



Integrated Device Technology, Inc.

LOW-COST EMBEDDED ORION™ RISC MICROPROCESSOR

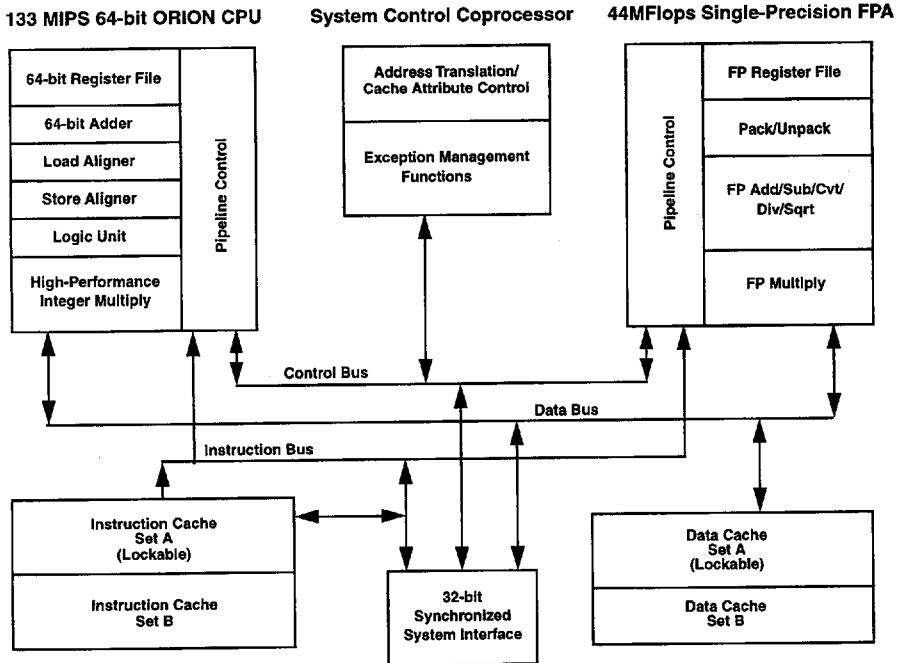
ORION
IDT79R4640™
IDT79RV4640™
Preliminary

FEATURES

- High-performance embedded 64-bit microprocessor
 - 64-bit integer operations
 - 64-bit registers
 - Based on the MIPS RISC Architecture
 - 80MHz, 100MHz, 133MHz operation frequency
 - 32-bit bus interface brings 64-bit power to 32-bit system cost
- High-performance DSP capability
 - 66.7 Million Integer Mul-Accumulate operations/sec @ 133MHz
 - 44 MFlops floating point operations @ 133MHz
- High-performance microprocessor
 - 133 MIPS @ 133MHz
 - 66.7 M Mul-Add/second @ 133MHz
 - 44 MFlops @ 133MHz
 - >300,000 dhrystone (2.1)/sec capability @ 133MHz (175 dhrystone MIPS)
- High level of integration
 - 64-bit, 133 MIPS integer CPU
 - 44MFlops single precision floating-point unit
 - 8KB instruction cache; 8KB data cache
 - Integer multiply unit with 66.7M Mul-Add/sec

- Upwardly software compatible with IDT RISController Family
- Easily upgradable to 64-bit system
- Low-power operation
 - Active power management powers-down inactive units
 - Standby mode
- Large, efficient on-chip caches
 - Separate 8KB Instruction and 8KB Data caches
 - Over 1500MB/sec bandwidth from internal caches
 - 2-set associative
 - Write-back and write-through support
 - Cache locking to facilitate deterministic response
 - High performance write protocols for graphics and data communications
- Bus compatible with ORION family
 - System interfaces to 67 MHz, provides bandwidth up to 266 MB/sec
 - Direct interface to 32-bit wide systems
 - Synchronized to external reference clock for multi-master operation
- Improved real-time support
 - Fast interrupt decode
 - Optional cache locking

BLOCK DIAGRAM



The IDT logo is a registered trademark and ORION, R4650, R4640, RV4640, R4600, R3081, R3052, R3051, R3041, RISController, and RISCORE are trademarks of Integrated Device Technology, Inc.

