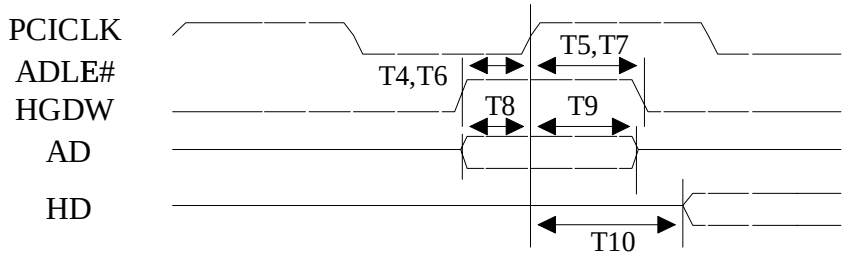


Figure 3.3 CPU Read PCI Slave Cycle

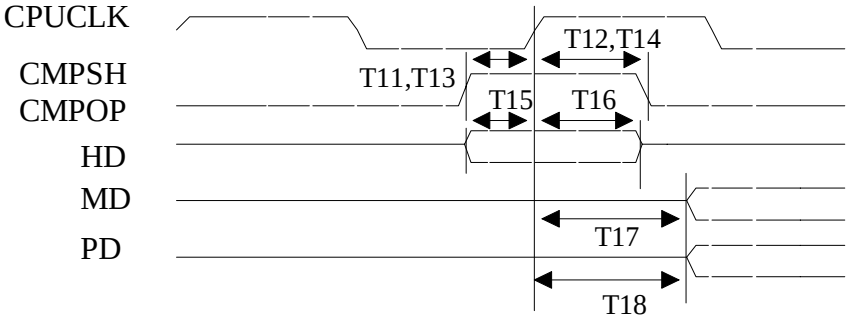


Figure 3.4 CPU Write DRAM Cycle

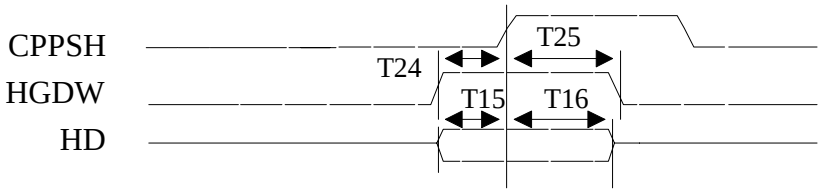


Figure 3.5 CPU Write PCI Post Write Buffer

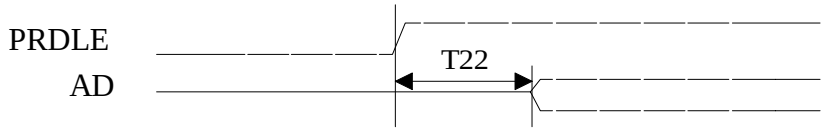


Figure 3.6 Write Posted Data onto PCI Bus

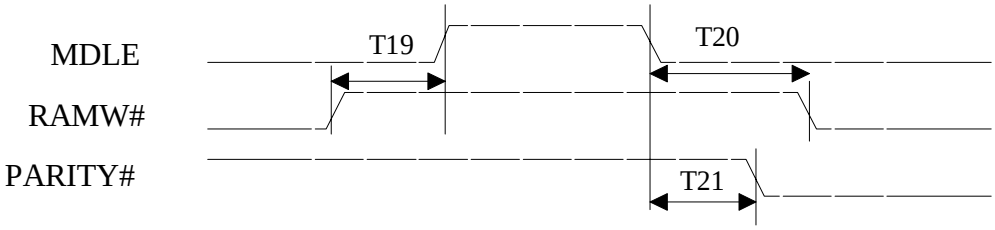


Figure 3.7 PARITY# Generation Reading DRAM Cycle

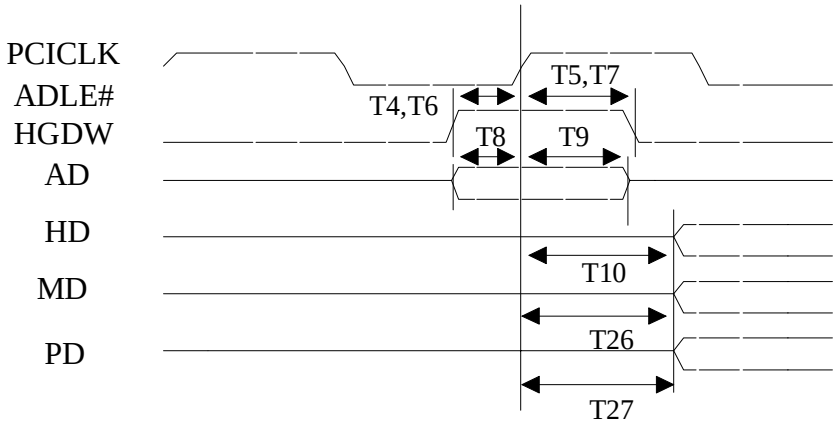


Figure 3.8 CPU Read PCI Slave Cycle

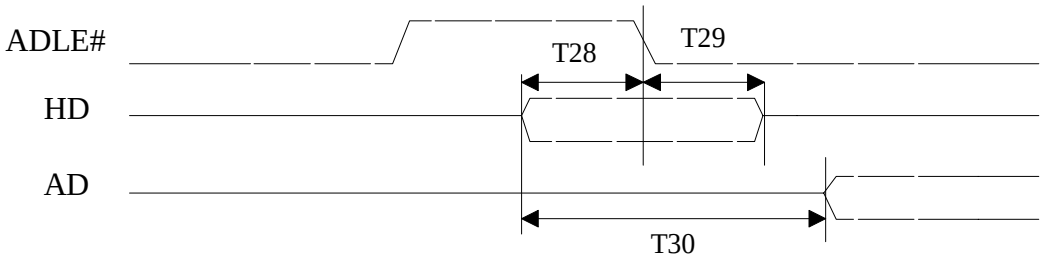


Figure 3.9 PCI Master Read Secondary Cache

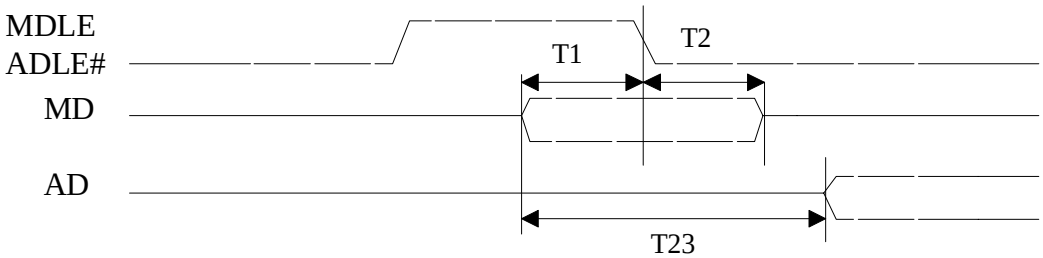


Figure 3.10 PCI Master Read DRAM

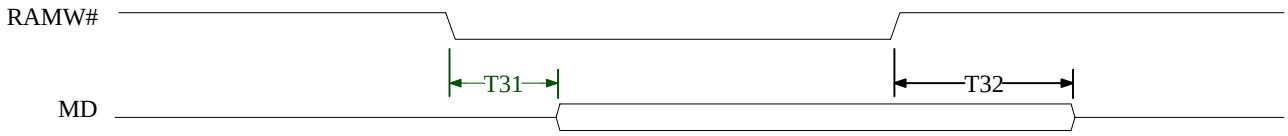


Figure 3.11 CPU Write DRAM Cycle

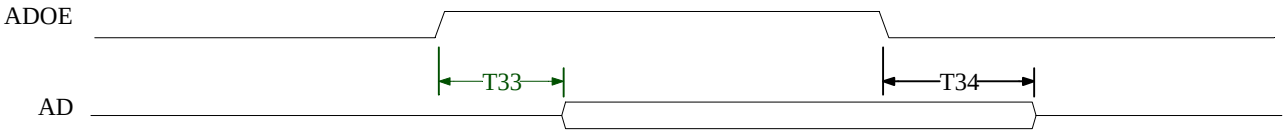


Figure 3.12 Write Posted Data onto PCI Bus

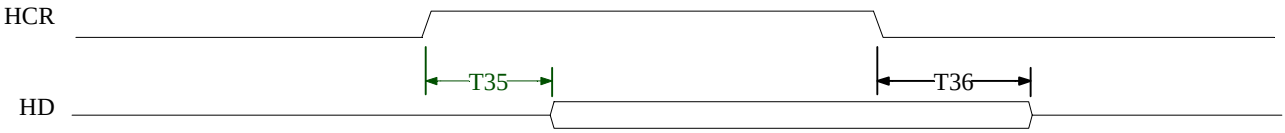


Figure 3.13 CPU Read DRAM or PCI Cycle

4 SiS5503

4.1 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 Reroutibilty of four PCI Interrupts to Any Unused IRQ Interrupt**
- **Supports Flash ROM**
- **Built-in RTC with 242 Bytes Extended CMOS SRAM**
- **Built-in PCI IDE**
  - Fully Compatible with PCI Local Bus Specification V2.0.
  - Accommodates 8 Bits, 16 Bits, and 32 Bits Data Transfer.
  - Supports PCI Burst Read/Write Operation.
  - Supports Read Ahead & Posted Write Buffers for Concurrent System Operation.
  - Controls Two IDE Channels and Max. Connects 4 IDE Drives.
  - Supports PIO Mode 3 Timing IDE Specification.
  - Programmable Command and Recovery Timing for Reads and Writes Per Channel.
  - Auto IDE Channel Speed Setting with Software Driver.
  - Hardware and Software Chip Disable Capability
  - Supports Power Down Feature
- **Meet PCI Specification Buffer Strength**
- **160-Pin PQFP**
- **0.6 µm CMOS Technology**

4.2 Functional Description

The SiS5503 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 SiS5503 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 242 bytes extended CMOS SRAM, and a built-in PCI IDE.

