System Engineering Notes
No. 864
Differences Between the SYM53C810, SYM53C810A, and SYM53C860
Rev 2.0, May 1997

The SYM53C810A is a drop-in enhancement of the SYM53C810, although some software changes may be required to take advantage of the enhanced performance of the SYM53C810A. The SYM53C860 is functionally identical to the SYM53C810A, with the added capability for Fast-20 SCSI transfers. This engineering note describes the functional differences, register/bit differences, programming differences, and AC specification differences in all three products.

SUMMARY OF CONTENTS

FUNCTIONAL DIFFERENCES

This section describes the differences in functionality between the SYM53C810 and the SYM53C810A/SYM53C860. Many of these features are selected by setting or clearing register bits; detailed information on these register bits in the sections "Operating Register/Bit Differences" on page 5 and "PCI Configuration Register Differences" on page 6.

PCI Cache Line Size Register

The SYM53C810A/SYM53C860 supports the PCI specification for an 8-bit Cache Line Size register located in PCI configuration space; it can sense the react to non-aligned addresses corresponding to cache line boundaries. The chip balances the needs of both the hardware designer and the software designer by selecting the cache line size to be the smaller of the software controlled burst size set in the DMODE register, or the value in the PCI Cache Line Size register. If caching is not needed, clear the Cache Line Size register to disable PCI cache mode. To use the Cache Line Size register, the SYM53C810A/SYM53C860 supports the following commands: Memory Write and Invalidate, Memory Read Line, and Memory Read Multiple. For more information on these commands, see the section titled "PCI Cache Mode" on page 9.

Memory Write and Invalidate PCI Access Cycles

This mode causes the SYM53C810A/SYM53C860 to issue the PCI Write and Invalidate command instead of a Memory Write command when certain conditions have been met (for more information on these conditions, refer to the section "Programming Differences" on page 7). It is used in conjunction with the Cache Line Size mode and the Cache Line Size register, and will only operate if the Cache Line Size Enable (CLSE) bit has been set.

Memory Read Line PCI Access Cycles

The PCI Read Line mode function in previous members of the SYM53C8XX family has been modified in the SYM53C810A/SYM53C860 to reflect PCI cache line size register specifications. See the section "Programming Differences" on page 7 for more information on the use of this command.

Memory Read Multiple PCI Access Cycles

The Read Multiple command reads in multiple data cache lines in a single PCI bus ownership. This reduces PCI bus overhead. See the section "Programming Differences" on page 7 for more information on the use of this command.

SCRIPTS Instruction Prefetch

The SYM53C810A/SYM53C860 pre-fetch logic, when enabled, will fetch up to 8 double words (dwords) of SCRIPTS instructions in bursts of 4 or 8 dwords. Pre-fetching instructions allows the SYM53C810A/SYM53C860 chip to make more efficient use of the system PCI bus. This improves overall system performance.

SCSI Selected As ID bits

These new bits in the STEST0 register contain the read-only, encoded SCSI ID value that the SYM53C810A/SYM53C860 was selected or reselected as during the SCSI Selection or Reselection phase.

Selectable IRQ Disable

The SYM53C810A/SYM53C860 chip can disable the IRQ pin with only one bit. When the IRQ Disable bit is set, the device will not assert the IRQ pin when an interrupt condition occurs. The interrupt is not lost or ignored, but merely masked at the pin. Clearing the control bit when an interrupt is pending will immediately cause the IRQ pin to assert.

3.3 V/5 PCI interface

The SYM53C810A/SYM53C860 can attach directly to a 3.3 Volt or a 5Volt PCI interface, due to separate V_{DD} pins for the PCI bus drivers. This allows the devices to be used on the “universal board” recommended by the PCI Special Interest Group.

Load and Store SCRIPTS Instruction

The Load and Store instruction is an efficient way to move data to/from memory from/to internal register without using the Move Memory instruction. It also enables the SYM53C810A/SYM53C860 to transfer bytes to addresses relative to the DSA register.

