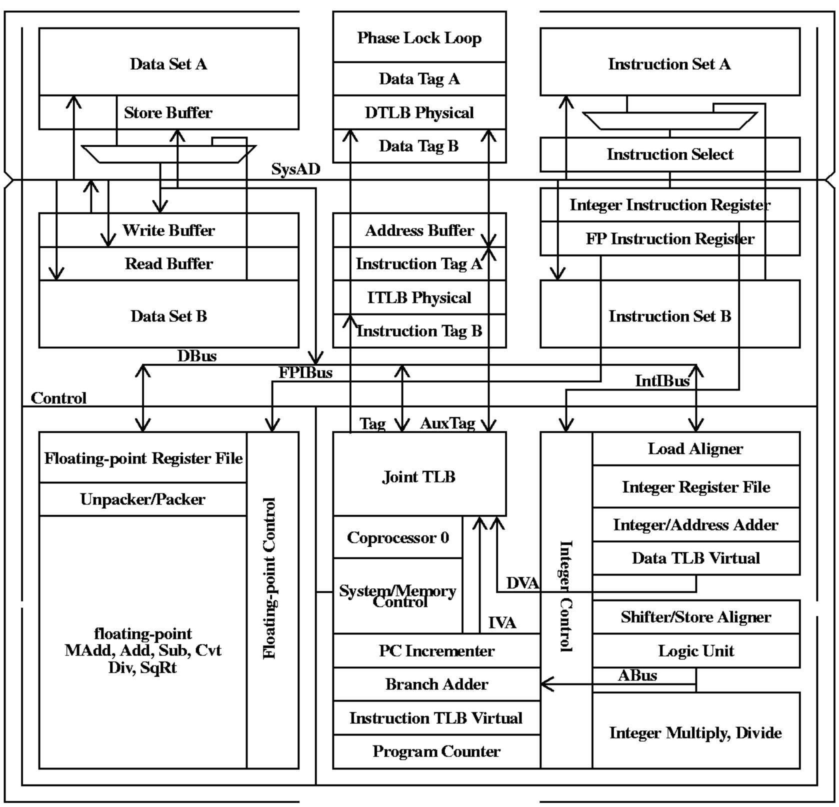
QED RISCMark™ RM5260™

64-Bit Superscalar Microprocessor

FEATURES:

- Dual Issue superscalar microprocessor - can issue one integer and one floating-point instruction per cycle
  - — 133, 150, 175 and 200 MHz operating frequency
  - — 260 Dhrystone2.1 MIPS
  - — SPECInt95 4.8. SPECfp95 5.1
- High-performance system interface compatible with R4600, R4700 and R5000
  - — 64-bit multiplexed system address/data bus for optimum price/performance up to 100 MHz operating frequency
  - — high-performance write protocols maximize uncached write bandwidth
  - — Operates at processor clock multipliers 2 through 8
  - — 5V tolerant I/O's
  - — IEEE 1149.1 JTAG boundary scan
- Integrated on-chip caches - up to 3.2GBps internal data rate
  - — 16KB instruction - 2 way set associative
  - — 16KB data - 2 way set associative
  - — Virtually indexed, physically tagged
  - — Write-back and write-through on per page basis
  - — Pipeline restart on first double for data cache misses
- Integrated memory management unit
  - — Fully associative joint TLB (shared by I and D translations)
  - — 48 dual entries map 96 pages
  - — Variable page size (4KB to 16MB in 4x increments)
- High-performance floating-point unit - up to 400 MFLOPS
  - — Single cycle repeat rate for common single precision operations and some double precision operations
  - — Two cycle repeat rate for double precision multiply and double precision combined multiply-add operations
  - — Single cycle repeat rate for single precision combined multiply-add operation
- MIPS IV instruction set
  - — Floating-point multiply-add instruction increases performance in signal processing and graphics applications
  - — Conditional moves to reduce branch frequency
  - — Index address modes (register + register)
- Embedded application enhancements
  - — Specialized DSP integer Multiply-Accumulate instruction and 3 operand multiply instruction
  - — I and D cache locking by set
  - — Optional dedicated exception vector for interrupts
- Fully static CMOS design with power down logic
  - — Standby reduced power mode with WAIT instruction
  - — 5 Watts typical at 3.3V, less than 70 mWatts in Standby
- 208-pin Power Quad-4 package

BLOCK DIAGRAM:



DESCRIPTION:

The QED RM5260 is a highly integrated superscalar microprocessor that implements a superset of the MIPS IV Instruction Set Architecture (ISA). It has a high-performance 64-bit integer unit, a high-throughput, fully pipelined 64-bit floating-point unit, an operating system friendly memory management unit with a 48-entry fully associative TLB, a 16 KByte 2-way set associative instruction cache, a 16 KByte 2-way set associative data cache, and a high-performance 64-bit system interface. The RM5260 can issue both an integer and a floating-point instruction in the same cycle.

The RM5260 is ideally suited for high-end embedded control applications such as internetworking, high-performance image manipulation, high-speed printing, and 3-D visualization.

HARDWARE OVERVIEW:

The RM5260 offers a high-level of integration targeted at high-performance embedded applications. The key elements of the RM5260 are briefly described below.

Superscalar Dispatch

The RM5260 has an efficient asymmetric superscalar dispatch unit which allows it to issue an integer instruction and a floating-point computation instruction simultaneously. With respect to superscalar issue, integer instructions include alu, branch, load/store, and floating-point load/store, while floating-point computation instructions include floating-point add, subtract, combined multiply-add, converts, etc. In combination with its high-throughput fully pipelined floating-point execution unit, the superscalar capability of the RM5260 provides unparalleled price/performance in computationally intensive embedded applications.

CPU Registers

Like all MIPS ISA processors, the RM5260 CPU has a simple, clean user visible state consisting of 32 general purpose registers, two special purpose registers for integer multiplication and division, a program counter, and no condition code bits. Figure 1 shows the user visible state.

Pipeline

For integer operations, loads, stores, and other non-floating-point operations, the RM5260 uses the simple 5-stage pipeline also found in the R4600, R4700, and R5000 devices. In addition to this standard pipeline, the RM5260 uses an extended 7-stage pipeline for floating-point operations. Like the R5000, the RM5260 does virtual to physical translation in parallel with cache access.

Figure 2 on page 3 shows the RM5260 integer pipeline. As illustrated in the figure, up to five integer instructions can be executing simultaneously.

Integer Unit

Like the R5000, the RM5260 implements the MIPS IV Instruction Set Architecture, and is therefore fully upward compatible with applications that run on processors implementing the earlier generation MIPS I-III instruction sets. Additionally, the RM5260 includes two implementation specific instructions not found in the baseline MIPS IV ISA but that are useful in the embedded market place. Described in detail in a later section, these instructions are integer multiply-accumulate and 3-operand integer multiply.