Since 5503 includes a PCI to ISA bridge and a PCI IDE, it naturally becomes a multifunction device. The PCI/ISA bridge is defined as function 0 device while PCI IDE is 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 SiS5503 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 SIOGNT# is asserted, the master bridge translates the ISA master or DMA cycles onto the PCI bus based on the decoding status from ISA address decoder. When SIOGNT# 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.

SiS5503 always converts the single interrupt acknowledge cycle (from 5501) 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 SIOGNT# 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 SiS5503 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 SiS5503. The SiS5503 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 SiS5503 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 SiS5503 can only support 24-bit address for DMA devices.

4.2.4 Interrupt Controller

The SiS5503 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.

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 SiS5503 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 5503 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 80h to I/O port 22h and then setting bit 3 of I/O port 23h.

4.2.7 Built-in PCI IDE

The internal PCI IDE contains five blocks. They are PCI Bus interface and decode, system configuration & control, IDE interface ckt, read ahead buffers, and posted write buffers.

PCI Bus Interface and Decode

The internal PCI IDE operates as a slave device. It 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 R/W are 8-bit only.

System Configuration & Control & PCI Configuration

This block contains PCI configuration header and registers to meet PCI specifications.

The internal PCI IDE supports PCI type 0 configuration cycles of configuration mechanism #1.

IDE Interface Ckt

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

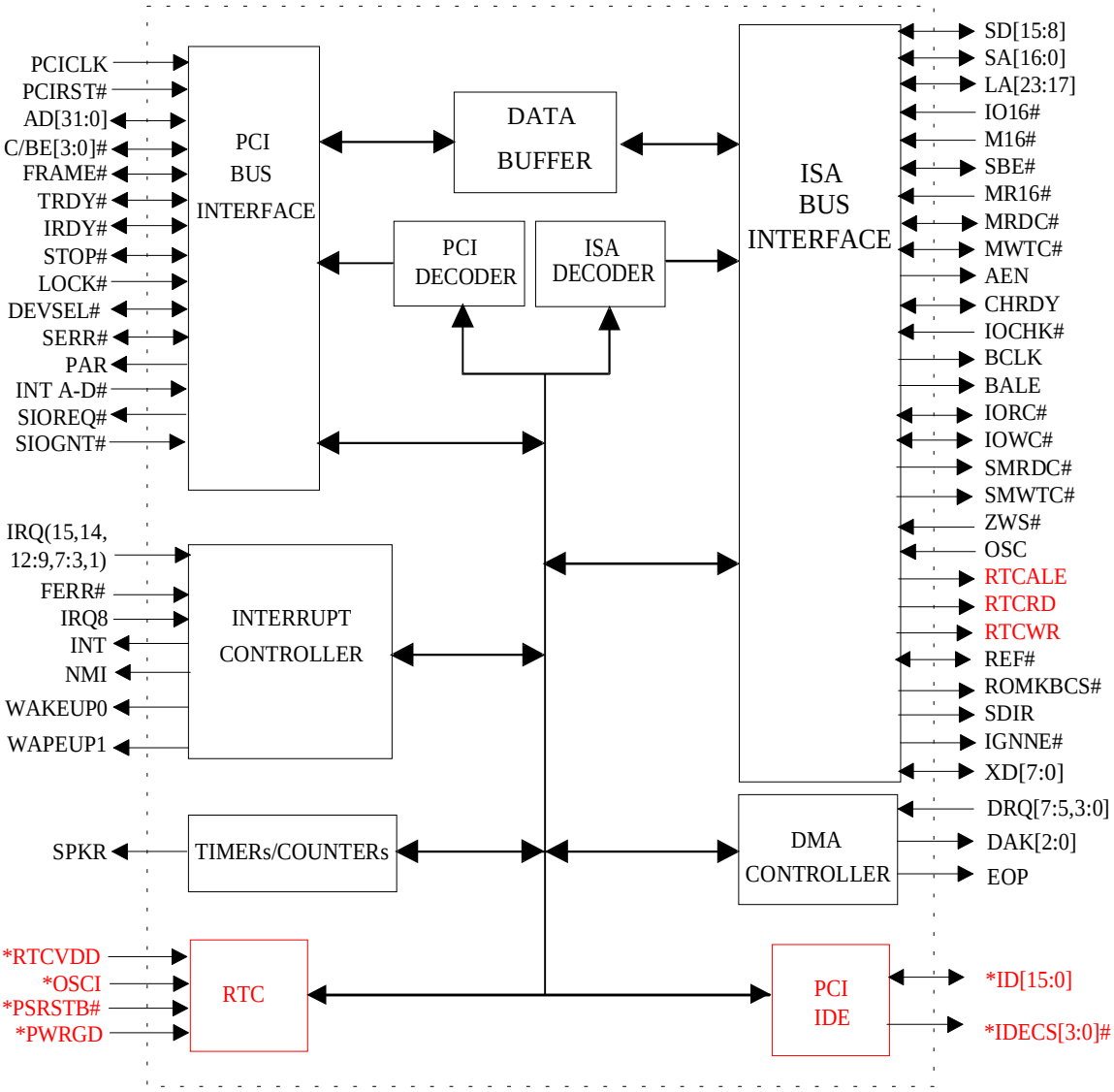
Posted Write Buffers & Read Ahead Buffers

The internal PCI IDE has two kinds of buffers, posted write buffers and read ahead buffers. They can be enabled or disabled independently.

The posted write buffers 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 read ahead buffers can eliminate the idle cycle of the PCI Bus side to improve read operation.

4.3 Functional Block Diagram



* : Multi-function pin

Figure 4.1 SiS5503 Functional Block Diagram

4.4 Configuration Registers

Registers 00h, 01h VID - Vendor Identification Register

Bits 15:0 1039h (Read Only)

Registers 02h, 03h DID - Device Identification Register

Bits 15:0 0008h (Read Only)

Registers 04h, 05h COM - Command Register = 07h

Bits 15:4 Reserved. Read as 0's

Bit 3 SCE (Special Cycle Enable) = 0

Bit 2 BME (Bus Master Enable) = 1

Bit 1 MSE (Memory Space Enable) = 1

Bit 0 IOSE (I/O Space Enable) = 1

Registers 07h-06h DS - Device Status Register

Bits 15:14 Reserved. Read as 0's

Bit 13 MA (Master-Abort Status).

When the 5503 generates a master-abort, MA is set to a 1. Software clears MA to 0 by writing a 1 to this bit location.

Bit 12 RTA (Received Target-Abort Status).

When the 5503 receives a target-abort, RTA is set to a 1. Software clears RTA to 0 by writing a 1 to this bit location.

Bit 11 Reserved. Read as a 0

Bits 10:9 DEVT (SIO DEVSEL# Timing Status).

The 5503 always generates DEVSEL# with medium timing, DEVT=01

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

Register 08h RID - Revision Identification Register

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**

Register 40h **BIOSCON - 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 SIOGNT#. This bit defaults to 0.

Bit 4 **PCI Posted Write Buffer Enable**

The default value is 0 (disabled).

Bits [3:0] determine how the 5503 responds to F segment, E segment, and extended segment (FFF80000-FFFDFFFF) accesses. 5503 will positively respond to extended segment access when bit 0 is set. Bit 1, combining with bits [3:2], enables 5503 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			√ *		5503 positively responds to E segment access.
01		√			5503 subtractively responds to F segment access.
10	√		√ *		5503 positively responds to E and F segment access.
11	√				5503 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

Bits 7	6	5	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 **F0000h~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.

Bit 7 **A23**

Bit 6 **A22**

Bit 5 **A21**

Bit 4 **A20**

Bit 3 **A19**

Bit 2 **A18**

Bit 1 **A17**

Bit 0 **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.

Bit 7 **A23**

Bit 6 **A22**

Bit 5 **A21**

Bit 4 **A20**

Bit 3 **A19**

Bit 2 **A18**

Bit 1 **A17**

Bit 0 **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.**

4.5 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.

4.6 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 80h

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 **Programmable Output Pin**

0: Pin 20 is used as WAKEUP0 when internal IDE is disabled

1: Pin 20 is used as "Programmable Output Pin" that can generate one write pulse by writing register 82h when internal IDE is disabled

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 80h is set to 1. The new added feature will allow EPROM to be flashed anytime. Bit 2 of the register 80h is added and the setting of both bit 2 and bit 5 will now control the EPROM flash operation.

Register 80h bit 5	Register 80h 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 **Relocatable ISA Configuration Registers Control**

ISA configuration registers are now relocatable through bit 1 of ISA configuration register 80h. Upon power on, ISA configuration registers are located between index 80h to 8Fh by default. These index can be relocated to 70h to 7Fh by programming bit 1 of register 80h to 1.

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 81h

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 **Internal PCI IDE Enable/Disable**

0: Enable

1: Disable

This function is available only when the internal PCI IDE is enabled by hardware tarp, pull SDIR# high.

Bit 0 **Reserved**

Register 82h **If this register is written by any values, the pin 20 will generate one write pulse when bit 4 of register 80h "Programmable Output Pin" is enabled**

Register 83h

Bits 7:2 **Reserved**

Bit 1 **ISA Bus Refresh Cycle Enable/Disable**

0: Enable

1: Disable

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

0: 50mA/2.2V (default value)

1: 95mA/2.2V

Register 84h **BIOS Register**

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

Register 85h

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

Register 88h

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 system 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 system standby state.

Register 89h

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 8Ah

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 8Bh

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.