COMMERCIAL TEMPERATURE RANGE

JUNE 1996

DESCRIPTION

The IDT79R4640 is a low-cost member of the Integrated Device Technology, Inc. ORION family, targeted to a variety of performance-hungry embedded applications. The R4640 continues the ORION tradition of high-performance through high-speed pipelines, high-bandwidth caches and bus interface, 64-bit architecture, and careful attention to efficient control. The cost of this performance is reduced by removing functional units frequently not required for many embedded applications.

The R4640 supports a wide variety of embedded processor-based applications, such as internetworking equipment (routers, switches), office automation equipment (printers, scanners), and consumer multimedia game systems. Also, being upwardly software-compatible with the RISCController family as well as bus- and upwardly software-compatible with the IDT ORION family, the R4640 will serve in many of the same applications. And, the R4640 supports applications that require integer digital signal processing (DSP) functions.

The R4640 brings ORION performance levels to lower cost systems. ORION performance is preserved by retaining large on-chip two-way set-associative caches, a streamlined high-speed pipeline, high-bandwidth, 64-bit execution, and facilities such as early restart for data cache misses.

These techniques combine to allow the system designer more than 1.5GB/sec aggregate internal bandwidth, 266 MB/sec bus bandwidth, 175 Dhrystone MIPS, 44MFlops, and 66.7 M Mul-Add/sec. An array of development tools facilitates rapid development of R4640-based systems, allowing a wide variety of customers to take advantage of the processor's high-performance capabilities while maintaining short time-to-market goals.

HARDWARE OVERVIEW

Some key elements of the R4640 are briefly described below. More detailed information is available in the *IDT79R4640/IDT79R4650 RISC Processor Hardware User's Manual*.

Pipeline

The R4640 uses a 5-stage pipeline that is similar to the IDT79R3000 and the IDT79R4700 processors. The simplicity of this pipeline allows the R4640 to cost less than super-scalar processors and require less power than super-pipelined processors. So, unlike superscalar processors, applications that have large data dependencies or require a great deal of load/stores can still achieve peak performance.

Integer Execution Engine

The R4640 implements the MIPS-III Instruction Set Architecture, and thus is fully upward compatible with applications running on the earlier generation parts. The

R4640 is software-compatible with the R4650, and includes the instruction set found in the R4700 microprocessor, targeted at higher performance while maintaining binary compatibility with earlier R30xx processors. The extensions result in better code density, greater multiprocessing support, improved performance for commonly used code sequences in operating system kernels, and faster execution of floating-point intensive applications. All resource dependencies are made transparent to the programmer, insuring transportability among implementations of the MIPS instruction set architecture. In addition, MIPS-III specifies new instructions defined to take advantage of the 64-bit architecture of the processor.

Finally, the R4640 also implements additional instructions, which are considered extensions to the MIPS-III architecture. These instructions improve the multiply and multiply-add throughput of the CPU, making it well suited to a wide variety of imaging and DSP applications. These extensions, which use opcodes allocated by MIPS Technologies for this purpose, are supported by a wide variety of development tools.

The MIPS integer unit implements a load/store architecture with single cycle ALU operations (logical, shift, add, sub) and autonomous multiply/divide unit. The 64-bit register resources include: 32 general-purpose orthogonal integer registers, the HI/LO result registers for the integer multiply/divide unit, and the program counter. In addition, the on-chip floating-point co-processor adds 32 floating-point registers, and a floating-point control/status register.

Register File

The R4640 has 32 general-purpose 64-bit registers. These registers are used for scalar integer operations and address calculation. The register file consists of two read ports and one write port and is fully bypassed to minimize operation latency in the pipeline.