Fast-20 SCSI Transfers (SYM53C860 only)

The SYM53C860 can perform 8-bit synchronous SCSI transfers as fast as 20 MB/s. The SYM53C860 is pin-compatible with the SYM53C810A. However, the SCSI clock (SCLK) input frequency must be increased to support Fast-20 synchronous transfers. The SYM53C860 is software compatible with the SYM53C810A, but software changes are required to enable Fast-20 synchronous negotiation. The required software changes are discussed in section “Migrating Existing Software to Support Fast-20 Transfer Rates (SYM53C860 Only)” on page 11.

If the SYM53C860 is used with Symbios Logic SDMS software, an 80 MHz clock is required in all designs; Symbios Logic recommends using an 80MHz clock even in systems that do not use SDMS. By choosing the appropriate values for the clock divider bits in the SCNTL3 register, the SYM53C860 with an 80 MHz clock can still transfer SCSI data at the Fast SCSI-2 transfer of the SYM53C810.

Pinout Differences

Figure 1 is the pin diagram for the SYM53C810A and SYM53C860. Both chips are drop-in replacements for the SYM53C810.

PCI V_{DD} Pins

The V_{DD}-I pins on the SYM53C810A/SYM53C860 are power supplies to the PCI I/O pins. These separate pins can accept a V_{DD} source of 3.3 or 5 Volts. All other V_{DD} pins must be supplied 5 V. The pins used for the PCI bus 3.3/5 Volt operation are 3, 16, 28, 40, and 90

SCLK Frequency for Fast-20 Operation (SYM53C860 only)

Since the frequency of a Fast-20 bus is twice that of a Fast SCSI-2 bus, the input clock frequency on the SCLK pin must also be doubled to 80 MHz. The SYM53C860 requires an 80 MHz clock to reach full Fast-20 speeds. To migrate from a Fast SCSI-2 system with a 40 MHz clock, be sure to divide the clock by a factor of two more to achieve the same synchronous transfer rate in a system with an 80 MHz clock.