The RM5260 integer unit includes thirty-two general purpose 64-bit registers, a load/store architecture with single cycle ALU operations (add, sub, logical, shift) and an

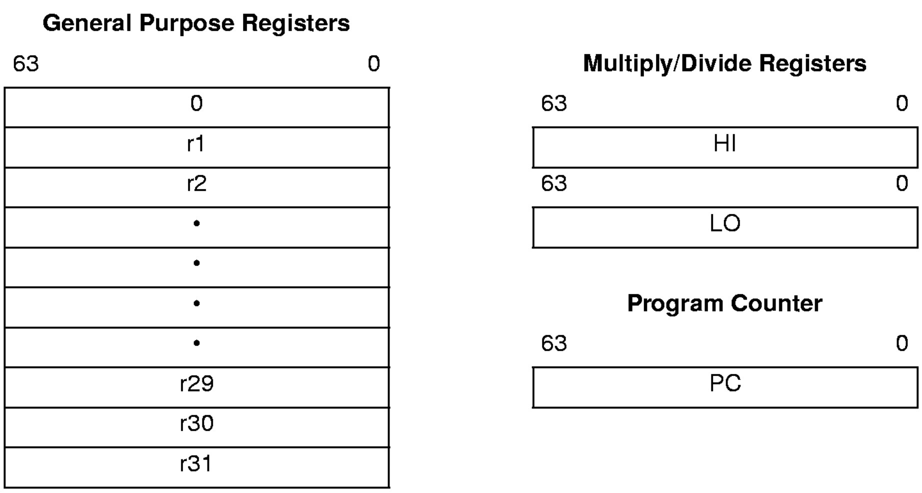
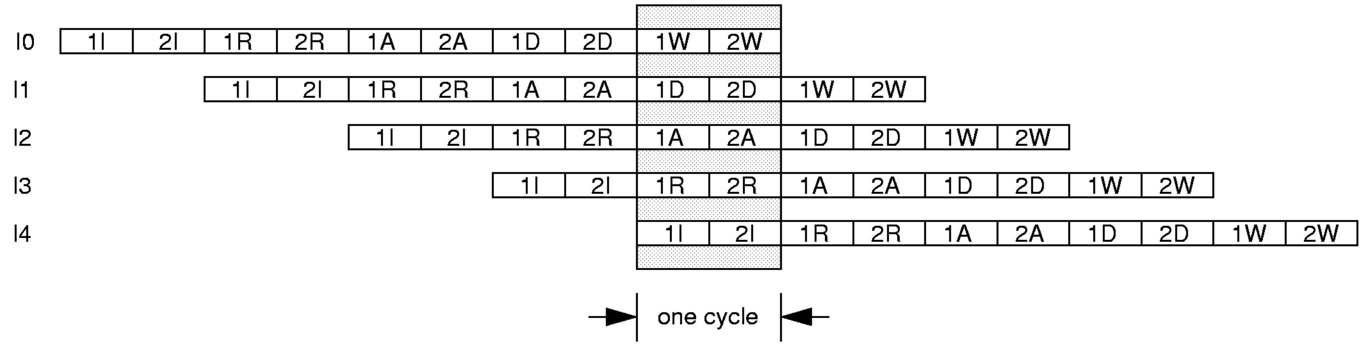


Figure 1 CPU Registers



1I-1R: Instruction cache access
2I: Instruction virtual to physical address translation
2R: Register file read, Bypass calculation, Instruction decode, Branch address calculation
1A: Issue or slip decision, Branch decision
1A: Data virtual address calculation
1A-2A: Integer add, logical, shift
2A: Store Align
2A-2D: Data cache access and load align
1D: Data virtual to physical address translation
2W: Register file write

Figure 2 Pipeline

autonomous multiply/divide unit. Additional register resources include: the *HI/LO* result registers for the two-operand integer multiply/divide operations, and the program counter (*PC*).

Register File

The RM5260 has thirty-two general purpose registers with register location 0 (*r0*) hard-wired to a zero value. These registers are used for scalar integer operations and address calculation. The register file has two read ports and one write port and is fully bypassed to minimize operation latency in the pipeline.

ALU

The RM5260 ALU consists of the integer adder/subtractor, the logic unit, and the shifter. The adder performs address calculations in addition to arithmetic operations, the logic unit performs all logical and zero shift data moves, and the shifter performs shifts and store alignment operations. Each of these units is optimized to perform all operations in a single processor cycle.

Integer Multiply/Divide

The RM5260 has a dedicated integer multiply/divide unit optimized for high-speed multiply and multiply-accumulate operations. Table 1 shows the performance of the multiply/divide unit on each operation.

The baseline MIPS IV ISA specifies that the results of a multiply or divide operation be placed in the *Hi* and *Lo* registers. These values can then be transferred to the general purpose register file using the Move-from-Hi and Move-from-Lo (**MFHI**/**MFLO**) instructions.

In addition to the baseline MIPS IV integer multiply instructions, the RM5260 also implements the multiply instruction, **MUL**, first introduced in the R4650. This instruction specifies that the multiply result go directly to the integer register file rather than the *Lo* register. The portion of the multiply that would have normally gone into the *Hi* register is discarded. For applications where it is known that the upper half of the multiply result is not required, using the **MUL** instruction eliminates the necessity of executing an explicit **MFLO** instruction.

Also included in the RM5260 is the multiply-add instruction, **MAD**, likewise introduced in the R4650. This instruction multiplies two operands and adds the resulting product to the current contents of the *Hi* and *Lo* registers. The multiply-accumulate operation is the core primitive of almost all signal processing algorithms allowing the RM5260 to eliminate the need for a separate DSP engine in many embedded applications.

By pipelining the multiply-accumulate function and dynamically determining the size of the input operands, the

Table 1: Integer Multiply/Divide Operations

Opcode	Operand Size	Latency	Repeat Rate	Stall Cycles
MULT/U, MAD/U	16 bit	3	2	0
	32 bit	4	3	0
MUL	16 bit	3	2	1
	32 bit	4	3	2
DMULT, DMULTU	any	7	6	0
DIV, DIVD	any	36	36	0
DDIV, DDIVU	any	68	68	0

RM5260 is able to maximize throughput while still using an area efficient implementation.

Floating-Point Co-Processor

The RM5260 incorporates a high-performance fully pipelined floating-point co-processor which includes a floating-point register file and autonomous execution units for multiply/add/convert and divide/square root. The floating-point coprocessor is a tightly coupled co-execution unit, decoding and executing instructions in parallel with, and in the case of floating-point loads and stores, in cooperation with the integer unit. As described earlier, the superscalar capabilities of the RM5260 allow floating-point computation instructions to issue concurrently with integer instructions.

Floating-Point Unit

The RM5260 floating-point execution unit supports single and double precision arithmetic, as specified in the IEEE Standard 754. The execution unit is broken into a separate divide/square root unit and a pipelined multiply/add unit. Overlap of divide/square root and multiply/add is supported.

The RM5260 maintains fully precise floating-point exceptions while allowing both overlapped and pipelined operations. Precise exceptions are extremely important in object-oriented programming environments and highly desirable for debugging in any environment.

The floating-point unit's operation set includes floating-point add, subtract, multiply, divide, square root, reciprocal, reciprocal square root, conditional moves, conversion between fixed-point and floating-point format, conversion between floating-point formats, and floating-point compare.

Table 2 gives the latencies of the floating-point instructions in internal processor cycles.

Floating-Point General Register File

The floating-point general register file, FGR, is made up of thirty-two 64-bit registers. With the floating-point load and store double instructions, **LDC1** and **SDC1**, the floating-point unit can take advantage of the 64-bit wide data cache and issue a floating-point co-processor load or store doubleword instruction in every cycle.

The floating-point control register space contains two registers; one for determining configuration and revision information for the coprocessor and one for control and status information. These are primarily used for diagnostic software, exception handling, state saving and restoring, and control of rounding modes. To support superscalar operation, the FGR has four read ports and two write ports, and is fully bypassed to minimize operation latency in the pipeline. Three of the read ports and one write port are used to support the combined multiply-add instruction while the fourth read and second write port allows a concurrent floating-point load or store.

Table 2: Floating-Point Instruction Cycles

Operation	Latency	Repeat Rate
fadd	4	1
fsub	4	1
fmult	4/5	1/2
fmadd	4/5	1/2
fmsub	4/5	1/2
fdiv	21/36	19/34
fsqrt	21/36	19/34
frecip	21/36	19/34
frsqrt	38/68	36/66
fcvt.s.d	4	1
fcvt.s.w	6	3
fcvt.s.l	6	3
fcvt.d.s	4	1
fcvt.d.w	4	1
fcvt.d.l	4	1
fcvt.w.s	4	1
fcvt.w.d	4	1
fcvt.l.s	4	1
fcvt.l.d	4	1
fcmp	1	1
fmov	1	1
fmovc	1	1
fabs	1	1
fneg	1	1

Note: Numbers are represented as single/double precision format.

System Control Co-processor (CP0)

The system control co-processor, co-processor 0 or CP0, in the MIPS architecture is responsible for the virtual memory sub-system, the exception control system, and the diagnostics capability of the processor. In the MIPS architecture, the system control co-processor (and thus the kernel software) is implementation dependent. The RM5260 CP0 is logically identical to that of the R5000.