Arithmetic Logic Unit

The R4640 ALU consists of the integer adder and logic unit. The adder performs address calculations in addition to arithmetic operations; the logic unit performs all of the logic and shift operations. Each unit is highly optimized and can perform an operation in a single pipeline cycle.

Integer Multiply/Divide

The R4640 uses a dedicated integer multiply/divide unit, optimized for high-speed multiply and multiply-accumulate operation. Table 1 shows the performance,

expressed in terms of pipeline clocks, achieved by the R4640 integer multiply unit.

Opcode	Operand Size	Latency	Repeat	Stall
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	6	5	0
DIV, DIVU	any	36	36	0
DDIV, DDIVU	any	68	68	0

Table 1: R4640 Integer Multiply Operation

The MIPS-III architecture defines that the results of a multiply or divide operation are placed in the HI and LO registers. The values can then be transferred to the general purpose register file using the MFHI/MFLO instructions.

The R4640 adds a new multiply instruction, "MUL", which can specify that the multiply results bypass the "Lo" register and are placed immediately in the primary register file. By avoiding the explicit "Move-from-Lo" instruction required when using "Lo", throughput of multiply-intensive operations is increased.

An additional enhancement offered by the R4640 is an atomic "multiply-add" operation, MAD, used to perform multiply-accumulate operations. This instruction multiplies two numbers and adds the product to the current contents of the HI and LO registers. This operation is used in numerous DSP algorithms, and allows the R4640 to cost reduce systems requiring a mix of DSP and control functions.

Finally, aggressive implementation techniques feature low latency for these operations along with pipelining to allow new operations to be issued before a previous one has fully completed. Table 1 also shows the repeat rate (peak issue rate), latency, and number of processor stalls required for the various operations. The R4640 performs automatic operand size detection to determine the size of the operand, and implements hardware interlocks to prevent overrun, allowing this high-performance to be achieved with simple programming.

Floating-Point Coprocessor

The R4640 incorporates an entire single-precision floating-point coprocessor on chip, including a floating-point register file and execution units. The floating-point coprocessor forms a "seamless" interface with the integer unit, decoding and executing instructions in parallel with the integer unit.

The floating-point unit of the R4640 directly implements single-precision floating-point operations, which enables the R4640 to perform functions such as graphics rendering without requiring extensive die area or power consumption. The single-precision unit of the R4640 is directly compatible with the single-precision operation of the R4700, and features the same latencies and repeat rates.

The R4640 does not directly implement the double-precision operations found in the R4700. However, to maintain software compatibility, the R4640 will signal a trap when a double-precision operation is initiated, allowing the requested function to be emulated in software. Alternatively, the system architect could use a software library emulation of double-precision functions, selected at compile time, to eliminate the overhead associated with trap and emulation.

Floating-Point Units

The R4640's floating-point execution units perform single precision arithmetic, as specified in IEEE Standard 754. The execution unit is broken into a separate multiply unit and a combined add/convert/divide/square root unit. Overlap of multiply and add/subtract is supported. The multiplier is partially pipelined, allowing a new multiplication instruction to begin every 6 cycles.

As in the IDT79R4700, the R4640 maintains fully precise floating-point exceptions while allowing both overlapped and pipelined operations. Precise exceptions are extremely important in mission-critical environments, such as ADA, and highly desirable for debugging in any environment.

The floating-point unit's operation set includes floating-point add, subtract, multiply, divide, square root, conversion between fixed-point and floating-point format, conversion among floating-point formats, and floating-point compare. These operations comply with IEEE Standard 754. Double precision operations are not directly supported; attempts to execute double-precision floating point operations, or refer directly to double-precision registers, result in the R4640 signalling a "trap" to the CPU, enabling emulation of the requested function.

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

Operation	Instruction Latency
ADD	4
SUB	4
MUL	8
DIV	32
SQRT	31
CMP	3
FIX	4
FLOAT	6
ABS	1
MOV	1
NEG	1
LWC1	2
SWC1	1

Table 2: Floating-Point Operation

Floating-Point General Register File

The floating-point register file is made up of thirty-two 32-bit registers. These registers are used as source or target registers for the single-precision operations.