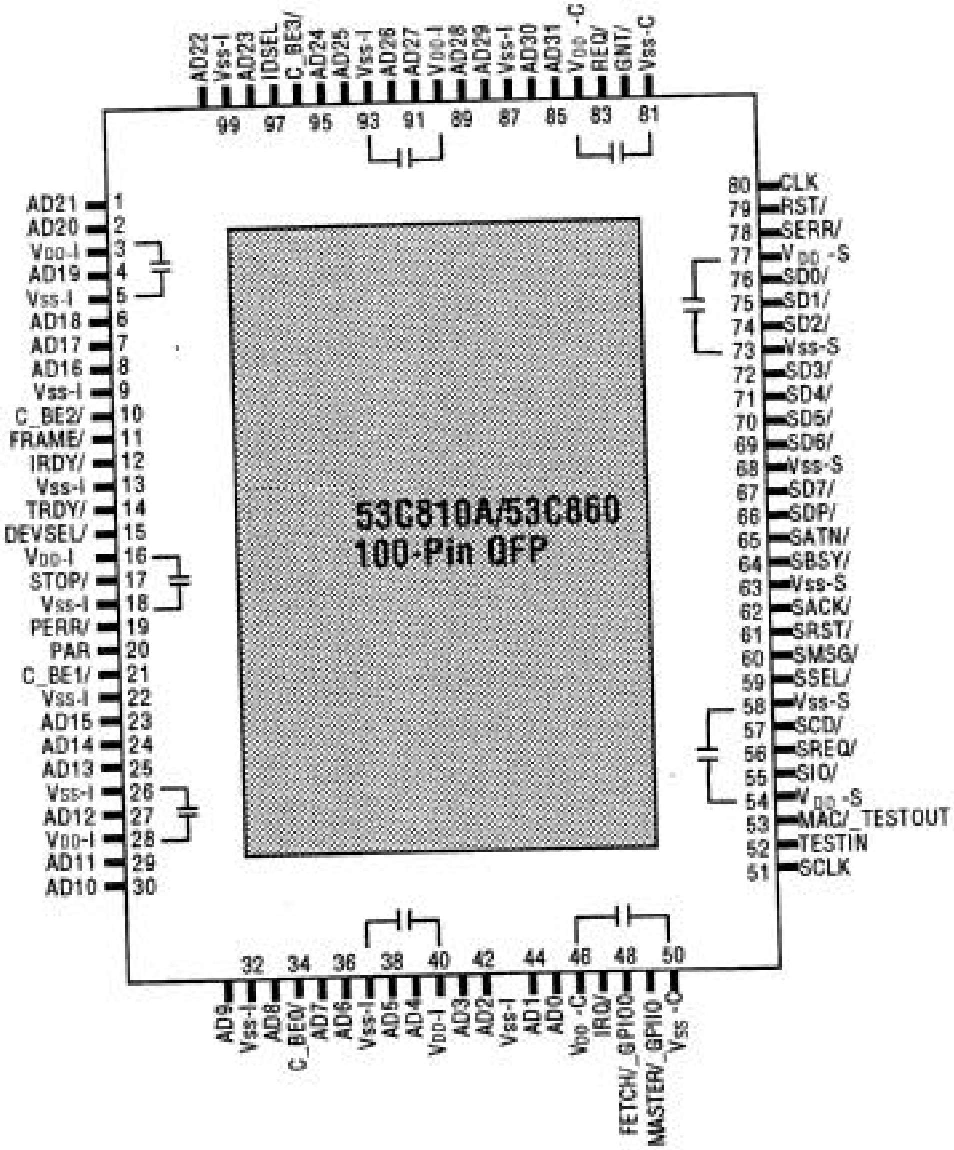

Figure 1. SYM53C810A/SYM53C860 Pin Diagram

OPERATING REGISTER/BIT DIFFERENCES

The register differences between the SYM53C810A/SYM53C860 and the SYM53C810 are related to the new features in the SYM53C810A/SYM53C860.

SCNTL 3 (03h) (SYM53C860 only)

Bit 7, Fast-20 Enable. Setting this bit enables Fast-20 transfers in systems that have an 80 MHz clock. The default value of this bit is 0. This bit should remain cleared in systems that have a 40 MHz clock.

Bits 6-4 and 2-0, SCSI Clock Divider bits. These bits select a factor by which the frequency of SCLK is divided before being presented to the SCSI transfer control logic. The two sets of bits should be written to the same value for normal SCSI transfers. The synchronous portion of the SCSI core can be run at a fast clock rate for fast SCSI, using bits 6-4. Table 1 shows the supported bit settings.

Table 1: SCSI Clock Conversion Factors

SCF2 CCF2	SCF1 CCF1	SCF0 CCF0	Factor Frequency	SCSI Clock (MHz)
0	0	0	SCLK/3	50.01-66
0	0	1	SCLK/1	16.67-25
0	1	0	SCLK/1.5	25.01-37.5
0	1	1	SCLK/2	37.51-50
1	0	0	SCLK/3	50.01-75
1	0	1	SCLK/4	75-80
1	1	0	Reserved	
1	1	1	Reserved	

CTEST3 (0Bh)

Bit 0, Write and Invalidate Enable (WRIE). This bit, when set, will cause Memory Write and Invalidate commands to be issued on the PCI bus after certain conditions have been met. These conditions are listed under the heading “Memory Write and Invalidate Command” on page 9.

DMODE Register (38h)

Bit 3, Enable Read Line. This bit enables Read Line mode. If PCI cache mode is enabled, a Read Line command is issued on all read cycles if other conditions are met. These conditions are listed under the heading “Memory Read Line” on page 10.

Bit 2, Enable Read Multiple (ERMP). This bit, when set, will cause Read Multiple commands to be issued on the PCI bus after certain conditions have been met. These conditions are listed under the heading “Memory Read Multiple” on page 11.

DCNTL (3Bh)

Bit 7, Cache Line Size Enable (CLSE). Setting this bit enables the device to sense and react to cache line boundaries set up by the DMODE or Cache Line Size register, whichever contains the smaller value. Clearing this bit disables the cache line size logic and the SYM53C810A/SYM53C860 monitors the cache line size via the DMODE register.

Bit 6, Pre-Fetch Flush (PFF). Setting this bit will cause the pre-fetch unit to flush its contents. The bit will reset after the flush is complete.

Bit 5, Pre-fetch Enable Bit (PFEN). Setting this bit enables the pre-fetch unit if the burst size is greater than or equal to four.

Bit 1, IRQ Disable. Setting this bit disables the IRQ pin; clearing the bit enables normal operation. As with any register other than the ISTAT, this register can not be accessed except by a SCRIPT instruction during SCRIPTS execution.