4.7 PCI IDE Configuration Registers

31 16	15 0			
Device ID = 0601h		Vendor ID = 1039h		00h
Status = 0000-0000-0000-0000		Command = 0000-0000-1000-1001		04h
Base Class = 01h	Sub-Class = 01h	Prog. If = 00h	Revision ID = 00h	08h
BIST = 00h	Header Type = 80h	Latency Timer = 00h	Cache Line Size = 00h	0Ch
XXXX17xx		0	1	10h
XXXX1Fxx		0	1	14h
XXXX37xx		0	1	18h
XXXX3Fxx		0	1	1Ch
Reserved (00000000)		0	1	20h
Reserved (00000000)		0	1	24h
Reserved (00000000)				28h
Reserved (00000000)				2Ch
Expansion ROM Base Address None (00000000)				30h
Reserved (00000000)				34h
Reserved (00000000)				38h
Max_Lat = 00h	Min_Gnt = 00h	Interrupt Pin = 00h	Interrupt Line	3Ch

Register 40h Built-in PCI IDE Control Register 1

Bit 7 IDE Channel and Address Select Bit 0

Bit 6 Auto Power Down Mode

0: Disable

1: Enable

Bits 5:3 Channel 1 Wait State (B2-B0)

Bits 2:0 Channel 0 Wait State (B2-B0)

	READ (PCICLKs)				WRITE (PCICLKs)			
Wait-State B3B2B1B0	**Recovery**			**Active**	**Recovery**			**Active**
	St.[1]	**1st**[1]	**2nd**[1]		**St.**	**1st**	**2nd**	
000	5	4	3	3	7	6	5	4
001	5	4	3	4	11	10	9	4
010	5	4	3	6	11	10	9	6
011	7	6	5	6	13	12	11	6

100	7	6	5	8	13	12	11	8
101	9	8	7	8	15	14	13	8
110	11	10	9	8	19	18	17	8
111(default)	13	12	11	8	19	18	17	8

***Note*:** 1. **St.** means Standard Mode, **1st** means 1st Enhanced Mode and **2nd** means 2nd Enhanced Mode. For the detailed information, please refer to "Register 41h".

2. The above table contains normal cycle timing. When the read ahead buffers and posted write buffers are enabled, the cycle time for read cycle is 4T, and for write cycle, 6T.

Register 41h Built-in PCI IDE Control Register 2

Bits 7 Post Write Buffer

0: Disable

1: Enable

Bit 6 Reserved, must be "1"

Bit 5 Reserved, must be "0"

Bit 4 1st Enhanced Speed Mode (Recovery time = Standard Speed - 1T)

0: Disable

1: Enable

Bit 3 Address Remapping

0: Disable

1: Enable

Bit 2 Read Prefetch Buffer

0: Disable

1: Enable

Bit 1 2nd Enhanced Speed Mode (Recovery time = 1st Enhanced Speed - 1T)

0: Disable

1: Enable

Bit 0 IDE Channel and Address Select Bit 1

Reg. 41h bit 0	Reg. 40h bit 7	
0	0	CH0 1FX, 3F6 only(IDECS0, IDECS1)
0	1	CH0 17X, 376 only(IDECS0, IDECS 1)
1	0	CH0 1FX, 3F6/CH1 17X, 376(IDECS0, IDECS1, IDECS2, IDECS3)
1	1	Reserved

4.8 Pin Assignment and Description

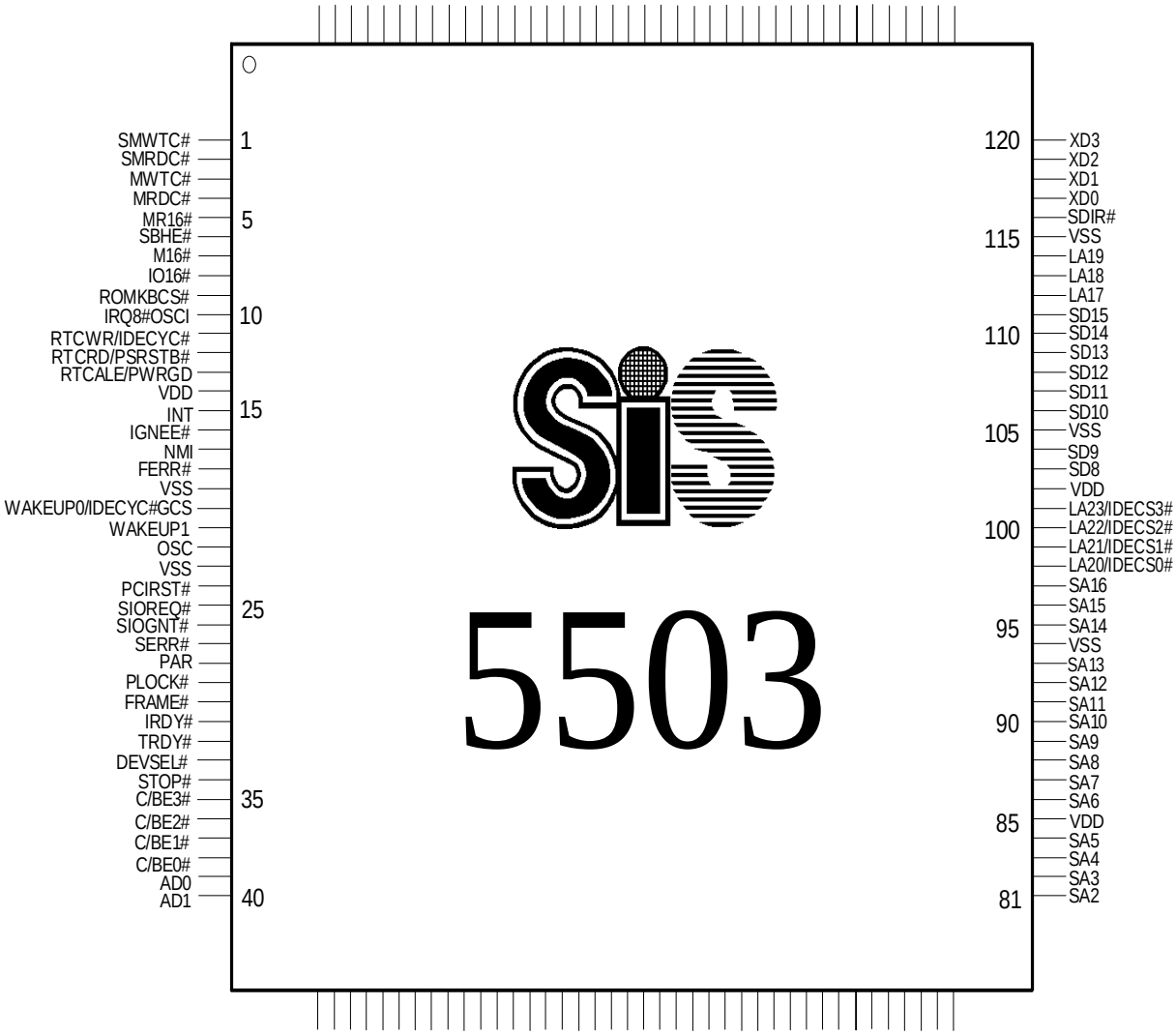
4.8.1 Hardware Trap

Pin No.	Symbol	Description
116	SDIR#	To enable the internal IDE, this signal needs to be pulled high via a 10K ohms resistor. Reversly, if this signal is pulled low, the internal IDE is disabled.
9	ROMKBCS#	If super I/O device with built-in RTC is used, this signal should be connected GND via a 10k ohms resistor.
13	RTCALE/PWRGD	This signal is an input pin upon power up. 5503 will strobe this pin on the rising edge of PCIRST# to identify that internal or external RTC is used. If this signal is sampled high, internal RTC is used. On the other hand, if it is sampled low(AS signal of external RTC is pulled low via a 4.7K ohms resistor), external RTC is employed.

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

5503 RTC	5503 IDE	ext. RTC	SIO RTC	pin 13	pin 12	pin 11	pin 10	pin 20
√	√			PWRGD	PSRSTB#	IDECYC#	OSCI	WAKEUP0 GCS
√				PWRGD	PSRSTB#	Undef.	OSCI	WAKEUP0 GCS
	√		√	Undef.	Undef.	IDECYC#	IRQ8#	WAKEUP0 GCS
	√	√		RTCALE	RTCRD#	RTCWR#	IRQ8#	IDECYC#
		√		RTCALE	RTCRD#	RTCWR#	IRQ8#	WAKEUP0 GCS
			√	Undef.	Undef.	Undef.	IRQ8#	WAKEUP0 GCS

4.8.2 Pin Assignment



4.8.3 Pin Listing (# means active low)

1=SMWTC#	41=AD2	81=SA2	121=XD4
2=SMRDC#	42=AD3	82=SA3	122=XD5
3=MWTC#	43=VDD	83=SA4	123=XD6
4=MRDC#	44=AD4	84=SA5	124=XD7
5=MR16#	45=AD5	85=VDD	125=EOP
6=SBHE#	46=AD6	86=SA6	126=SPKR
7=M16#	47=AD7	87=SA7	127=RFH#
8=IO16#	48=AD8	88=SA8	128=ZWS#
9=ROMKBCS#	49=AD9	89=SA9	129=BALE
10=IRQ8#/OSCI	50=AD10	90=SA10	130=IOCHK#
11=RTCWR/IDECYC#	51=AD11	91=SA11	131=CHRDY
12=RTCRD/PSRSTB#	52=AD12	92=SA12	132=AEN
13=RTCALE/PWRGD	53=AD13	93=SA13	133=IOWC#
14=VDD	54=AD14	94=VSS	134=IORC#
15=INT	55=AD15	95=SA14	135=VDD
16=IGNEE#	56=AD16	96=SA15	136=BCLK
17=NMI	57=AD17	97=SA16	137=IRQ1
18=FERR#	58=AD18	98=LA20/IDECS3#	138=IRQ3
19=VSS	59=AD19	99=LA21/IDECS2#	139=IRQ4
20=WAKEUP0 /IDECYC#/GCS	60=AD20	100=LA22/IDECS1#	140=IRQ5
21=WAKEUP1	61=AD21	101=LA23/IDECS0#	141=VSS
22=OSC	62=AD22	102=VDD	142=IRQ6
23=VSS	63=AD23	103=SD8	143=IRQ7
24=PCIRST#	64=AD24	104=SD9	144=IRQ9
25=SIOREQ#	65=AD25	105=VSS	145=IRQ10
26=SIOGNT#	66=AD26	106=SD10	146=IRQ11
27=SERR#	67=AD27	107=SD11	147=IRQ12
28=PAR	68=VSS	108=SD12	148=IRQ14
29=PLOCK#	69=PCICLK	109=SD13	149=IRQ15
30=FRAME#	70=AD28	110=SD14	150=RTCVDD
31=IRDY#	71=AD29	111=SD15	151=DAK2
32=TRDY#	72=AD30	112=LA17	152=DAK1
33=DEVSEL#	73=AD31	113=LA18	153=DAK0
34=STOP#	74=INTD#	114=LA19	154=DRQ0
35=C/BE3#	75=INTC#	115=VSS	155=DRQ1
36=C/BE2#	76=INTB#	116=SDIR#	156=DRQ2
37=C/BE1#	77=INTA#	117=XD0	157=DRQ3
38=C/BE0#	78=VSS	118=XD1	158=DRQ5
39=AD0	79=SA0	119=XD2	159=DRQ6
40=AD1	80=SA1	120=XD3	160=DRQ7