References to these registers as 64-bit registers (as supported in the R4700) will cause a trap to be signalled to the integer unit.

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 involved with diagnostic software, exception handling, state saving and restoring, and control of rounding modes.

System Control Coprocessor (CP0)

The system control co-processor in the MIPS architecture is responsible for the virtual to physical address translation and cache protocols, 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.

In the R4640, significant changes in CP0 relative to the R4600 have been implemented. These changes are designed to simplify memory management, facilitate debug, and speed real-time processing.

System Control Co-Processor Registers

The R4640 incorporates all system control co-processor (CP0) registers on-chip. These registers provide the path through which the virtual memory system's address translation is controlled, exceptions are handled, and operating modes are controlled (kernel vs. user mode, interrupts enabled or disabled, cache features). In addition, the R4640 includes registers to implement a real-time cycle counting facility, which aids in cache diagnostic testing, assists in data error detection, and facilitates software debug. Alternatively, this timer can be used as the operating system reference timer, and can signal a periodic interrupt.

Table 3 shows the CP0 registers of the R4640.

Number	Name	Function
0	IBase	Instruction address space base (new in R4640)
1	IBound	Instruction address space bound (new in R4640)
2	DBase	Data address space base (new in R4640)
3	DBound	Data address space bound (new in R4640)
4-7, 10, 20-25, 29, 31	-	Not used
8	BadVAddr	Virtual address on address exceptions
9	Count	Counts every other cycle
11	Compare	Generate interrupt when Count = Compare
12	Status	Miscellaneous control/status
13	Cause	Exception/Interrupt information
14	EPC	Exception PC
15	PRId	Processor ID
16	Config	Cache and system attributes
17	CAIg	Cache attributes for the 8 512MB regions of the virtual address space
18	IWatch	Instruction breakpoint virtual address
19	DWatch	Data breakpoint virtual address
26	ECC	Used in cache diagnostics
27	CacheErr	Cache diagnostic information
28	TagLo	Cache index information
30	ErrorEPC	CacheError exception PC

Table 3: R4640 CPO Registers

Operation modes

The R4640 supports two modes of operation: user mode and kernel mode.

Kernel mode operation is typically used for exception handling and operating system kernel functions, including

CP0 management and access to IO devices. In kernel mode, software has access to the entire address space and all of the co-processor 0 registers, and can select whether to enable co-processor 1 accesses. The processor enters kernel mode at reset, and whenever an exception is recognized.

User mode is typically used for applications programs. User mode accesses are limited to a subset of the virtual address space, and can be inhibited from accessing CP0 functions.

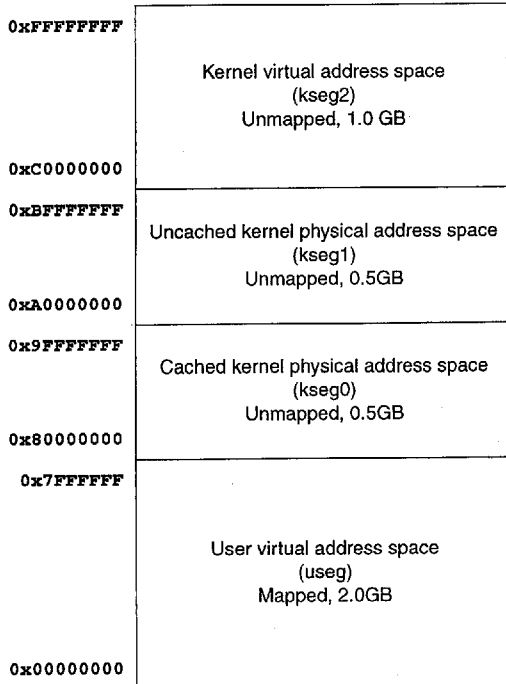


Figure 3: Mode Virtual Addressing (32-bit mode)

Virtual to Physical Address Mapping

The 4GB virtual address space of the R4640 is shown in figure 3. The 4 GB address space is divided into addresses accessible in either kernel or user mode (kuseg), and addresses only accessible in kernel mode (kseg2:0).