STEST0 REGISTER (4Ch)

Bits 6-4, SCSI Selected As ID (SSAID). This SSAID bits contain the encoded value of the SCSI ID that the SYM53C810A/SYM53C860 was selected or reselected as during a SCSI selection or reselection phase. These bits are read only and contain 3 bits which encode the possible 0-7 IDs that could be used to select the SYM53C810A/SYM53C860. This register works in conjunction with the RESPID register, which contains the allowable response IDs of the SYM53C810A/SYM53C860. During a SCSI selection or reselection phase when a valid ID has been put on the bus, and the SYM53C810A/SYM53C860 responds to that ID, the “selected as” ID will be written into the SSAID field.

Bit 7 of this register is reserved and must not be set.

PCI CONFIGURATION REGISTER DIFFERENCES

Cache Line Size Register

The SYM53C810A/SYM53C860 has a Cache Line Size register in PCI configuration space. The value of this register corresponds to the cache subsystem’s cache line size.

Configuration Command Register (04h)

Bit 4, Write and Invalidate Mode. This bit, when set, will cause Memory Write and Invalidate cycles to be issued on the PCI bus after certain conditions have been met. For more information on these conditions, refer to the section “Memory Write and Invalidate Command” on page 9. To enable Write and Invalidate Mode, bit 0 in the CTEST3 register (SYM53C810A/SYM53C860 Operating Registers) must also be set.

PROGRAMMING DIFFERENCES

Transparent Instruction Prefetch

When enabled (by setting the Prefetch Enable bit in the DCNTL register), the prefetch logic in the SYM53C810A/SYM53C860 fetches 4 or 8 dwords of instructions. The prefetch logic automatically determines the maximum burst size that it can perform, based on the burst length as determined by the values in the DMODE register of the Cache Line Size register (if cache mode is enabled). If the unit cannot perform bursts of at least four dwords, it will disable itself.

The SYM53C810A/SYM53C860 may flush the contents of the prefetch unit under certain conditions, listed below, to ensure that the chip always operates from the most current version of the software. When one of these conditions apply, the contents of the prefetch unit are flushed automatically.

1. On every Memory Move instruction. The Memory Move (MMOV) instruction is often used to place modified code directly into memory. To make sure that the chip executes all recent modifications, the prefetch unit flushes its contents and loads the modified code every time a MMOV instruction is issued. To avoid inadvertently flushing the prefetch unit contents, use the new No Flush Memory to Memory Move instruction for all MMOV operations that do not modify code within the next 4 to 8 dwords.
2. On every write to the DSP.
3. On all Transfer Control instructions when the transfer conditions are met. This is necessary because the next instruction to be executed is not the sequential next instruction in the prefetch unit.
4. When the Pre-Fetch Flush bit (DCNTL bit 5) is set. The unit flushes whenever this bit is set. The bit is self-clearing.

No Flush Memory to Memory Move Instruction

This new instruction performs a Memory Move (MMOV) without flushing the prefetch unit. The current Memory Move instruction automatically flushes the FIFO. NFMMOV should be used if the source and destination are not within four instructions of the current MMOV instruction.

No Flush Memory Move Instruction Syntax

MOVE MEMORY NOFLUSH count, source_address, destination_address

Operands: *count* is a 24-bit expression which indicates the number of bytes to transfer

source_address is the absolute 32-bit staring address of data in memory

destination_address is the absolute 32-bit starting address of data in memory

Example: *MOVE MEMORY NOFLUSH 0x06, inquiry_address, cmd_address*

Load and Store Instruction

The Load and Store instruction provides a more efficient way to move data from/to memory to/from an internal register in the chip without using the normal memory move instruction.

The load and store instructions are represented by two-dword op codes. The first dword contains the DCMD and DCB register values. The second dword contains the DSPS value. This is either the actual memory location or where to load or store, or the offset from the DSA, depending on the value of Bit 28 (DSA Relative).

A maximum of 4 bytes may be moved with these instructions. The register address and memory address must have the same byte alignment, and the count set such that it does not cross dword boundaries. The memory address may not map back to the chip. If it does, a PCI illegal read/write cycle will occur, and the chip will issue an interrupt (Illegal Instruction Detected) immediately following.