The memory management unit controls the virtual memory system page mapping. It consists of an instruction address translation buffer, ITLB, a data address translation buffer, DTLB, a Joint instruction and data address translation buffer, JTLB, and co-processor registers used by the virtual memory mapping sub-system.

System Control Co-Processor Registers

The RM5260 incorporates all system control co-processor (CP0) registers on-chip. These registers provide the path through which the virtual memory system's page mapping is examined and modified, exceptions are handled, and operating modes are controlled (kernel vs. user mode,

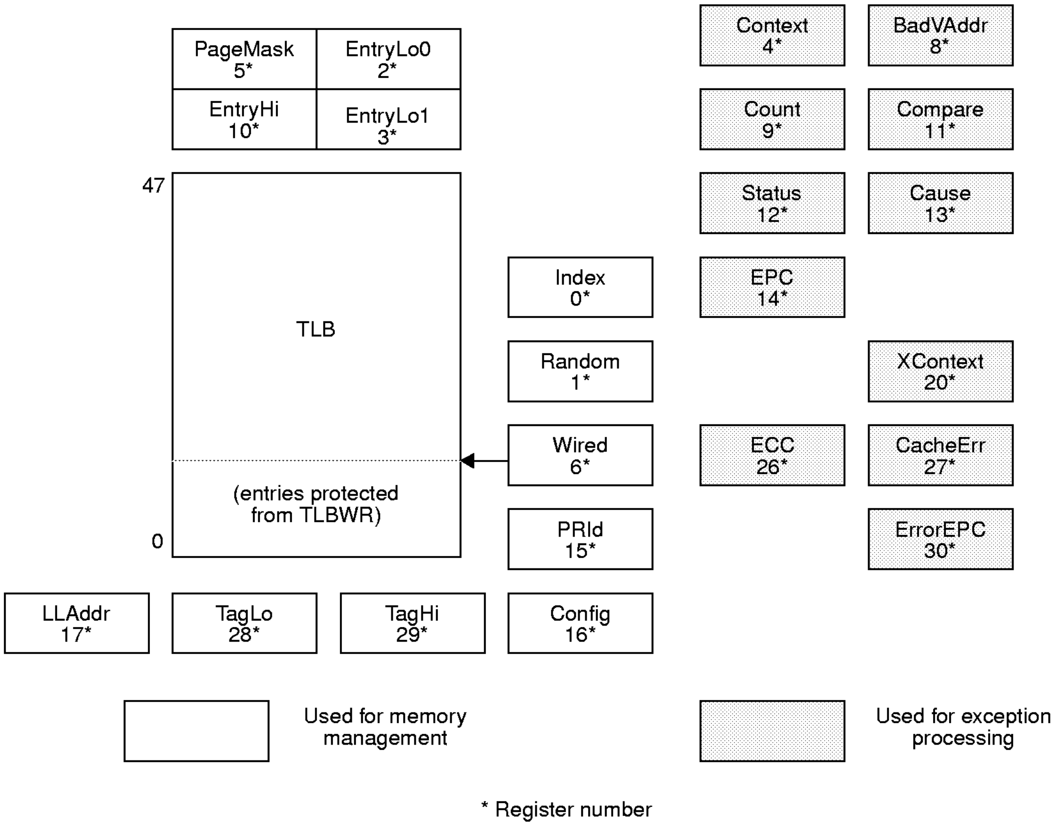


Figure 3 CP0 Registers

interrupts enabled or disabled, cache features). In addition, the RM5260 includes registers to implement a real-time cycle counting facility, to aid in cache diagnostic testing, and to assist in data error detection.

Figure 3 shows the CP0 registers.

Virtual to Physical Address Mapping

The RM5260 provides three modes of virtual addressing:

- user mode
- supervisor mode
- kernel mode

This mechanism is available to system software to provide a secure environment for user processes. Bits in the CP0 *Status* register determine which virtual addressing mode is used. In the user mode, the RM5260 provides a single, uniform virtual address space of 1TB (2GB in 32-bit mode).

When operating in the kernel mode, four distinct virtual address spaces, totalling over 2.5TB (4GB in 32-bit mode), are simultaneously available and are differentiated by the high-order bits of the virtual address.

The RM5260 processors also support a supervisor mode in which the virtual address space over 2TB (2.5GB in 32-bit mode), divided into three regions based on the high-order bits of the virtual address.

Figure 4 on page 6 shows the address space layout for 32-bit operation.

When the RM5260 is configured as a 64-bit microprocessor, the virtual address space layout is an upward compatible extension of the 32-bit virtual address space layout.

Joint TLB

For fast virtual-to-physical address translation, the RM5260 uses a large, fully associative TLB that maps 96 virtual pages to their corresponding physical addresses. As indicated by its name, the joint TLB (JTLB) is used for both instruction and data translations. The JTLB is organized as 48 pairs of even-odd entries, and maps a virtual address and address space identifier into the large, 64GB physical address space.

Two mechanisms are provided to assist in controlling the amount of mapped space and the replacement characteristics of various memory regions. First, the page size can be configured, on a per-entry basis, to use page sizes in the range of 4KB to 16MB (in multiples of 4). A CP0 register, *Page Mask*, is loaded with the desired page size of a mapping, and that size is stored into the TLB along with the virtual address when a new entry is written. Thus, operating systems can create special purpose maps; for example, a typical frame buffer can be memory mapped using only one TLB entry.

0xFFFFFFFF	Kernel virtual address space (kseg3) Mapped, 0.5GB
0xE0000000	
0xDFFFFFFF	Supervisor virtual address space (ksseg) Mapped, 0.5GB
0xC0000000	
0xBFFFFFFF	Uncached kernel physical address space (kseg1) Unmapped, 0.5GB
0xA0000000	
0x9FFFFFFF	Cached kernel physical address space (kseg0) Unmapped, 0.5GB
0x80000000	
0x7FFFFFFF	User virtual address space (kuseg) Mapped, 2.0GB
0x00000000	

Figure 4 Kernel Mode Virtual Addressing (32-bit mode)

The second mechanism controls the replacement algorithm when a TLB miss occurs. The RM5260 provides a random replacement algorithm to select a TLB entry to be written with a new mapping; however, the processor also provides a mechanism whereby a system specific number of mappings can be locked into the TLB, thereby avoiding random replacement. This mechanism allows the operating system to guarantee that certain pages are always mapped for performance reasons and for deadlock avoidance. This mechanism also facilitates the design of real-time systems by allowing deterministic access to critical software.

The JTLB also contains information that controls the cache coherency protocol for each page. Specifically, each page has attribute bits to determine whether the coherency algorithm is: uncached, non-coherent write-back, non-coherent write-through with write-allocate, non-coherent write-through without write-allocate, sharable, exclusive, or update. Note that both of the write-through protocols bypass the secondary cache since the secondary does not support writes of less than a complete cache line.

The non-coherent protocols are used for both code and data on the RM5260 with data using write-back or write-through depending on the application.The write-through modes support the same efficient frame buffer handling as the R4600 and R4700.

The coherent attributes, if used, generate coherent transaction types on the system interface. Like the R5000, however, cache coherency is not supported and, therefore, the coherent attributes should never be used.

Instruction TLB

The RM5260 uses a 2-entry instruction TLB (ITLB) to minimize contention for the JTLB, eliminate the timing critical path of translating through a large associative array, and save power. Each ITLB entry maps a 4KB page. The ITLB improves performance by allowing instruction address translation to occur in parallel with data address translation. When a miss occurs on an instruction address translation by the ITLB, the least-recently used ITLB entry is filled from the JTLB. The operation of the ITLB is completely transparent to the user.

Data TLB

The RM5260 uses a 4-entry data TLB (DTLB) for the same reasons cited above for the ITLB. Each DTLB entry maps a 4KB page. The DTLB improves performance by allowing data address translation to occur in parallel with instruction address translation. When a miss occurs on a data address translation by the DTLB, the DTLB is filled from the JTLB. The DTLB refill is pseudo-LRU: the least recently used entry of the least recently used pair of entries is filled. The operation of the DTLB is completely transparent to the user.