4.8.4 Pin Description

PCI Interface

Pin No.	Symbol	Type	Function
35-38	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 5503 is a PCI bus master and inputs when it is a PCI slave.
73-70 67-44 42-39	AD[31:0]	I/O	Address and data are multiplexed on AD[31:0]. During the address phase of a transaction, AD[31:0] contains a physical address. During the data phase, AD[31:0] contains data. When the 5503 is a PCI master, it drives address on AD[31:2] and drives AD[1:0] low during the address phase. During the data phase, it drives data or latches data on AD[31:0] for write or read cycle respectively. When the 5503 is a target, AD[31:0] are inputs during the address phase. During the data phases, the 5503 drives data on AD[31:0] for read cycle, or latches data for write cycle.
30	FRAME#	I/O	FRAME# is asserted to indicate the beginning of a bus transaction and asserted until the last data phase. When the PCI master is ready to complete the final data phase, it deasserts FRAME#. When the 5503 is the target, FRAME# is an input to the 5503. 5503 drives FRAME# out when it is the PCI master. FRAME# is tri-state during reset.
31	IRDY#	I/O	When 5503 is the PCI master, it drives IRDY# to complete the current data phase of the transaction. During write cycles, the assertion of IRDY# indicates the 5503 has driven valid data on AD[31:0]. During read cycles, it indicates the 5503 is ready to latch the data. IRDY# is an input to the 5503 when the 5503 is the target and an output when the 5503 is a master.

32	TRDY#	I/O	TRDY# is an output when the 5503 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 5503 is a PCI master, TRDY# is an input signal.
33	DEVSEL#	I/O	The 5503 asserts DEVSEL# when 5503's configuration registers or internal registers are addressed. DEVSEL# is also asserted when the 5503 subtractively decodes a cycle. When 5503 is the PCI master, DEVSEL# is an input to indicate a PCI target has responded to a 5503 initiated transaction. For all PCI transactions, the 5503 also samples DEVSEL# to decide to subtractively decode the cycle.
34	STOP#	I/O	When the 5503 is a target it asserts STOP# to request master to stop the current transaction. When 5503 is a master, the inputted STOP# causes the 5503 to stop the current transaction.
28	PAR	O	PAR is even parity across AD[31:0] and C/BE[3:0]# and regardless of whether or not all lines carry meaningful information. Both address and data phases the PAR is generated. PAR is driven one PCI clock after the corresponding address or data. PAR is driven by 5503 during the address phase of 5503 initiated transactions. During the data phase, 5503 also drives PAR when (1) 5503 is the master of a PCI write transaction. (2) 5503 is the target of a read transaction.
27	SERR#	I	The 5503 generates a NMI to the CPU when SERR# is active.
29	PLOCK#	I	The 5503 is locked when it samples PLOCK# is negated during the address phase. Any master attempt to access 5503 at this time, 5503 will initiate a retry to terminate the transaction. The locked state lasts until 5503 samples both PLOCK# and FRAME# are negated.

69	PCICLK	I	PCICLK provides the fundamental timing and the internal operating frequency for the 5503. PCICLK is a buffered input of 5501's PCICLKO.
24	PCIRST#		The PCI reset brings 5503's registers, signals, etc. to a known state.
77-74	INT[A:D]#	I	PCI Interrupt A to Interrupt D. INT[A:D]# can be remapped to one of eleven ISA compatible interrupts, please refer to registers 41h to 44h for more detailed information.

ISA Interface

97-95 93-86 84-79	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.
114-112	LA[19:17]	I/O	Latched system address. They are inputs when an external bus master is in control and are outputs at all other times.
124-117	XD[7:0]	I/O	Peripheral Data Bus lines.
111-106 104,103	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.
134	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.
133	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.
132	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.
131	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.
130	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.
129	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.
128	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.
136	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.

127	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.
22	OSC	I	OSC is the buffered input of the external 14.318MHz oscillator.
125	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.
160-154	DRQ7-5,3-0	I	DMA Request inputs are used by external devices to indicate when they need service from the internal DMA controllers.
137-140, 142,143, 144-149	IRQ1, 3-7, IRQ 9, 10-12,14,15	I	These are the asynchronous interrupt request inputs to the 8259 controller.

Multi-function Pins

Pin NO.	Symbol	Type	Function
10	IRQ8#/OSCI	I	• When using internal RTC: This pin is used as the time base of the built-in RTC. This signal is from the 32.768 KHz crystal oscillator. • When using external RTC: This pin is used as IRQ8#, which is the asynchronous interrupt request input to the 8259 controller.
12	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.

11	RTCWR/IDECYC#	O	• When using internal RTC or super I/O device's RTC: This pin is used as IDECYC#(IDE cycle indicator). • 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.
13	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.
20	WAKEUP0/IDECYC#/GCS	O	• When using internal RTC: The signal is directly connected to the 5501 when it is used as WAKEUP0. When activated, it will reload the 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. This pin can also be used to issue a logical high write signal with pulse width equal to IOWC#, and at this time the WAKEUP0 function doesn't exist any longer. This function can be enabled by setting bit 4 of I/O port 22h index 80h. Once it is enabled, the write pulse is generated by writing index register 82h. • When using external RTC and internal IDE: This pin is used as IDECYC#
101-98	LA[23:20]/IDECS[3:0]#	I/O	Normally, these signals are ISA latched system address. When IDECYC# is asserted, they become IDE Chip Select signals. Latched system address. They are inputs when an external bus master is in control and are outputs at all other times.

Others

151-153	DAK[2:0]	O	The DAK output lines indicate that a request for DMA service has been granted by the 5503 or that a 16 bit master has been granted the bus. One external 74138 decodes these signals to DAK[7:0] for the corresponding peripheral.
150	RTCVDD		Battery power for RTC
25	SIOREQ#	O	SIO Request. The 5503 SIOREQ# to request the PCI bus.
26	SIOGNT#	I	SIO Grant. It is driven by the 5501 to indicate that the PCI arbiter has granted the use of the PCI bus to the 5503.
9	ROMKBCS#	O	Keyboard or System ROM Chip Select
126	SPKR	O	Speaker is the output for the speaker.
116	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.
21	WAKEUP1	O	This signal is directly connected to the 5501. When activated, it will reload the 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.
18	FERR#	I	Floating point error from the CPU. It is driven active when a floating point error occurs.
16	IGNEE#	OD	IGNEE# is normally in high impedance state, and is asserted to inform CPU to ignore a numeric error. A resistor connected to either 3.3V (for P54C) or 5V (P5) is required to maintain a correct voltage level to CPU.
17	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.
15	INT	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.

14,43,85 102,135	VDD		+5V DC power
19,23,68 78,94,105 115,141	VSS		Ground

4.9 Electrical Characteristics

4.9.1 Absolute Maximum Ratings

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.

4.9.2 DC Characteristics

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

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/IDECYC#/GCS, WAKEUP1, SIOREQ#, PAR, C/BE3#, C/BE2#, C/BE1#, C/BE0#, AD[31:0], SDIR#, XD[7:0], SPKR, BCLK, DAK[2:0], INT, NMI, IGNEE#.

2. I_{OL2} and I_{OH2} are applicable to RFH#, CHRDY, AEN, BALE, EOP, SD[15:8].
3. I_{OL3} and I_{OH3} are applicable to SMWTC#, SMRDC#, MWTC#, MRDC#, SBHE#, LA[23:20]/IDECS[3:0]#, LA[21:17], SA[16:0], IOWC#, IORC#. Please refer to Register description.
4. I_{OL4} and I_{OH4} are applicable to FRAME#, IRDY#, TRDY#, DEVSEL#, STOP#.