Bits A1, A0	**Number of bytes allowed to load/store**
00	One, two, three or four
01	One, two, or three
10	One or two
11	One

The SIOM and DIOM bits in the DMODE register determine whether the destination or source address of the instruction is in Memory space or I/O space. The Load/Store utilizes the PCI commands for I/O READ and I/O WRITE to access the I/O space.

Bit Encoding of the Load and Store Instruction

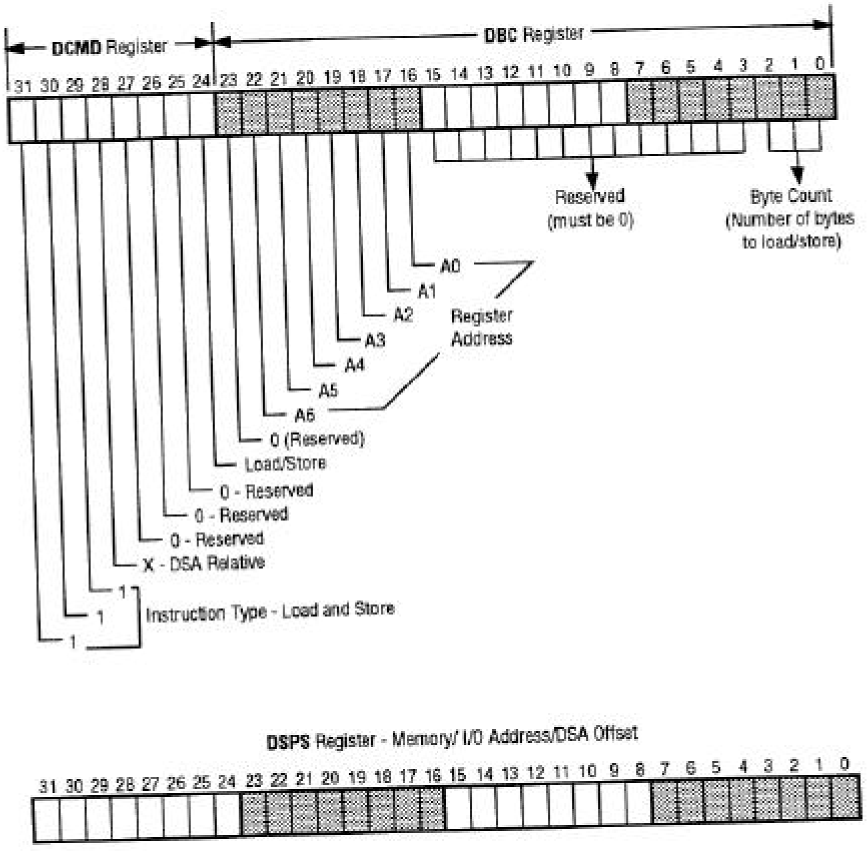

Figure 2. Load and Store Instruction

First Dword:

Bit 31-29, Instruction Type. These bits should be 111, indicating the Load and Store instruction.

Bit 28, DSA Relative. When this bit is clear, the value in the DSPS is the actual 32-bit memory address to perform load/store to/from. When this bit is set, the chip determines the memory address to perform the load/store to/from by adding the 24-bit signed offset value in the DSPS to the DSA.

Bits 27-25, Reserved.

Bit 24, Load/Store. When this bit is set, the instruction is a load. When cleared, it is a Store.

Bit 23, Reserved.

Bits 22-16, Register Address. A6-A0 select the register to load/store to/from within the SYM53C810A/SYM53C860.

Bits 15-3, Reserved.

Bits 2-0, Byte Count. This value is the number of bytes to load/store.

Second Dword:

Bits 31-0, Memory/IO Address / DSA Offset.

Load Instruction Syntax

LOAD [FROM] register, byte_count, source_address

Operands: *FROM* indicates that the source_address is an offset and should be added to the DSA register to obtain the physical address of the destination (DSA relative).

register is one of the registers in the SYM53C8XX register set.

byte_count is the number of bytes (1-4) to be transferred from the source_address.

source_address is the physical address, or offset from the DSA to obtain the physical address, of the data to be loaded into the register.