Cache Memory

In order to keep the RM5260's high-performance pipeline full and operating efficiently, the RM5260 incorporates on-chip instruction and data caches that can be accessed in a single processor cycle. Each cache has its own 64-bit data path and both caches can be accessed simultaneously. The cache subsystem provides the integer and floating-point units with an aggregate bandwidth of 3.2GB per second at an internal clock frequency of 200MHz.

Instruction Cache

The RM5260 incorporates a two-way set associative on-chip instruction cache. This virtually indexed, physically tagged cache is 16KB in size and is protected with word parity.

Since the cache is virtually indexed, the virtual-to-physical address translation can occur in parallel with the cache access, thus further increasing performance by allowing these two operations to occur simultaneously. The tag holds a 24-bit physical address and a valid bit, and has a single bit of parity protection.

The instruction cache is 64-bits wide and can be accessed each processor cycle. Accessing 64 bits per cycle allows the instruction cache to supply two instructions per cycle to the superscalar dispatch unit. For typical code sequences where a floating-point load or store and a floating-point computation instruction are being issued together in a loop, the entire bandwidth available from the instruction cache will be consumed.

Cache miss refill writes 64 bits per cycle to minimize the cache miss penalty. The line size is eight instructions (32 bytes) to maximize the performance of communication between the processor and the memory system.

Like the R4650, the RM5260 supports cache locking. The contents of one set of the cache, set A, can be *locked* by setting a bit in the coprocessor 0 *Status* register. Locking the set prevents its contents from being overwritten by a subsequent cache miss. Refill will occur only into set B. This mechanism allows the programmer to lock critical code into the cache thereby guaranteeing deterministic behavior for the locked code sequence.

Data Cache

For fast, single cycle data access, the RM5260 includes a 16KB on-chip data cache that is two-way set associative with a fixed 32-byte (eight words) line size.

The data cache is protected with byte parity and its tag is protected with a single parity bit. It is virtually indexed and physically tagged to allow simultaneous address translation and data cache access.

The normal write policy is write-back, which means that a store to a cache line does not immediately cause memory to be updated. This increases system performance by reducing bus traffic and eliminating the bottleneck of waiting for each store operation to finish before issuing a subsequent memory operation. Software can, however, select write-through on a per-page basis when appropriate, such as for frame buffers. Cache protocols supported for the data cache are:

1. **Uncached.** Reads to addresses in a memory area identified as uncached will not access the cache. Writes to such addresses will be written directly to main memory without updating the cache.
2. **Write-back.** Loads and instruction fetches will first search the cache, reading main memory only if the desired data is not cache resident. On data store operations, the cache is first searched to determine if the target address is cache resident. If it is resident, the cache contents will be updated, and the cache line marked for later write-back. If the cache lookup misses, the target line is first brought into the cache and then the write is performed as above.
3. **Write-through with write allocate.** Loads and instruction fetches will first search the cache, reading main memory only if the desired data is not cache resident. On data store operations, the cache is first searched to determine if the target address is cache resident. If it is resident, the cache contents will be updated and main memory will also be written leaving the *write-back* bit of the cache line unchanged. If the cache lookup misses, the target line is first brought into the cache and then the write is performed as above.
4. **Write-through without write allocate.** Loads and instruction fetches will first search the cache, reading main memory only if the desired data is not cache resident. On data store operations, the cache is first searched to determine if the target address is cache resident. If it is resident, the cache contents will be updated and main memory will also be written leaving the *write-back* bit of the cache line unchanged. If the cache lookup misses, then only main memory is written.

Associated with the Data Cache is the store buffer. When the RM5260 executes a **STORE** instruction, this single-entry buffer gets written with the store data while the tag comparison is performed. If the tag matches, then the data is written into the Data Cache in the next cycle that the Data Cache is not accessed (the next non-load cycle). The store buffer allows the RM5260 to execute a store every processor cycle and to perform back-to-back stores without penalty. In the event of a store immediately followed by a load to the same address, a combined merge and cache write will occur such that no penalty is incurred.

The RM5260 cache attributes for both the instruction and data caches are summarized in Table 3.

Table 3: Cache Attributes

Characteristics	Instruction	Data
Size	16KB	16KB
Organization	2-way set associative	2-way set associative
Line size	32B	32B
Index	$vAddr_{11..0}$	$vAddr_{11..0}$
Tag	$pAddr_{31..12}$	$pAddr_{31..12}$
Write policy	n.a.	write-back/write-through
line transfer order	read: sub-block order	read: sub-block order
	write: sequential	write: sequential
miss restart after transfer of	entire line	first double
Parity	per-word	per-byte
Cache locking	set A	set A

Write buffer

Writes to external memory, whether cache miss write-backs or stores to uncached or write-through addresses, use the on-chip write buffer. The write buffer holds up to four 64-bit address and data pairs. The entire buffer is used for a data cache write-back and allows the processor to proceed in parallel with memory update. For uncached and write-through stores, the write buffer significantly increases performance by decoupling the SysAD bus transfers from the instruction execution stream.

System Interface

The RM5260 provides a high-performance 64-bit multiplexed address/data system interface for optimum price/performance. This interface is compatible with the RM5270, R4600, R4700, and R5000 system interfaces. Unlike the R4600 and R4700 which maintain a constant factor of two between the pipeline clock and the input clock, the RM5260 is like the RM5270 and R5000, and multiplies the **SysClock** input by an integer between 2 and 8, inclusive, to produce the pipeline clock.

The system interface consists of a 64-bit Address/Data bus with 8 check bits and a 9-bit command bus. In addition, there are 6 handshake signals and 6 interrupt inputs. The interface has a simple timing specification and is capable of transferring data between the processor and memory at a peak rate of 600MB/sec with a 75MHz SysClock.

Figure 5 on page 8 shows a typical embedded system using the RM5260. In this example, a bank of DRAMs and a memory controller ASIC share the processor's SysAD bus while the memory controller provides separate ports to a boot ROM and an I/O system.

System Address/Data Bus

The 64-bit System Address Data (SysAD) bus is used to transfer addresses and data between the RM5260 and the rest of the system. It is protected with an 8-bit parity check bus, SysADC.

The system interface is configurable to allow easy interfacing to memory and I/O systems of varying frequencies. The data rate and the bus frequency at which the RM5260 transmits data to the system interface are programmable via boot time mode control bits. Also, the rate at which the processor receives data is fully controlled by the external device. Therefore, either a low-cost interface requiring no read or write buffering, or a faster, high-performance interface can be designed to communicate with the RM5260. Again, the system designer has the flexibility to make these price/performance trade-offs.

System Command Bus

The RM5260 interface has a 9-bit System Command (SysCmd) bus. The command bus indicates whether the SysAD bus carries an address or data. If the SysAD carries an address, then the SysCmd bus also indicates what type of transaction is to take place (for example, a read or write). If the SysAD carries data, then the SysCmd bus also gives information about the data (for example, this is the last data word transmitted, or the data contains an error). The SysCmd bus is bidirectional to support both processor requests and external requests to the RM5260. Processor requests are initiated by the RM5260 and responded to by an external device. External requests are issued by an external device and require the RM5260 to respond.

The RM5260 supports one to eight byte and block transfers on the SysAD bus. In the case of a sub-doubleword transfer, the three low-order address bits give the byte address of the transfer, and the SysCmd bus indicates the number of bytes being transferred.

Handshake Signals

There are six handshake signals on the system interface. Two of these, **RdRdy*** and **WrRdy***, are used by an external device to indicate to the RM5260 whether it can accept a new read or write transaction. The RM5260 samples these signals before deasserting the address on read and write requests.

ExtRqst* and **Release*** are used to transfer control of the SysAD and SysCmd buses from the processor to an external device. When an external device needs to control the interface, it asserts **ExtRqst***. The RM5260 responds by asserting **Release*** to release the system interface to slave state.