The R4640 supports the use of multiple user tasks sharing common virtual addresses, but mapped to separate physical addresses. This facility is implemented via the "base-bounds" registers contained in CP0.

When a user virtual address is asserted (load, store, or instruction fetch), the R4640 compares the virtual address with the contents of the appropriate "bounds" register (instruction or data). If the virtual address is "in bounds", the value of the corresponding "base" register is added to

the virtual address to form the physical address for that reference. If the address is not within bounds, an exception is signalled.

This facility enables multiple user processes in a single physical memory without the use of a TLB. This type of operation is further supported by a number of development tools for the R4640, including real-time operating systems and "position independent code".

Kernel mode addresses do not use the base-bounds registers, but rather undergo a fixed virtual to physical address translation.

Debug Support

To facilitate software debug, the R4640 adds a pair of "watch" registers to CP0. When enabled, these registers will cause the CPU to take an exception when a "watched" address is appropriately accessed.

Interrupt Vector

The R4640 also adds the capability to speed interrupt exception decoding. Unlike the R4700, which utilizes a single common exception vector for all exception types (including interrupts), the R4640 allows kernel software to enable a separate interrupt exception vector. When enabled, this vector location speeds interrupt processing by allowing software to avoid decoding interrupts from general purpose exceptions.

Cache Memory

In order to keep the R4640's high-performance pipeline full and operating efficiently, the R4640 incorporates on-chip instruction and data caches that can each be accessed in a single processor cycle. Each cache has its own 64-bit data path and can be accessed in parallel. The cache subsystem provides the integer and floating-point units with an aggregate bandwidth of over 1500 MB per second at a pipeline clock frequency of 133MHz. The cache subsystem is similar in construction to that found in the R4600, although some changes have been implemented. Table 6 is an overview of the caches found on the R4640.

Instruction Cache

The R4640 incorporates a two-way set associative on-chip instruction cache. This virtually indexed, physically tagged cache is 8KB in size and is parity protected.

Because the cache is virtually indexed, the virtual-to-physical address translation occurs in parallel with the cache access, thus further increasing performance by allowing these two operations to occur simultaneously. The tag holds a 20-bit physical address and valid bit, and is parity protected.

The instruction cache is 64-bits wide, and can be refilled or accessed in a single processor cycle. Instruction fetches require only 32 bits per cycle, for a peak instruction bandwidth of 533MB/sec at 133MHz. Sequential accesses take advantage of the 64-bit fetch to

reduce power dissipation, and cache miss refill, can write 64 bits-per-cycle to minimize the cache miss penalty. The line size is eight instructions (32 bytes) to maximize performance.

In addition, the contents of one set of the instruction cache (set "A") can be "locked" by setting a bit in a CP0 register. Locking the set prevents its contents from being overwritten by a subsequent cache miss; refill occurs then only into "set B".

This operation effectively "locks" time critical code into one 4kB set, while allowing the other set to service other instruction streams in a normal fashion. Thus, the benefits of cached performance are achieved, while deterministic real-time response is preserved.

Data Cache

For fast, single cycle data access, the R4640 includes an 8KB on-chip data cache that is two-way set associative with a fixed 32-byte (eight words) line size. Table 4 lists the R4640 cache attributes.

Characteristics	Instruction	Data
Size	8KB	8KB
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.	writeback / writethru
Line transfer order	read sub-block order	read sub-block order
	write sequential	write sequential
Miss restart after transfer of	entire line	first word
Parity	per-word	per-byte
Cache locking	set A	set A

Table 4: R4640 Cache Attributes

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 writeback, 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 for certain address ranges, using the CAI_g register in CP0. Cache protocols supported for the data cache are:

- **Uncached.** Addresses in a memory area indicated as uncached will not be read from the cache. Stores to such addresses will be written directly to main memory,

without changing cache contents.