Examples: *LOAD SCRATCHA0, 4, 0xFFFE56*

LOAD FROM SCRATCHA3, 2, 0x02

Store Instruction Syntax

STORE[FROM] register, byte_count, destination_address

Operands: *FROM* indicates that the destination_address is an offset and should be added to the DSA register to obtain the physical address of the destination (DSA relative).

register is one of the registers in the SYM53C8XX register set.

Byte_count is the number of bytes (1-4) to be transferred to the destination_address.

destination_address is the physical address, or offset from the DSA to obtain the physical address, of the destination.

Examples: *STORE SCRATCHA0, 4, 0xFFFE56*

STORE FROM SCRATCHA3, 2, 0x02

PCI CACHE MODE

Enabling Cache Mode

Cache mode is controlled by the Cache Line Size Enable (CLSE) bit, bit 7 in the DCNTL register. Setting this bit causes the SYM53C810A/SYM53C860 to align to cache line boundaries before allowing any bursting, except during MMOVs in which the read an write addresses are Burst Size boundary misaligned.

Selection of Cache Line Size

The cache logic will select a cache line size based on the values for the burst size in the DMODE register and the PCI Cache Line Size register.

Note: The SYM53C810A/SYM53C860 will not automatically use the value in the PCI Cache Line Size register as the cache line size value. The chip scales the value of the Cache Line Size register down to the nearest binary burst size allowed by the chip (2, 4, 8, or 16), compares this value to the DMODE burst size, then selects the smallest as the value for the cache line size. The SYM53C810A/SYM53C860 will use this value for all burst data transfers.

Alignment

The SYM53C810A/SYM53C860 uses the calculated burst size value to monitor the current address for alignment to the cache line size. When it is not aligned the chip disables bursting, allowing only single dword transfers until a cache line boundary is reached. When the chip is aligned, bursting is reenabled it will burst in increments specified by the Cache Line Size register as explained above. If the Cache Line Size register is not set (default = 00h), the DMODE burst size is automatically used as the cache line size.

MMOV Misalignment

The SYM53C810A/SYM53C860 will not operate in a cache alignment mode when a MMOV instruction is issued and the read and write addresses are different distances from the nearest cache line boundary. For example, if the read address is 0x21F and the write address is 0x42F, and the cache line size is eight (8), the addresses are byte aligned, but they are not the same distance from the nearest cache boundary. The read address is 1 byte from the cache boundary 0x220 and the write address is 17 bytes from the cache boundary 0x440. In this situation, the chip will not align to cache boundaries and will operate as a SYM53C810.

MEMORY WRITE AND INVALIDATE COMMAND

The Write and Invalidate mode is enabled by setting bit 0 in the CTEST3 register and Bit 4 of the PCI Configuration Command Register. This will cause Write and Invalidate commands to be issued on the PCI bus when certain conditions have been met. These conditions are:

1. The CLSE and WRIE bits, and PCI Config Command register, bit 4 must be set.
2. The cache line size register must contain a legal burst size (2, 4, 8 or 16) value AND that value must be less than or equal to the DMODE burst size.
3. The chip must have enough bytes in the DMA FIFO to complete a full cache line burst.
4. The chip must be aligned to a cache line boundary.

When these conditions have been met, the SYM53C810A/SYM53C860 will issue a Write and Invalidate command instead of a Memory Write command during all PCI write cycles.

Multiple Cache Transfers

When multiple cache lines of data have been read in during a MMOV instruction (See the description for the Read Multiple command), the SYM53C810A/SYM53C860 will issue a Write and Invalidate command using the burst size necessary to transfer all the data in one transfer. For example, if the cache line size is 4, and the chip read in 16 dwords of data using a Read Multiple command, the chip will switch the burst size to 16, and issue a Write and Invalidate to transfer all 16 dwords in one bus ownership.

Latency

In accordance with the PCI specification, the chip's latency timer will be ignored when issuing a Write and Invalidate command such that when a latency time-out has occurred, the SYM53C810A/SYM53C860 will continue to transfer up until a cache line boundary. At that point, the chip will relinquish the bus, and finish the transfer at a later time using another bus ownership. If the chip is transferring multiple cache lines it will continue to transfer until the next cache boundary is reached.

PCI Target Retries

During a Write and Invalidate transfer, if the target device issues a retry (STOP, TRDY), the SYM53C810A/SYM53C860 will relinquish the bus and immediately try to finish the transfer on another bus ownership. However, the chip will not issue another Write and Invalidate command, rather, it will issue a normal Memory Write. This is in accordance with the PCI specification.