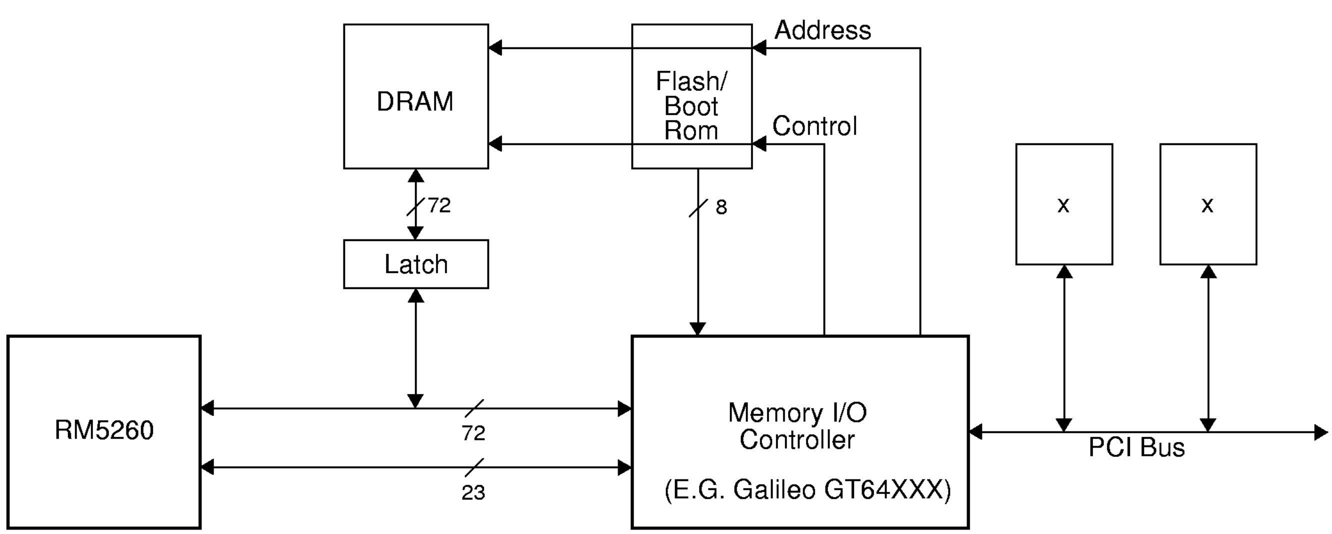


Figure 5 Typical Embedded System Block Diagram

ValidOut* and **ValidIn*** are used by the RM5260 and the external device respectively to indicate that there is a valid command or data on the SysAD and SysCmd buses. The RM5260 asserts **ValidOut*** when it is driving these buses with a valid command or data, and the external device drives **ValidIn*** when it has control of the buses and is driving a valid command or data.

Non-overlapping System Interface

The RM5260 requires a non-overlapping system interface, compatible with the R5000. This means that only one processor request may be outstanding at a time and that the request must be serviced by an external device before the RM5260 issues another request. The RM5260 can issue read and write requests to an external device, whereas an external device can issue null and write requests to the RM5260.

For processor reads the RM5260 asserts **ValidOut*** and simultaneously drives the address and read command on the SysAD and SysCmd buses. If the system interface has **RdRdy*** asserted, then the processor tristates its drivers and releases the system interface to slave state by asserting **Release***. The external device can then begin sending data to the RM5260.

Figure 6 on page 9 shows a processor block read request and the external agent read response. The read latency is 4 cycles (**ValidOut*** to **ValidIn***), and the response data pattern is DDxxDD. Figure 7 on page 9 shows a processor block write using write response pattern DDxxDDxx, or code 2, of the boot-time mode select options.