- **Writeback.** 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 see if the target address is cache resident. If it is resident, the cache contents will be updated, and the cache line marked for later writeback. If the cache lookup misses, the target line is first brought into the cache before the cache is updated.
- **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 see if the target address is cache resident. If it is resident, the cache contents will be updated and main memory will also be written; the state of the "writeback" bit of the cache line will be unchanged. If the cache lookup misses, the target line is first brought into the cache before the cache is updated.
- **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 see if the target address is cache resident. If it is resident, the cache contents will be updated, and the cache line marked for later writeback. If the cache lookup misses, then only main memory is written. Associated with the Data Cache is the store buffer. When the R4640 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 R4640 to execute a store every processor cycle and to perform back-to-back stores without penalty.

Write buffer

Writes to external memory, whether cache miss writebacks or stores to uncached or write-through addresses, use the on-chip write buffer. The write buffer holds up to four address and data pairs. The entire buffer is used for a data cache writeback and allows the processor to proceed in parallel with memory update. For uncached and write-through stores, the write buffer significantly increases performance over the R4000 family of processors.

System Interface

The R4640 supports a 64-bit system interface that is bus compatible with the R4700 system interface. In addition, the R4640 supports a 32-bit system interface mode, allowing the CPU to interface directly with a lower cost memory system.

The interface consists of a 64-bit Address/Data bus with 8 check bits and a 9-bit command bus protected with parity. In addition, there are 8 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 533MB/sec at 133MHz.

Figure 4 shows a typical system using the R4640. In this example two banks of DRAMs are used to supply and accept data with a DDxxDD data pattern.

The R4640 clocking interface allows the CPU to be easily mated with external reference clocks. The CPU input clock is the bus reference clock, and can be between 25 and 67MHz (somewhat dependent on maximum pipeline speed for the CPU).

An on-chip phase-locked-loop generates the pipeline clock from the system interface clock by multiplying it up an amount selected at system reset. Supported multipliers are values 2 through 8 inclusive, allowing systems to implement pipeline clocks at significantly higher frequency than the system interface clock.

System Address/Data Bus

The 64-bit System Address Data (SysAD) bus is used to transfer addresses and data between the R4640 and the rest of the system. It is protected with an 8-bit parity check bus, SysADC. When initialized for 32-bit operation, SysAD can be viewed as a 32-bit multiplexed bus, with 4 parity check bits.

The system interface is configurable to allow easier interfacing to memory and I/O systems of varying frequencies. The bus frequency and reference timing of

the R4640 are taken from the input clock. The rate at which the CPU transmits data to the system interface is programmable via boot time mode control bits. 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 R4640. Again, the system designer has the flexibility to make these price/performance trade-offs.

System Command Bus

The R4640 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 cache state of this data line is clean exclusive). The SysCmd bus is bidirectional to support both processor requests and external requests to the R4640. Processor requests are initiated by the R4640 and responded to by an external device. External requests are issued by an external device and require the R4640 to respond.

The R4640 supports single datum (one to eight byte) and 8-word block transfers on the SysAD bus. In the case of a single-datum transfer, the low-order 3 address bits gives the byte address of the transfer, and the SysCmd bus indicates the number of bytes being transferred. The choice of 32- or 64-bit wide system interface dictates whether a cache line block transaction requires 4 double