4.9.3 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]

Para-meter	Description	Min	Typ	Max	Unit
ISA Bus Interface Signals (Figure 4.2)					
	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 4.3, 4.4)					
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 4.13)**					
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 4.5 ~ 4.9, 4.11, 4.12)					
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

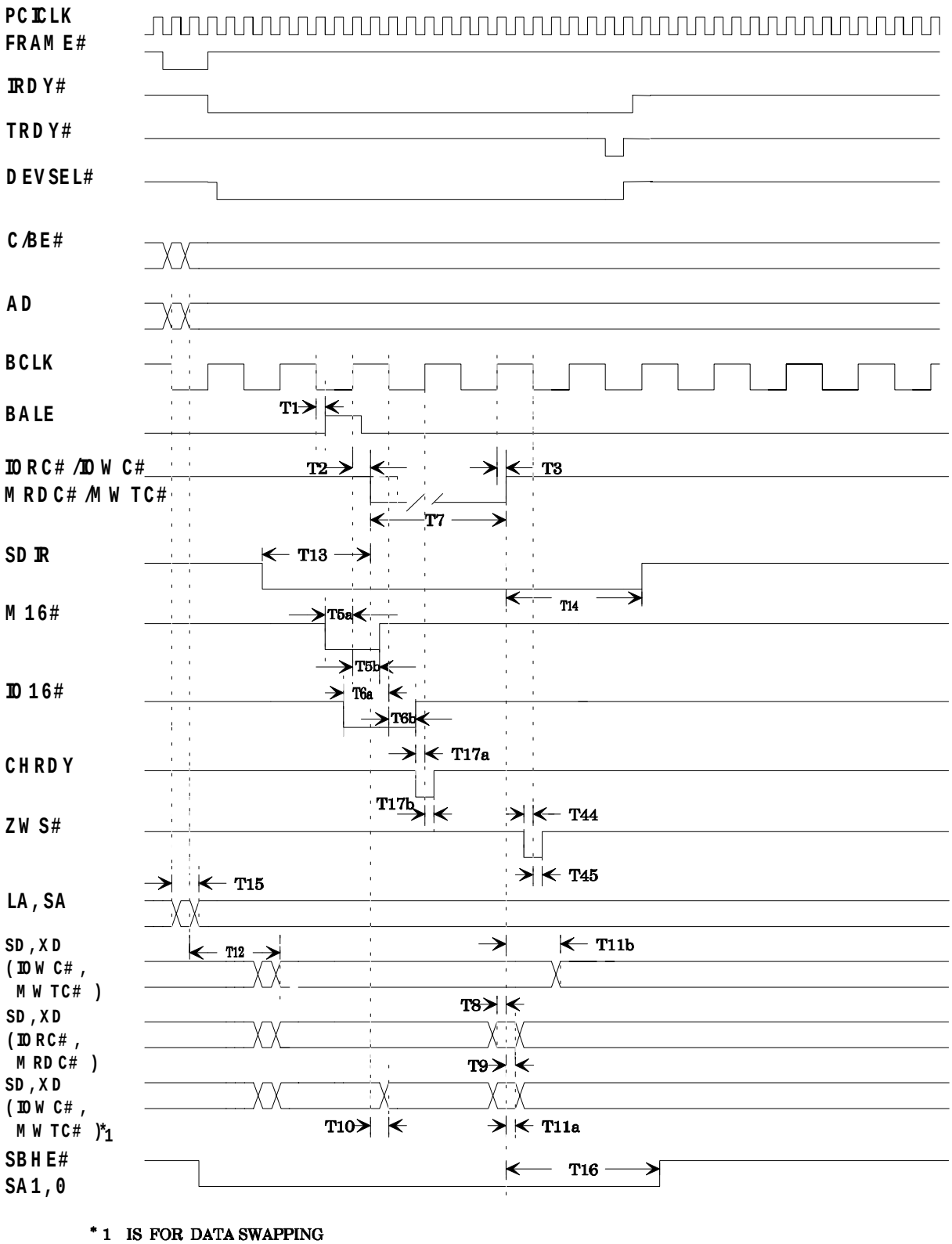
***1: No command delay**

PCI Bus AC Specifications (Figure 4.10) 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.**

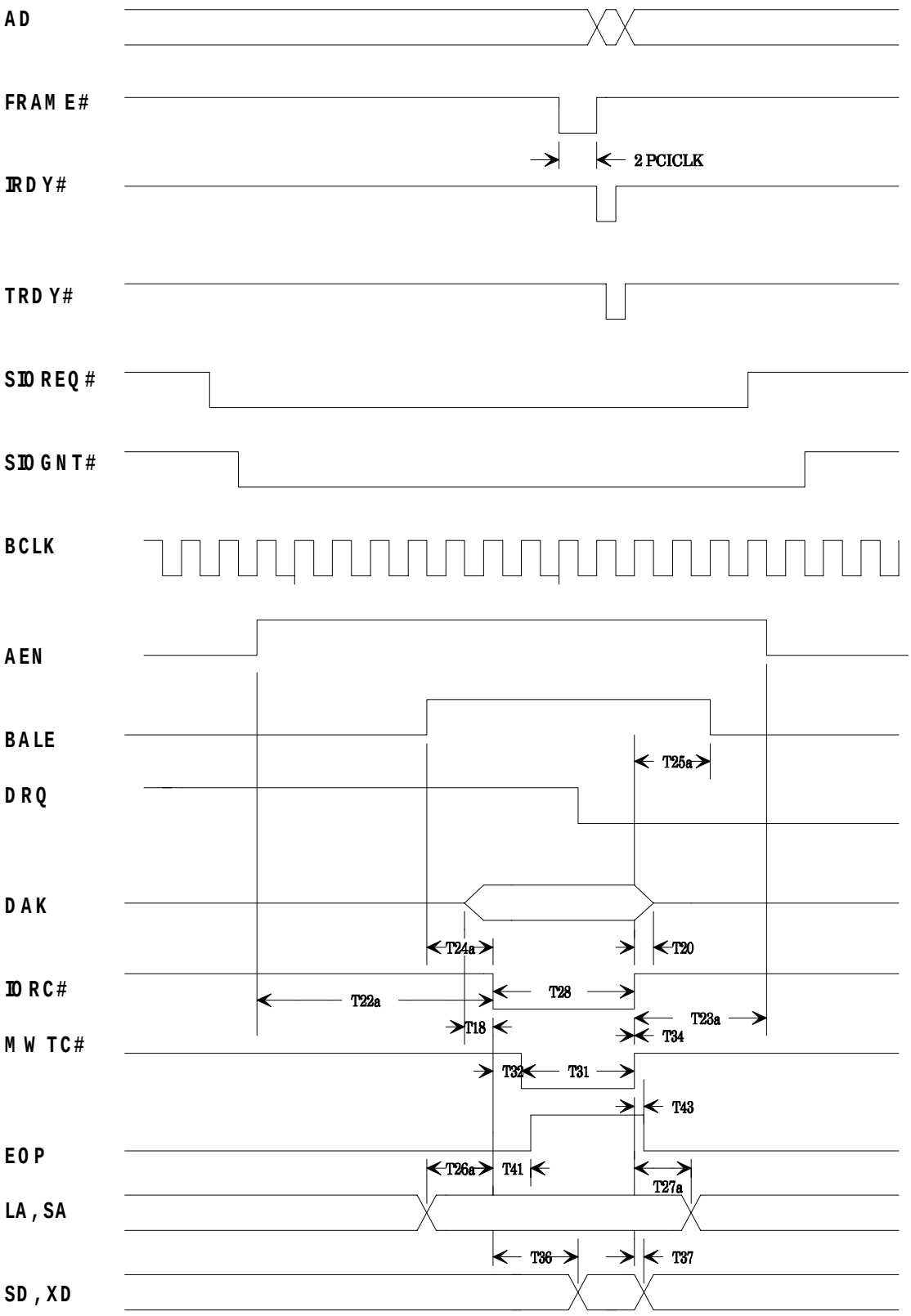
PCI IDE Timing (Figure 4.14 ~ 4.17)					
Para-meter	Description	Min	Typ	Max	Unit
t63	Read Active Time	3		8	PCICLK
t64	Read Recovery Time	3		13	PCICLK
t65	Write Active Time	4		8	PCICLK
t66	Write Recovery Time	5		17	PCICLK
t67	Read Cycle Tme (Prefetch Buffer Enable)	4	5		PCICLK
t68	Write Cycle Time (Post Write Buffer Enable)		6		PCICLK

4.9.4 AC Timing Diagram



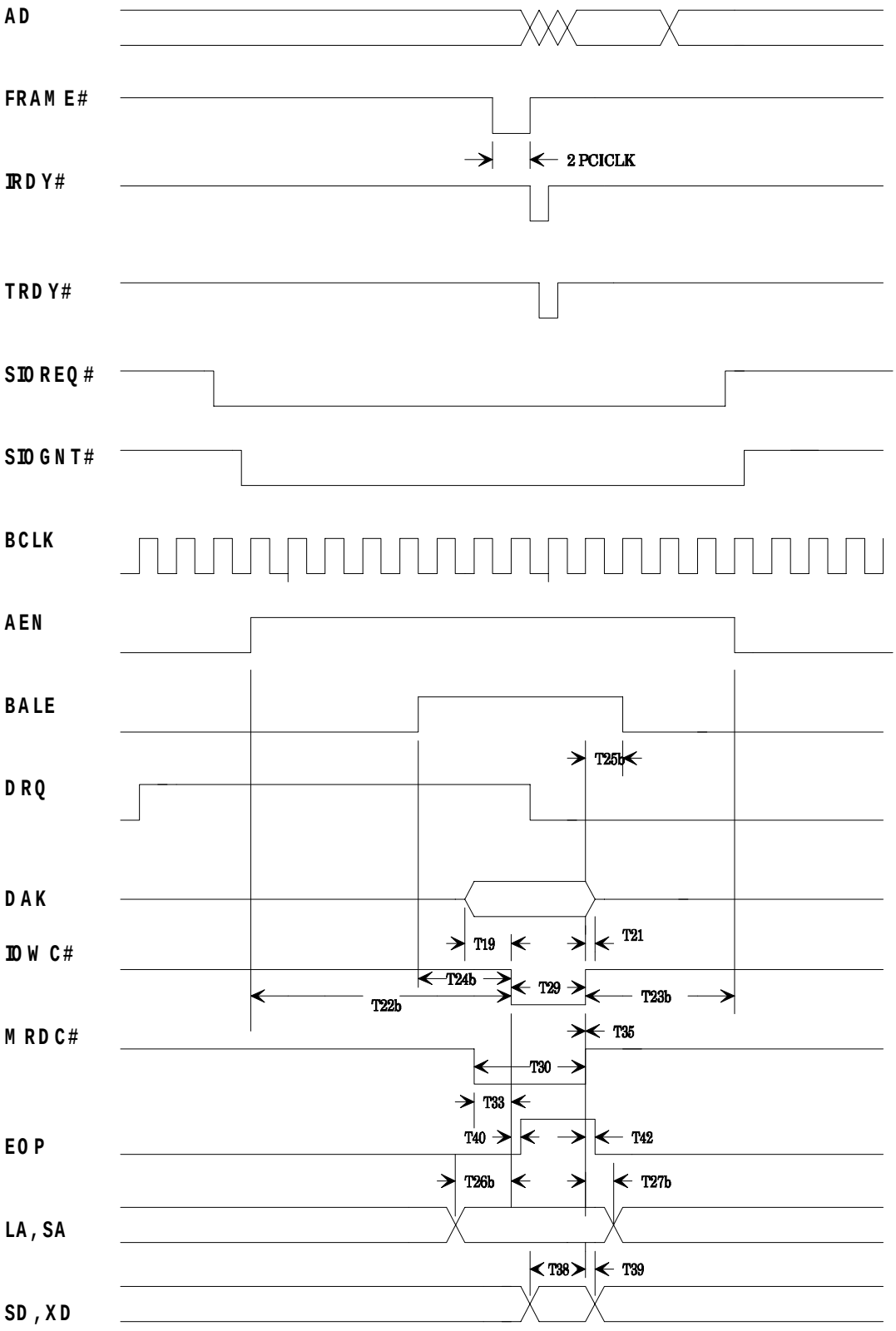
*1 IS FOR DATA SWAPPING

Figure 4.2 PCI to AT Bus Cycle



DMA cycle (IOWC#, MRDC#), DMACLK = BCLK

Figure 4.3 DMA Cycle (IOWC#, MRDC#)