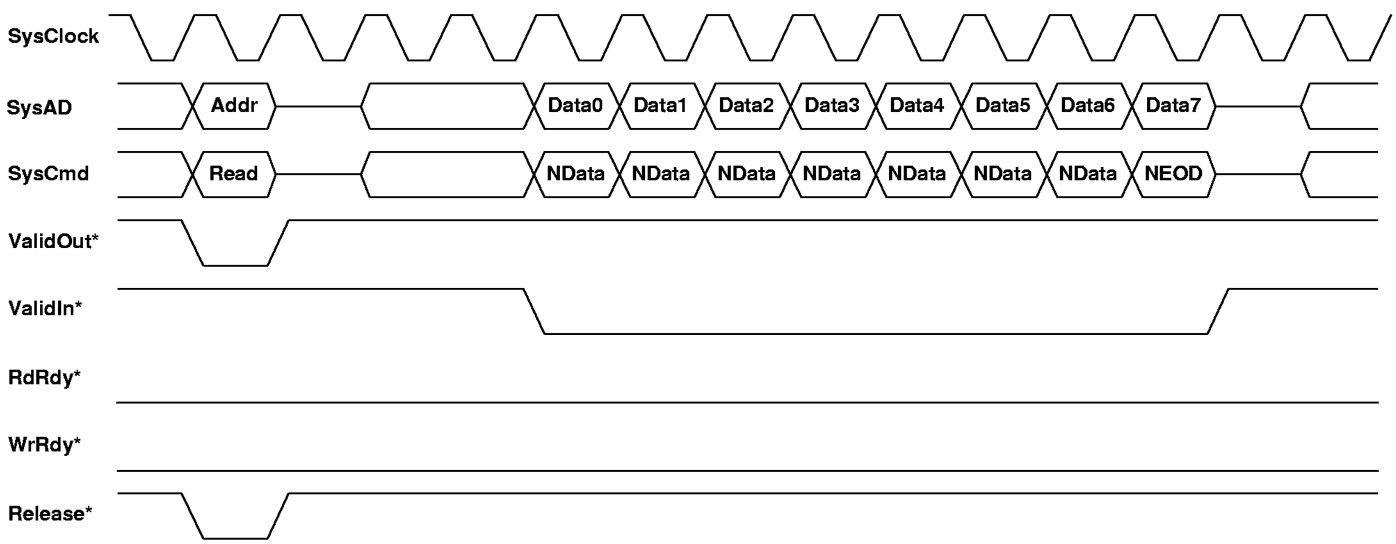


Figure 6 Processor Block Read

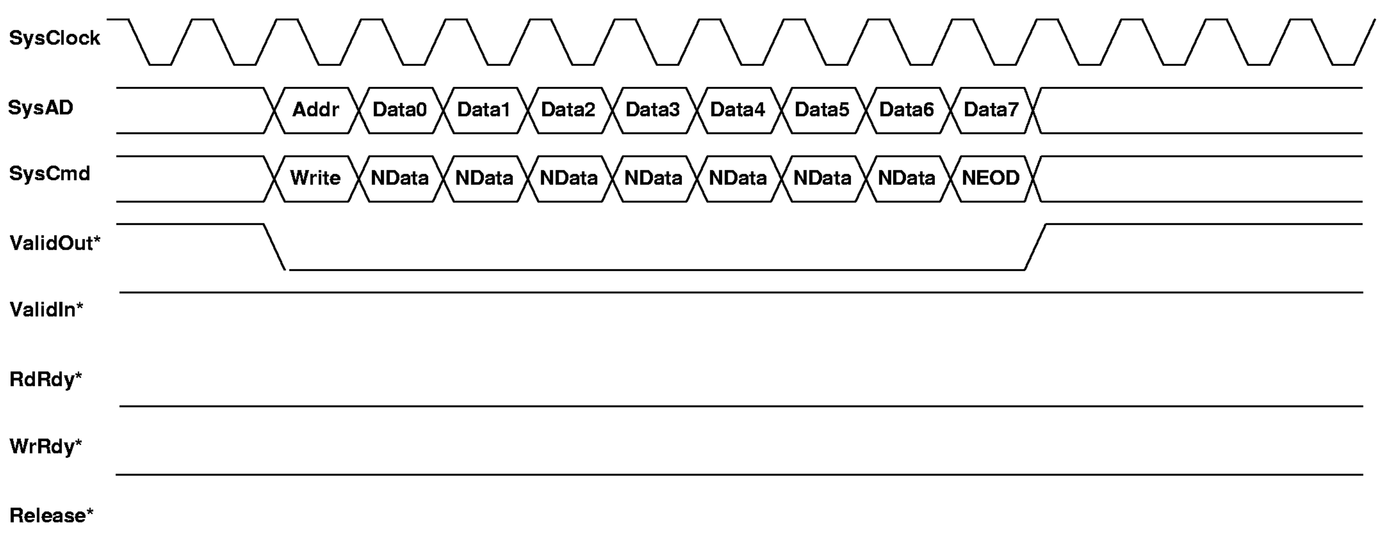


Figure 7 Processor Block Write

Enhanced Write Modes

Like the RM5270, R4700, and R5000, the RM5260 implements two enhancements to the original R4000 write mechanism: Write Reissue and Pipeline Writes. In write reissue mode, a write rate of one write every two bus cycles can be achieved. A write issues if **WrRdy*** is asserted two cycles earlier and is still asserted during the issue cycle. If it is not still asserted then the last write will reissue. Pipelined writes have the same two bus cycle write repeat rate, but can issue one additional write following the deassertion of **WrRdy***.

External Requests

The RM5260 can respond to certain requests issued by an external device. These requests take one of two forms: Write requests and Null requests. An external device executes a write request when it wishes to update one of the processors writable resources such as the internal interrupt register. A null request is executed when the external device wishes the processor to reassert ownership of the processor external interface; i.e., the external device wants the processor interface to go from slave state to master state. Typically, a null request will be executed after an external device, that has acquired control of the processor interface via **ExtRqst***, has completed a transaction between itself and system memory in a system where memory is connected directly to the SysAD bus. Normally, this transaction would be a DMA read or write from the I/O system.

Interrupt Handling

In order to provide better real time interrupt handling, the RM5260 supports the same dedicated interrupt vector introduced in the R4650. When enabled by the real time executive, by setting a bit in the Cause register, interrupts vector to a specific address which is not shared with any of the other exception types. This capability eliminates the need to go through the normal software routine for exception decode and dispatch thereby lowering interrupt latency.

Standby Mode

The RM5260 provides a means to reduce the amount of power consumed by the internal core when the CPU would otherwise not be performing any useful operations. This state is known as Standby Mode.

Executing the **WAIT** instruction enables interrupts and enters Standby Mode. When the wait instruction completes the W pipe stage, if the SysAD bus is currently idle, the internal processor clocks will stop thereby freezing the pipeline. The phase lock loop, or PLL, internal timer/counter, and the “wake up” input pins: **Int[5:0]***, **NMI***, **ExtReq***, **Reset***, and **ColdReset*** will continue to operate in their normal fashion. If the SysAD bus is not idle when the **WAIT** instruction completes the W pipe-stage, then the **WAIT** is treated as a **NOP**. Once the processor is in Standby, any interrupt, including the internally generated timer interrupt, will cause the processor to exit Standby and resume operation where it left off. The **WAIT** instruction is typically inserted in the idle loop of the operating system or real time executive.

JTAG Interface

The RM5260 interface supports JTAG boundary scan in conformance with IEEE 1149.1. The JTAG interface is especially helpful for checking the integrity of the processors pin connections.

Boot-Time Options

Fundamental operational modes for the processor are initialized by the boot-time mode control interface. The boot-time mode control interface is a serial interface operating at a very low frequency (**SysClock** divided by 256). The low frequency operation allows the initialization information to be kept in a low cost EPROM; alternatively the twenty or so bits could be generated by the system interface ASIC.

Immediately after the **VccOk** signal is asserted, the processor reads a serial bit stream of 256 bits to initialize all the fundamental operational modes. **ModeClock** runs continuously from the assertion of **VccOk**.

Boot-Time Modes

The boot-time serial mode stream is defined in Table 4. Bit 0 is the bit presented to the processor when **VccOk** is de-asserted; bit 255 is the last.

Table 4: Boot-Time Mode Bit Stream

Mode bit	Description
0	Reserved: Must be zero
4..1	Write-back data rate 0: DDDD 1: DDxDDx 2: DDxxDDxx 3: DxDxDxDx 4: DDxxxDDxxx 5: DDxxxxDDxxxx 6: DxxDxxDxxDxx 7: DDxxxxxxDDxxxxxx 8: DxxxDxxxDxxxDxxx 9-15 reserved
7..5	Pclock to SysClock Multiplier 0: Multiply by 2 1: Multiply by 3 2: Multiply by 4 3: Multiply by 5 4: Multiply by 6 5: Multiply by 7 6: Multiply by 8 7: reserved
8	Specifies byte ordering. Logically ORed with BigEndian input signal. 0: Little endian 1: Big endian
10..9	00: R4000 compatible non-block writes 01: reserved 10: pipelined non-block writes 11: non-block write re-issue
11	0: Enable the timer interrupt on Int[5] 1: Disable the timer interrupt on Int[5]
12	Reserved: Must be zero
14..13	Output driver strength - 100% = fastest 00: 67% strength 01: 50% strength 10: 100% strength 11: 83% strength
15	Reserved: Must be zero
17..16	System configuration identifiers - software visible in processor Config[21..20] register
255..18	Reserved: Must be zero

PIN DESCRIPTIONS:

The following is a list of interface, interrupt, and miscellaneous pins available on the RM5260.

Pin Name	Type	Description
System interface:		
ExtRqst*	Input	External request Signals that the system interface is submitting an external request.
Release*	Output	Release interface Signals that the processor is releasing the system interface to slave state
RdRdy*	Input	Read Ready Signals that an external agent can now accept a processor read.
WrRdy*	Input	Write Ready Signals that an external agent can now accept a processor write request.
ValidIn*	Input	Valid Input Signals that an external agent is now driving a valid address or data on the SysAD bus and a valid command or data identifier on the SysCmd bus.
ValidOut*	Output	Valid output Signals that the processor is now driving a valid address or data on the SysAD bus and a valid command or data identifier on the SysCmd bus.
SysAD(63:0)	Input/Output	System address/data bus A 64-bit address and data bus for communication between the processor and an external agent.
SysADC(7:0)	Input/Output	System address/data check bus An 8-bit bus containing parity check bits for the SysAD bus during data cycles.
SysCmd(8:0)	Input/Output	System command/data identifier bus A 9-bit bus for command and data identifier transmission between the processor and an external agent.
SysCmdP	Input/Output	Reserved for system command/data identifier bus parity For the RM5260, unused on input and zero on output.
Clock/control interface:		
SysClock	Input	System clock Master clock input used as the system interface reference clock. All output timings are relative to this input clock. Pipeline operation frequency is derived by multiplying this clock up by the factor selected during boot initialization
VccP	Input	Quiet Vcc for PLL Quiet Vcc for the internal phase locked loop.
VssP	Input	Quiet VSS for PLL Quiet Vss for the internal phase locked loop.
Interrupt interface:		
Int*(5:0)	Input	Interrupt Six general processor interrupts, bit-wise ORed with bits 5:0 of the interrupt register.
NMI*	Input	Non-maskable interrupt Non-maskable interrupt, ORed with bit 6 of the interrupt register.
JTAG interface:		
JTDI	Input	JTAG data in JTAG serial data in.
JTCK	Input	JTAG clock input JTAG serial clock input.
JTDO	Output	JTAG data out JTAG serial data out.
JTMS	Input	JTAG command JTAG command signal, signals that the incoming serial data is command data.
Initialization interface:		
BigEndian	Input	Allows the system to change the processor addressing mode without rewriting the mode ROM.