MEMORY READ LINE

The Read Line Mode function that exists in the previous SYM53C8XX chips has been modified in the SYM53C810A/SYM53C860 to reflect the PCI cache line size register specifications. The functionality of the Enable Read Line bit (bit 3 in DMODE) has been modified to more resemble the Write and Invalidate mode in terms of conditions that must be met before a Read Line command will be issued. However, the Read Line option will operate exactly like the previous SYM53C8XX chips when cache mode has been disabled by a CLSE bit reset or when certain conditions exist in the chip (explained below).

The Read Line mode is enabled by setting bit 3 in the DMODE register. If cache mode has been disabled, Read Line commands will be issued on every read data transfer, except op code fetches, as in previous SYM53C8XX chips.

If cache mode has been enabled, a Read Line command will be issued on all read cycles, except op code fetches, when the following conditions have been met:

1. The CLSE bit must be set (cache mode enabled).
2. The cache line size register must contain a legal burst size value (2, 4, 8 or 16) AND that value must be less than or equal to the DMODE burst size.
3. The amount of bytes to be transferred at the time a cache boundary has been reached must be equal to or greater than a full cache line size.
4. The chip must be aligned to a cache line boundary.

When these conditions have been met, the chip will issue a Read Line command instead of a Memory Read during all PCI read cycles. Otherwise, it will issue a normal Memory Read command.

MEMORY READ MULTIPLE

The SYM53C810A/SYM53C860 supports PCI Read Multiple functionality and will issue Read Multiple commands on the PCI bus when the Read Multiple Mode is enabled. This mode is enabled by setting bit 2 of the DMODE register (ERMP). The command will be issued when certain conditions have been met.

If cache mode has been enabled, a Read Multiple command will be issued on all read cycles, except op code fetches, when the following conditions have been met:

1. The CLSE and ERMP bits must be set (cache mode enabled).
2. The Cache Line Size register must contain a legal burst size value (2, 4, 8 or 16) AND that value must be less than or equal to the DMODE burst size.
3. The number of bytes to be transferred at the time a cache boundary has been reached must be equal to or greater than the DMODE burst size.
4. The chip must be aligned to a cache line boundary.

When these conditions have been met, the chip will issue a Read Multiple command instead of a Memory Read during all PCI read cycles.

Burst Size Selection

The Read Multiple command reads in multiple cache lines of data in a single bus ownership. The number of cache lines to be read is determined by the DMODE burst size bits. In other words, the chip will switch its normal operating burst size to reflect the DMODE burst size settings for the Read Multiple command. For example, if the cache line size is 4, and the DMODE burst size is 16, the chip will switch the current burst size from 4 to 16, and issue a Read Multiple. After the transfer, the chip will then switch the burst size back to the normal operating burst size of 4.

Read Multiple with Read Line Enabled

When both the Read Multiple and Read Line modes have been enabled, the Read Line command will not be issued if the above conditions are met. Instead, a Read Multiple command will be issued, even though the conditions for Read Line have been met.

If the Read Multiple mode is enabled and the Read Line mode has been disabled, Read Multiple commands will still be issued if the Read Multiple conditions are met.

MIGRATING EXISTING SOFTWARE TO SUPPORT FAST-20 TRANSFER RATES (SYM53C860 ONLY)

Fast-20 is simply an extension of the current Fast SCSI-2 synchronous transfers. It allows synchronous transfer periods to be negotiated down as low as 50 ns, which is half the 100 ns period allowed by Fast SCSI-2. This will allow a maximum transfer rate of 20 MB/s on a standard 8-bit narrow SCSI bus or 40 MB/s on a 16-bit wide SCSI bus.