DMA cycle (IORC#, MWTC#), DMACLK = BCLK

Figure 4.4 DMA Cycle (IORC#, MWTC#)

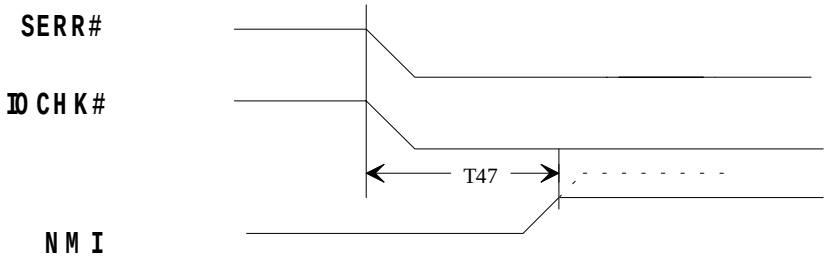


Figure 4.5

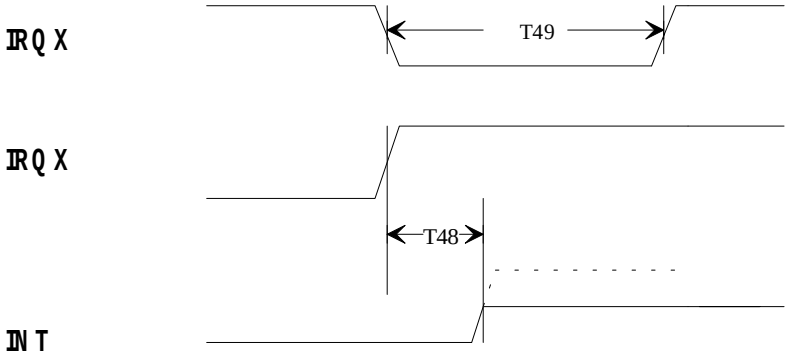


Figure 4.6

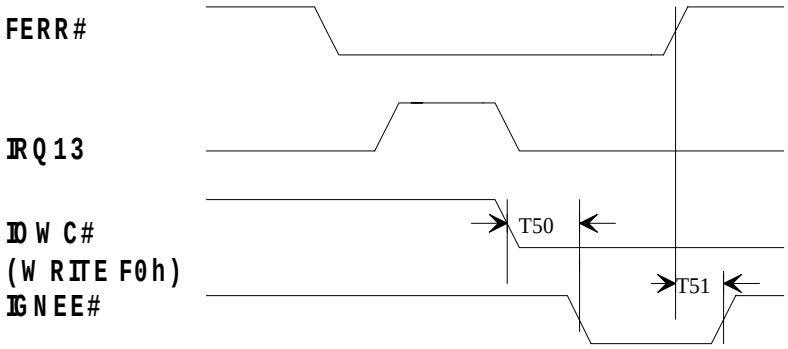


Figure 4.7

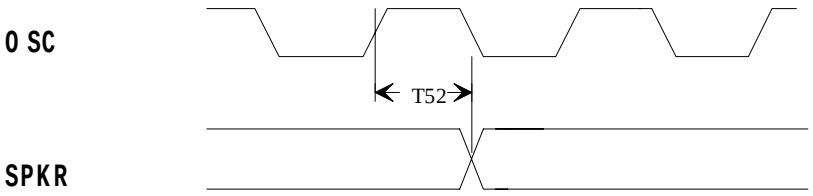


Figure 4.8

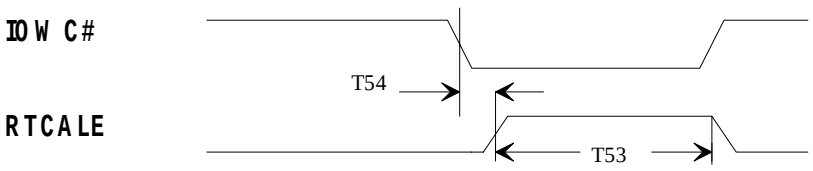


Figure 4.9

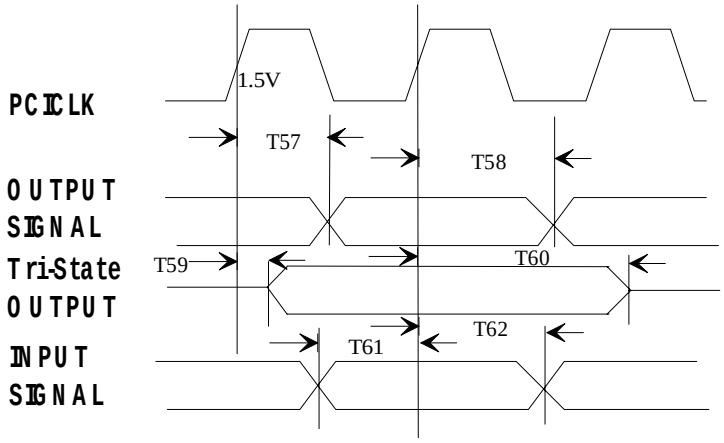


Figure 4.10