Pin Name	Type	Description
VCCOk	Input	VCC is OK When asserted, this signal indicates to the RM5260 that the 3.3V power supply has been above 3.0V for more than 100 milliseconds and will remain stable. The assertion of VCCOk initiates the reading of the boot-time mode control serial stream.
ColdReset*	Input	Cold reset This signal must be asserted for a power on reset or a cold reset. ColdReset must be de-asserted synchronously with SysClock.
Reset*	Input	Reset This signal must be asserted for any reset sequence. It may be asserted synchronously or asynchronously for a cold reset, or synchronously to initiate a warm reset. Reset must be de-asserted synchronously with SysClock.
ModeClock	Output	Boot mode clock Serial boot-mode data clock output at the system clock frequency divided by 256.
ModeIn	Input	Boot mode data in Serial boot-mode data input.

ABSOLUTE MAXIMUM RATINGS:[1]

Symbol	Rating	Limits	Unit
V_{TERM}	Terminal Voltage with respect to GND	–0.5[2] to +5.5	V
T_{CASE}	Operating Temperature	0 to +85	°C
T_{BIAS}	Case Temperature Under Bias	–55 to +125	°C
T_{STG}	Storage Temperature	–55 to +125	°C
I_{IN}	DC Input Current	20[3]	mA
I_{OUT}	DC Output Current	50	mA

Notes: 1. Stresses greater than those listed under ABSOLUTE MAXIMUM RATINGS may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect reliability.

2. V_{IN} minimum = -2.0V for pulse width less than 15ns. V_{IN} should not exceed 5.5 Volts.
3. When $V_{IN} < 0V$ or $V_{IN} > VCC$
4. Not more than one output should be shorted at a time. Duration of the short should not exceed 30 seconds.

RECOMMENDED OPERATING TEMPERATURE AND SUPPLY VOLTAGE:

Grade	Temperature	GND	VccInt	VccIO
Commercial	0°C to +85°C (Case)	0V	3.3V±5%	3.3V±5%

DC ELECTRICAL CHARACTERISTICS:

(VccInt = VccIO = 3.3V ± 5%; T_{CASE} = 0°C to +85°C)

Parameter	133/150/175/200 MHz		Conditions
	Minimum	Maximum	
V_{OL}		0.1V	$\|I_{OUT}\|$= 20 µA
V_{OH}	VccIO - 0.1V		
V_{OL}		0.4V	$\|I_{OUT}\|$ = 4 mA
V_{OH}	2.4V		
V_{IL}	-0.5V	0.2 x VccIO	
V_{IH}	0.7 x VccIO	VccIO + 0.5V	
I_{IN}		±20 µA ±20 µA ±250 µA	V_{IN} = 0 V_{IN} = VccIO V_{IN} = 5.5V
C_{IN}		10pF	
C_{OUT}		10pF	

POWER CONSUMPTION:

Parameter		Conditions Max. = 3.45 Vcc, Typ.[5] = 3.3 Vcc	133/44 MHz, 3.45V		150/50 MHz, 3.45V		175/58 MHz, 3.45V		200/100 MHz, 3.45V	
			Typ[5]	Max	Typ[5]	Max	Typ[5]	Max	Typ[5]	Max
ICC (mA)	standby	CL = 0pF	50	100	60	100	60	100	70	110
		CL = 50pF	50	100	60	100	60	100	70	110
	active	CL = 0pF, no SysAD activity	650	1250	725	1450	800	1650	1000	1950
		CL = 50pF, R4000 write protocol without FPU operation	725	1450	850	1650	1000	1900	1125	2200
		CL = 50pF, write re-issue or pipe-lined writes	800	1550	950	1800	1075	2100	1250	2400

Note: 5. Typical integer instruction mix and cache miss rates.

AC ELECTRICAL CHARACTERISTICS:

(VccInt = VccIO = 3.3V ± 5%; T_{CASE} = 0°C to +85°C)

Capacitive Load Deration:

Parameter	Symbol	133/150/175/200 MHz		Units
		Min	Max	
Load Derate	C_{LD}		2	ns/25pF

Clock Parameters:

Parameter	Symbol	Test Conditions	133 MHz		150 MHz		175 MHz		200 MHz		Units
			Min	Max	Min	Max	Min	Max	Min	Max	
SysClock High	t_{SCH}	Transition £ 5ns	4		4		4		4		ns
SysClock Low	t_{SCL}	Transition £ 5ns	4		4		4		4		ns
SysClock Frequency[6]			20	67	20	75	20	87.5	20	100	MHz
SysClock Period	t_{SCP}			50		50		50		50	ns
Clock Jitter for SysClock	t_{JI}			±250		±250		±200		±200	ps
SysClock Rise Time	t_{CR}			5		5		4		3	ns
SysClock Fall Time	t_{CF}			5		5		4		3	ns
ModeClock Period	$t_{ModeCKP}$			256*t_{SCP}		256*t_{SCP}		256*t_{SCP}		256*t_{SCP}	ns
JTAG Clock Period	$t_{JTAGCKP}$			4*t_{SCP}		4*t_{SCP}		4*t_{SCP}		4*t_{SCP}	ns

Note: 6. Operation of the RM5260 is only guaranteed with the Phase Lock Loop Enabled.

System Interface Parameters:[7]

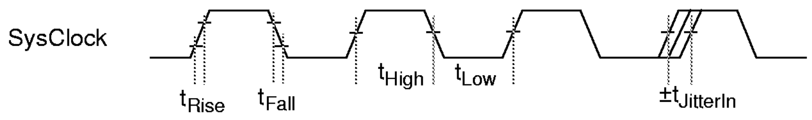
Parameter	Symbol	Test Conditions	133 MHz		150 MHz		175 MHz		200 MHz		Units
			Min	Max	Min	Max	Min	Max	Min	Max	
Data Output[8,9]	t_{DO}	mode14..13 = 10 (fastest)	1.0	4.5	1.0	4.5	1.0	4.5	1.0	4.5	ns
		mode14..13 = 11	1.0	5.0	1.0	5.0	1.0	5.0	1.0	5.0	ns
		mode14..13 = 00	1.0	5.5	1.0	5.5	1.0	5.5	1.0	5.5	ns
		mode14..13 = 01 (slowest)	1.0	6.5	1.0	6.5	1.0	6.5	1.0	6.5	ns
Data Setup[10]	t_{DS}	t_{rise} = see above table	3.0		3.0		2.5		2.5		ns
Data Hold[10]	t_{DH}	t_{fall} = see above table	1.0		1.0		1.0		1.0		ns

Notes: 7. Timings are measured from 1.5V of the clock to 1.5V of the signal.
8. Capacitive load for all output timings is 50pF.
9. Data Output timing applies to all signal pins whether tristate I/O or output only.
10. Setup and Hold parameters apply to all signal pins whether tristate I/O or input only.

Boot-Time Interface Parameters:

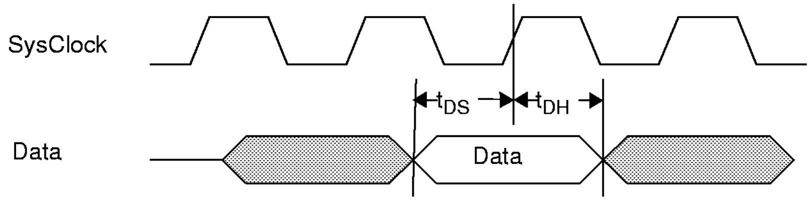
Parameter	Symbol	Test Conditions	133/150/175/200 MHz		Units
			Min	Max	
Mode Data Setup	t_{DS}(M)		4		SysClock cycles
Mode Data Hold	t_{DH}(M)		0		SysClock cycles

TIMING DIAGRAMS:

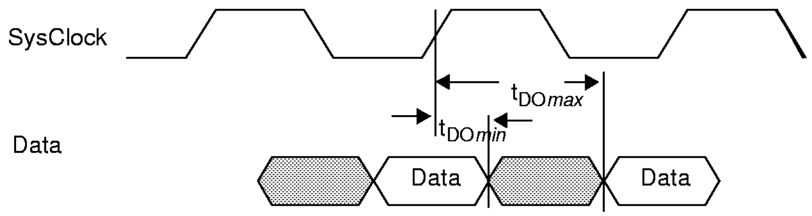


Clock Timing

System Interface Timing (SysAD, SysCMD, ValindIn*, ValidOut*, etc.)