To achieve Fast-20 transfer rates, existing software programs must be updated to reflect changes in the following areas of the SYM53C860:

1. SCNTL3 Register CCF bits
2. SCNTL3 Register SCF bits
3. SXFER Register XFERP bits
4. Change Clock Input from 40 MHz to 80 MHz
5. Fast-20 Mode enable bit, SCNTL3 register

Since Fast-20 is just an extension of Fast SCSI-2 synchronous transfers, the SYM53C860 implements it by extending the way synchronous transfers are performed. The SYM53C860 determines synchronous timings with clock dividers that divide down a clock input (usually 40 MHz) to run the SCSI core at the negotiated synchronous transfer rate. For example, to transfer data at 10 MB/s or 10 mega-transfers per second, we need to divide the 40 MHz clock by 4 to run the SCSI core at 10 MHz. The SYM53C860 requires an 80 Mhz clock. From a software point of view, two changes were made in the SYM53C860 to support Fast-20. The first is a new clock divider option that enables the chip to support an 80MHz clock. The second change is the Fast-20 Enable bit, which was added in the SCNTL3 register.

Clock Divider Bits

Two registers are used to divide down the clock. The first is the SCNTL3 register. The CCF bits in this register determine the SCSI core speed used for asynchronous transfers and any other timings (such as selection time-out). These bits are set based on the input clock frequency and do not change. The SCF bits are used to determine the timings for synchronous transfers and can be changed whenever the SYM53C860 connects to a different device on the SCSI bus.

The SCF bits in the SCNTL3 register are used in conjunction with the XFERP bits in the SXFER register to determine the synchronous period. In the above example of 10 MB/s we would have programmed the SCF bits to 001 for a divide by one factor and then programmed the XFERP bits for 000 for a divide by 4 factor. 40 MHz divided by 1 and then divided by 4 is 10 MHz. Other combinations of these two sets of bits can be used to select a variety of synchronous transfer rates. For more information on the bit combinations that are supported, see the clock divider bit descriptions in the section “Operating Register/Bit Differences.”

Fast-20 Enable Bit

The Fast-20 Enable bit adjusts the chip’s timings to be compliant with the Fast-20 proposed standard. It should be set when the synchronous transfer period is less than 100 ns and cleared when it is greater than or equal to 100 ns.

Loading the New Register Values

Since the Fast-20 Mode Enable bit and the clock dividers are in the SCNTL3 and SXFER registers, these registers can be automatically loaded during a selection or reselection by using Table Indirect Addressing. This allows the chips to transparently talk with a combination of Fast-20 and non-Fast-20 devices on the same SCSI bus.

Negotiating Synchronous Transfers

The easiest way to calculate the synchronous transfer period is by multiplying the clock period by the clock divider values, instead of dividing the clock frequency by the clock divider values. For example, a 40 MHz clock is a 25 ns period. (26 ns)*(1)*(4)-100ns, which is the Fast SCSI-2 synchronous transfer period.

If we are running an 80 MHz clock (12.5 ns period) and only negotiated for Fast SCSI-2 rather than Fast-20, the SCF bits would need to be programmed for SCLK/2 and the XFERP bits for 4 which would be (12.5ns)*(2)*(4)-100ns.

The SCSI-2 specification states that synchronous transfer rates must be a multiple of 4 ns. However with an 80 MHz clock, the period must be a multiple of 12.5 ns. Fast-20 is defined to be a 20 mega-transfer per second maximum, which would be a 50 ns period. Since 50 ns is not a multiple of 4, most SCSI devices cannot negotiate for this exact rate. Unless future revisions of the standard make a different recommendation, most devices will probably negotiate for a 48 ns period. The SYM53C860 cannot be programmed for a 48 ns period since it is not a multiple of 12.5 ns, so driver programs should specify a 50 ns period and the chip should negotiate for a 48 ns period. This is acceptable because the SCSI-2 specification allows you to transfer data at a slower rate than what you negotiated for, but not faster.

To program the chip for a full Fast-20 transfer rate of 50 ns using the required 80 MHz clock, program the SCF bits for SCLK/1 and select an XFERP of 4. This comes out to (12.5ns)*(1)*(4)=50 ns.

AC SPECIFICATION DIFFERENCES (SYM53C860 ONLY)

The SYM53C860 complies with all SCSI and Fast-20 timings and other specifications in the draft standards for SCSI-3 and Fast 20. For additional information on timings or other SCSI bus operations of the SYM53C810A/SYM53C860, please refer to these draft standards. To obtain a copy of the SCSI-3 or Fast-20 draft standard, please contact Global Engineering at (800) 854-7179. For additional information about Fast-20 and other SCSI standards, contact the SCSI BBS at (719) 574-0424.