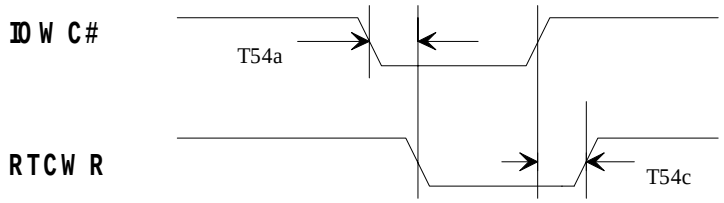


Figure 4.11

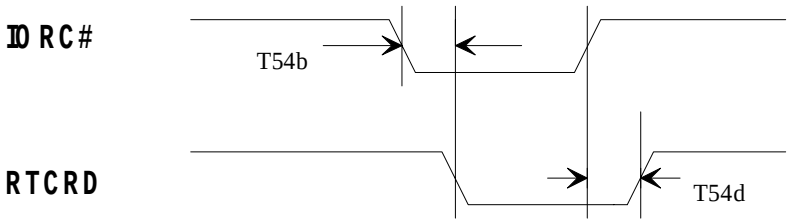


Figure 4.12

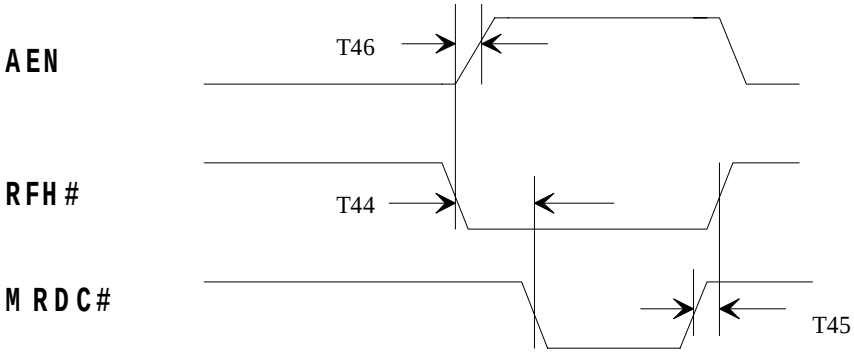


Figure 4.13

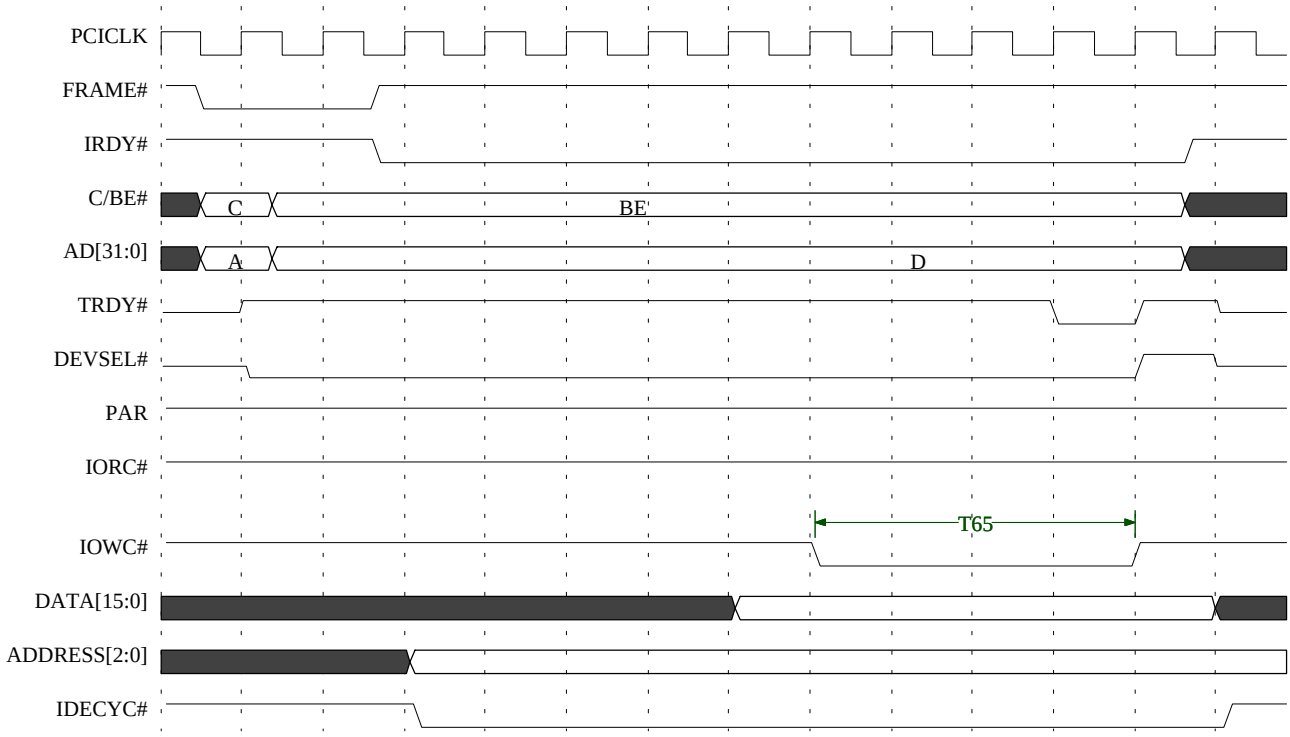


Figure 4.14 IDE Basic Write Cycle (One Word)

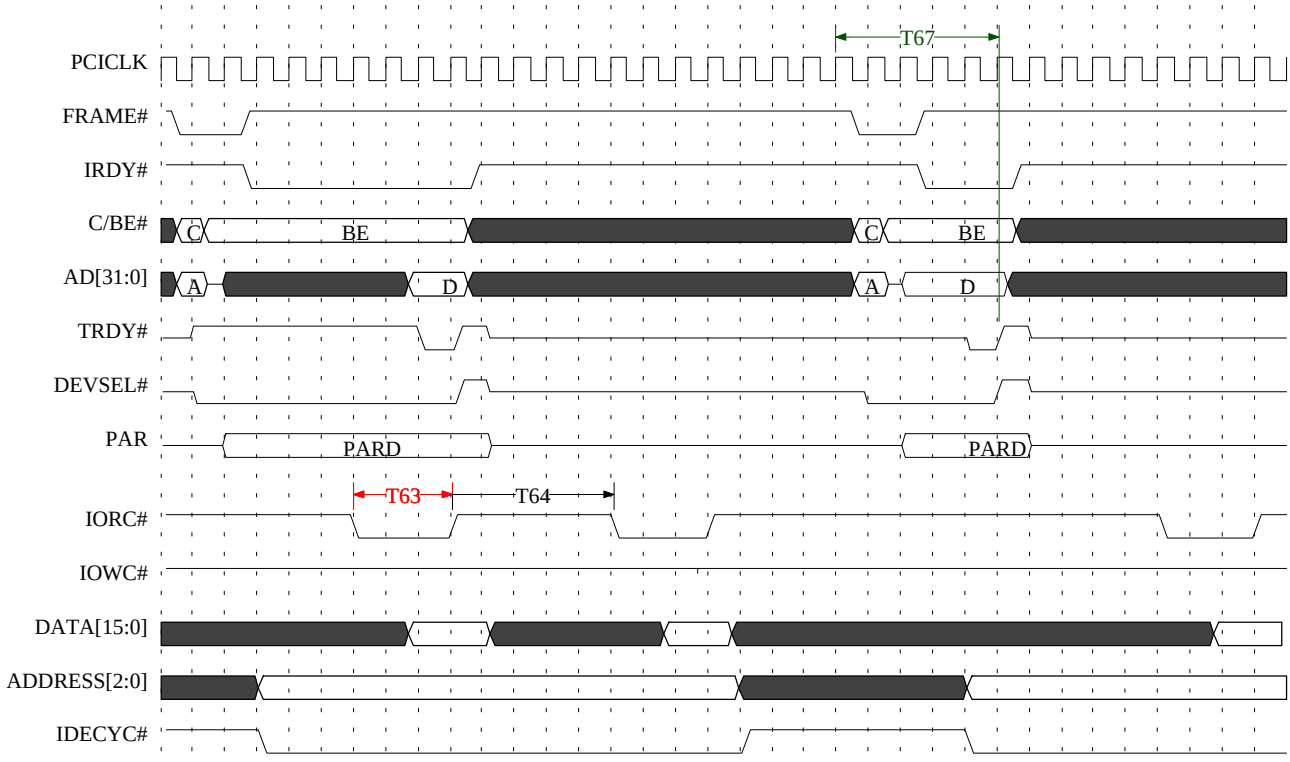


Figure 4.15 IDE Read-Prefetch Cycle (One Word)

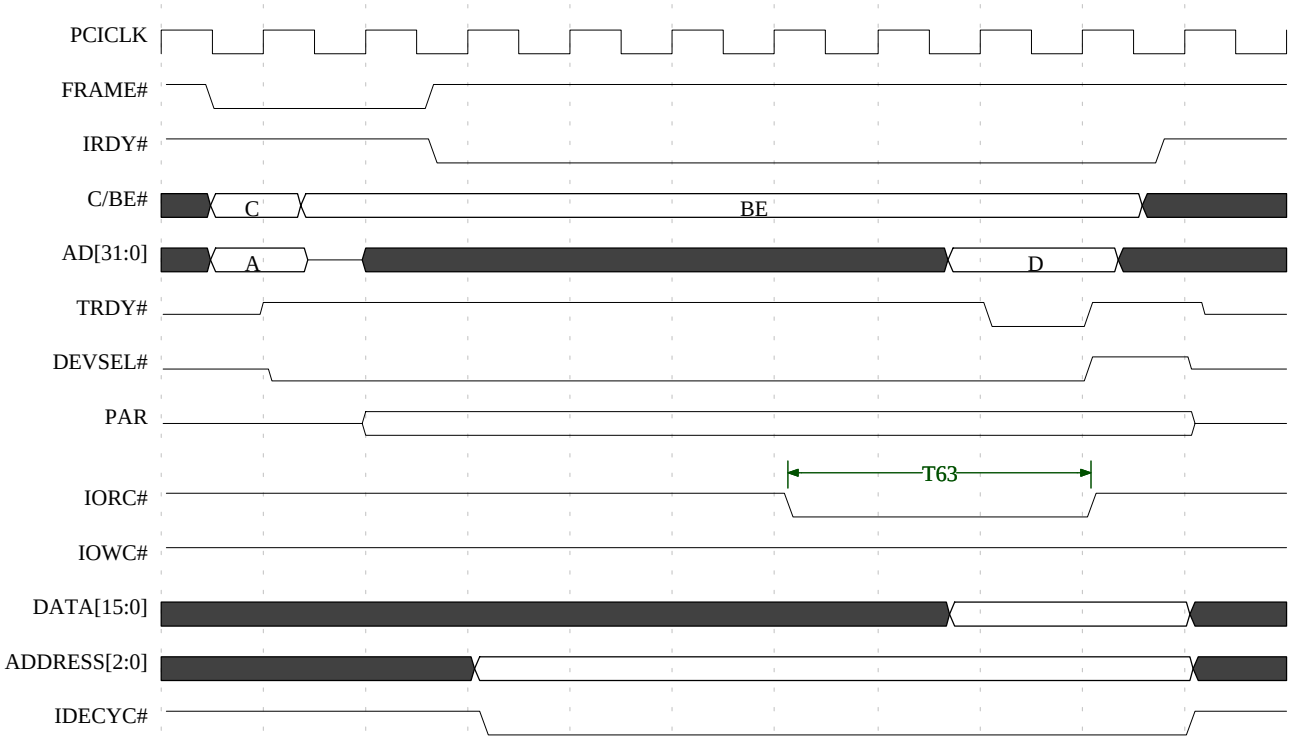


Figure 4.16 IDE Basic Read Cycle (One Word)