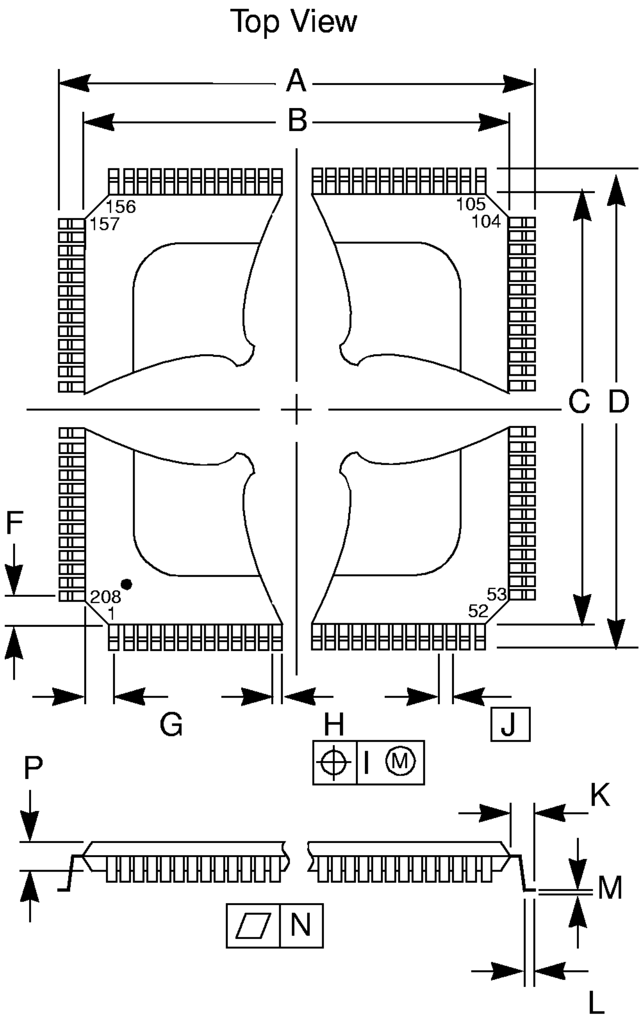


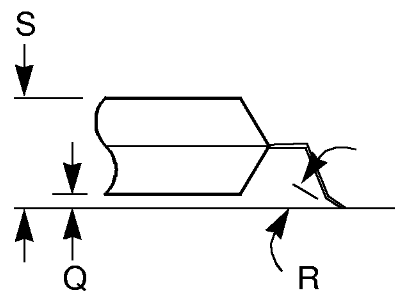
Input Timing



Output Timing

PACKAGING INFORMATION:





RM5260 208 PQUAD PACKAGE

ITEM	INCHES	MILLIMETERS
A	1.228 ± .010	31.2 ± 0.25
B	1.102 ±.004	28.0 ± 0.10
C	1.102 ± .004	28.0 ± 0.10
D	1.228 ± .010	31.2 ± 0.25
F	.049	1.25
G	.049	1.25
H	.0075 ± .0025	0.19 ± 0.06
I		
J	.0197	0.50
K	.049	1.25
L	.028 -.002 +.009	0.70 -0.05 +0.25
M	.003	0.08
N		
P	.133 -.008 +.011	3.37 -0.2 +0.3
Q	.013 - .003	0.33 - 0.08
R	7°	7°
S	.146 +.008	3.7 + 0.3
ThetaJA		
ThetaJC		

RM5260 208 P-QUAD PACKAGE PINOUT:

(Note: VccIO = VccInt)

Pin	Function	Pin	Function	Pin	Function	Pin	Function
1	VccIO	2	NC	3	NC	4	VccIO
5	Vss	6	SysAD4	7	SysAD36	8	SysAD5
9	SysAD37	10	VccInt	11	Vss	12	SysAD6
13	SysAD38	14	VccIO	15	Vss	16	SysAD7
17	SysAD39	18	SysAD8	19	SysAD40	20	VccInt
21	Vss	22	SysAD9	23	SysAD41	24	VccIO
25	Vss	26	SysAD10	27	SysAD42	28	SysAD11
29	SysAD43	30	VccInt	31	Vss	32	SysAD12
33	SysAD44	34	VccIO	35	Vss	36	SysAD13
37	SysAD45	38	SysAD14	39	SysAD46	40	VccInt
41	Vss	42	SysAD15	43	SysAD47	44	VccIO
45	Vss	46	ModeClock	47	JTDO	48	JTDI
49	JTCK	50	JTMS	51	VccIO	52	Vss
53	NC	54	NC	55	NC	56	VccIO
57	Vss	58	ModeIn	59	RdRdy*	60	WrRdy*
61	ValidIn*	62	ValidOut*	63	Release*	64	VccP
65	VssP	66	SysClock	67	VccInt	68	Vss
69	VccIO	70	Vss	71	VccInt	72	Vss
73	SysCmd0	74	SysCmd1	75	SysCmd2	76	SysCmd3
77	VccIO	78	Vss	79	SysCmd4	80	SysCmd5
81	VccIO	82	Vss	83	SysCmd6	84	SysCmd7
85	SysCmd8	86	SysCmdP	87	VccInt	88	Vss
89	VccInt	90	Vss	91	VccIO	92	Vss
93	Int0*	94	Int1*	95	Int2*	96	Int3*
97	Int4*	98	Int5*	99	VccIO	100	Vss
101	NC	102	NC	103	NC	104	NC
105	VccIO	106	NMI*	107	ExtRqst*	108	Reset*
109	ColdReset*	110	VccOK	111	BigEndian	112	VccIO
113	Vss	114	SysAD16	115	SysAD48	116	VccInt
117	Vss	118	SysAD17	119	SysAD49	120	SysAD18
121	SysAD50	122	VccIO	123	Vss	124	SysAD19
125	SysAD51	126	VccInt	127	Vss	128	SysAD20
129	SysAD52	130	SysAD21	131	SysAD53	132	VccIO
133	Vss	134	SysAD22	135	SysAD54	136	VccInt
137	Vss	138	SysAD23	139	SysAD55	140	SysAD24
141	SysAD56	142	VccIO	143	Vss	144	SysAD25
145	SysAD57	146	VccInt	147	Vss	148	SysAD26
149	SysAD58	150	SysAD27	151	SysAD59	152	VccIO
153	Vss	154	NC	155	NC	156	Vss
157	NC	158	NC	159	NC	160	NC
161	VccIO	162	Vss	163	SysAD28	164	SysAD60
165	SysAD29	166	SysAD61	167	VccInt	168	Vss
169	SysAD30	170	SysAD62	171	VccIO	172	Vss
173	SysAD31	174	SysAD63	175	SysADC2	176	SysADC6

Pin	Function	Pin	Function	Pin	Function	Pin	Function
177	VccInt	178	Vss	179	SysADC3	180	SysADC7
181	VccIO	182	Vss	183	SysADC0	184	SysADC4
185	VccInt	186	Vss	187	SysADC1	188	SysADC5
189	SysAD0	190	SysAD32	191	VccIO	192	Vss
193	SysAD1	194	SysAD33	195	VccInt	196	Vss
197	SysAD2	198	SysAD34	199	SysAD3	200	SysAD35
201	VccIO	202	Vss	203	NC	204	NC
205	NC	206	NC	207	VccIO	208	Vss

ORDERING INFORMATION:

RM5260 -123 A

- A: Package Type
  Q=Power Quad 4 (PQ-4)
- -123: Device Maximum Speed
- RM5260: Device Type

Valid Combinations:

RM5260-133Q
RM5260-150Q
RM5260-175Q
RM5260-200Q

Quantum Effect Design, Inc.
3255-3 Scott Blvd. Suite 200
Santa Clara, CA 95054
(408) 565-0300
(408) 565-0335 (fax)
http://www.qedinc.com
For a complete listing of QED Sales Representatives and Offices, please visit our internet website.