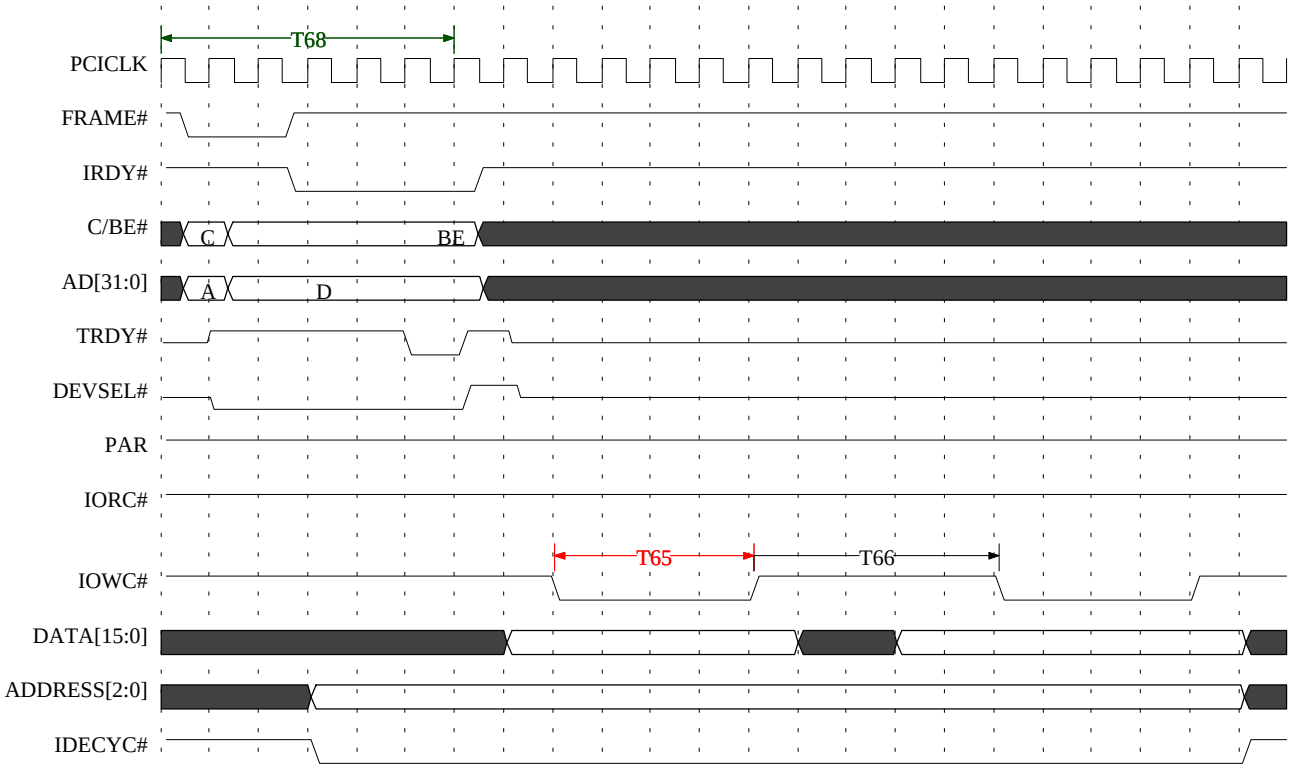
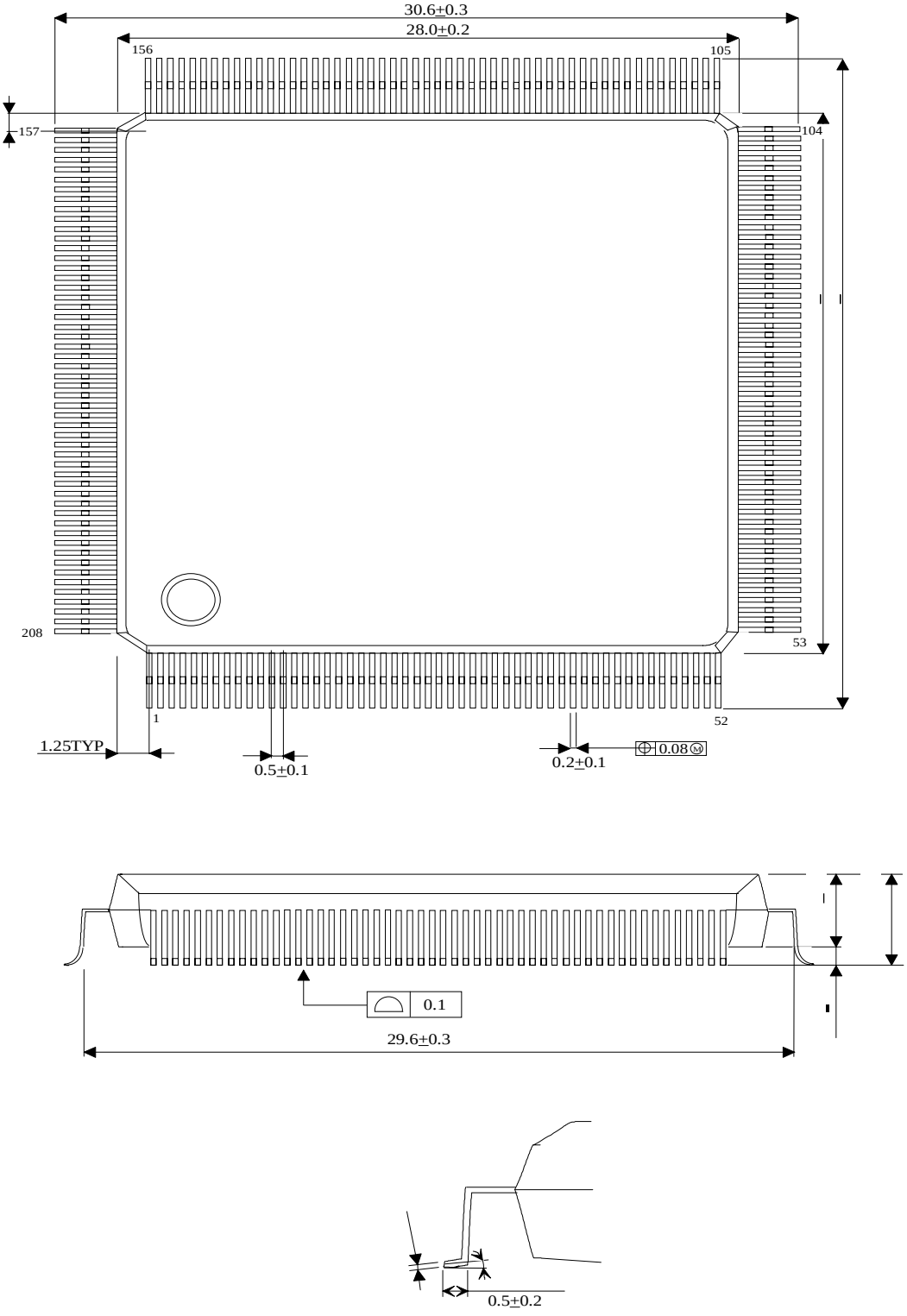


Figure 4.17 IDE Posted-Write Cycle (Double Words)

5. Mechanical Dimension

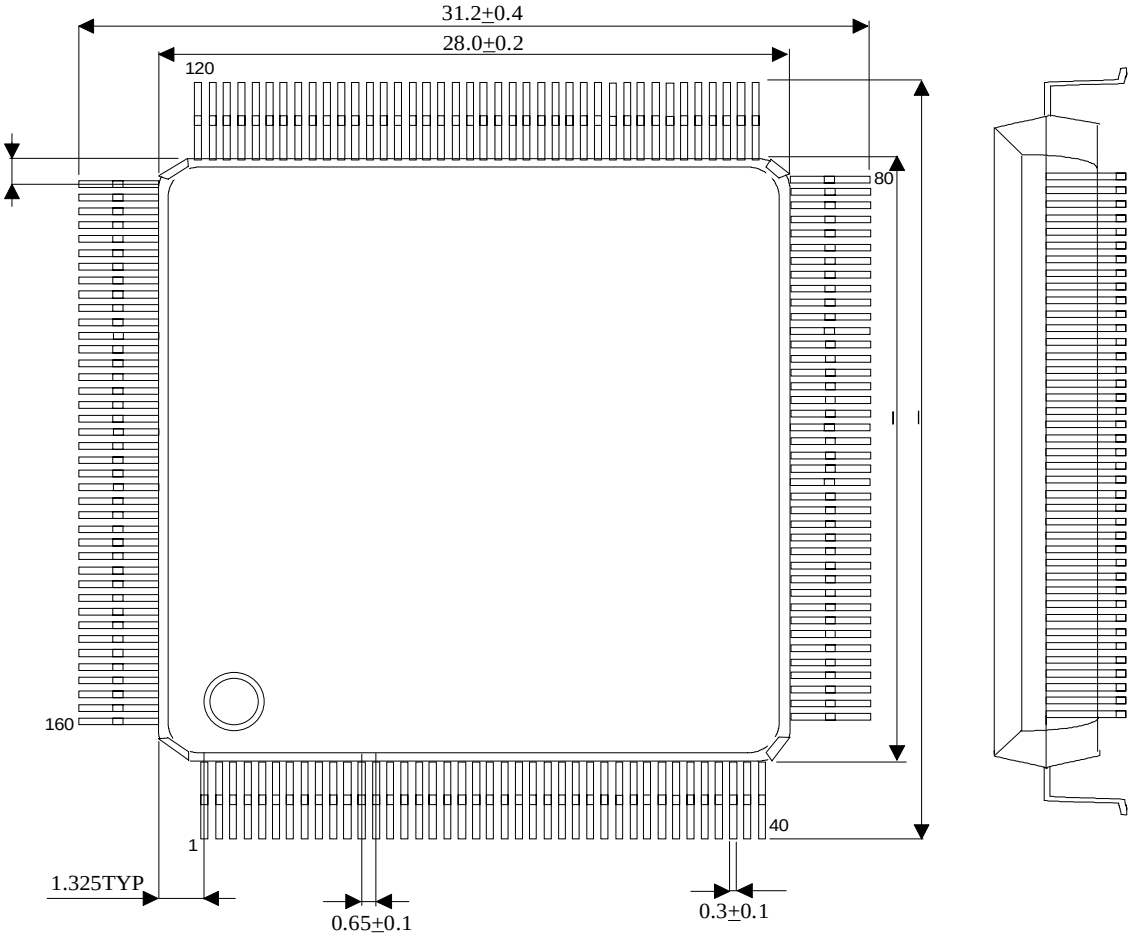
5.1 SiS5501, SiS5502 (208 pins)

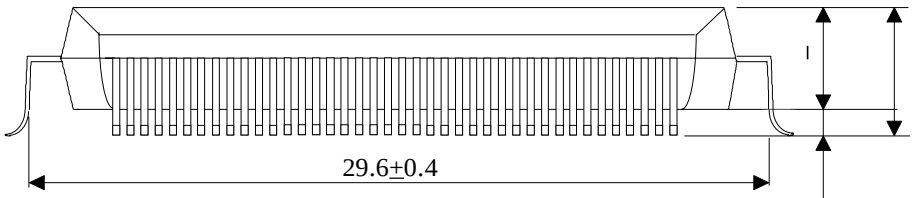
QFP208-P (208-Pin Plastic Flat Package) Unit:mm

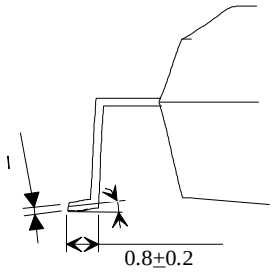


5.2 SiS5503 (160 pins)

QFP160-P (160-Pin Plastic Flat Package) Unit:mm







COPYRIGHT NOTICE

`

TRADEMARKS

DISCLAIMER

RESTRICTED RIGHTS LEGEND