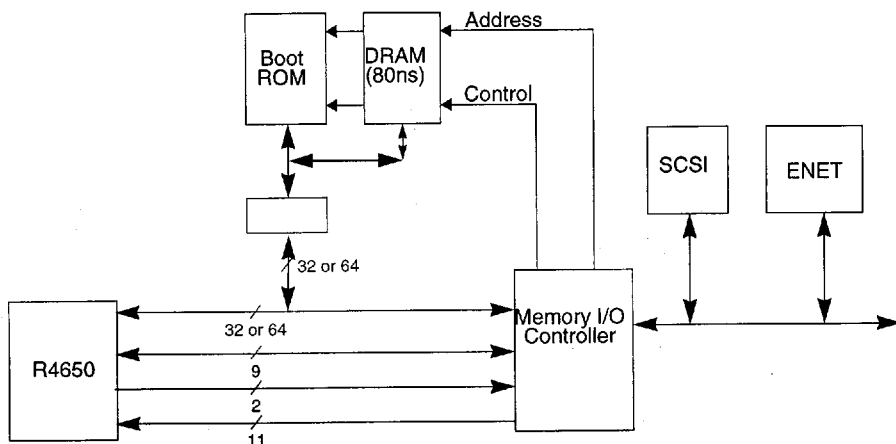


Figure 4: Typical R4650 System Architecture

word data cycles or 8 single word cycles, and whether a single datum transfer larger than 4 bytes needs to be broken into two smaller transfers.

Handshake Signals

There are six handshake signals on the system interface. Two of these, $\overline{\text{RdRdy}}$ and $\overline{\text{WrRdy}}$ are used by an external device to indicate to the R4640 whether it can accept a new read or write transaction. The R4640 samples these signals before deasserting the address on read and write requests.

$\overline{\text{ExtRqst}}$ and $\overline{\text{Release}}$ are used to transfer control of the SysAD and SysCmd buses between the processor and an external device. When an external device needs to control the interface, it asserts $\overline{\text{ExtRqst}}$. The R4640 responds by asserting $\overline{\text{Release}}$ to release the system interface to slave state.

$\overline{\text{ValidOut}}$ and $\overline{\text{ValidIn}}$ are used by the R4640 and the external device respectively to indicate that there is a valid command or data on the SysAD and SysCmd buses. The R4640 asserts $\overline{\text{ValidOut}}$ when it is driving these buses with a valid command or data, and the external device drives $\overline{\text{ValidIn}}$ when it has control of the buses and is driving a valid command or data.

Non-overlapping System Interface

The R4640 requires a non-overlapping system interface, compatible with the R4700. 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 R4640 issues another request. The R4640 can issue read and write requests to an external device, and an external device can issue read and write requests to the R4640.

The R4640 asserts $\overline{\text{ValidOut}}$ and simultaneously drives the address and read command on the SysAD and SysCmd buses. If the system interface has $\overline{\text{RdRdy}}$ or Read transactions asserted, then the processor tristates its drivers and releases the system interface to slave state by asserting $\overline{\text{Release}}$. The external device can then begin sending the data to the R4640.

Figure 5 shows a processor block read request and the external agent read response. The read latency is 4 cycles ($\overline{\text{ValidOut}}$ to $\overline{\text{ValidIn}}$), and the response data pattern is DDxxDD.

Figure 6 shows a processor block write.

Write Reissue and Pipeline Write

The R4700 and the R4640 implement additional write protocols designed to improve performance. This implementation doubles the effective write bandwidth. The write re-issue has a high repeat rate of 2 cycles per write. A write issues if $\overline{\text{WrRdy}}$ is asserted 2 cycles earlier and is still asserted at the issue cycle. If it is not still asserted, the last write re-issues again. Pipelined writes have the same 2-cycle per write repeat rate, but can issue one

more write after $\overline{\text{WrRdy}}$ de-asserts. They still follow the issue rule as R4x00 mode for other writes.

External Requests

The R4640 responds to requests issued by an external device. The requests can take several forms. An external device may need to supply data in response to an R4640 read request or it may need to gain control over the system interface bus to access other resources which may be on that bus.

The following is a list of the supported external requests:

- Read Response
- Null

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 (MasterClock 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 or a simple PAL.

Immediately after the $\overline{\text{Vccok}}$ Signal is asserted, the processor reads a serial bit stream of 256 bits to initialize all fundamental operational modes. After initialization is complete, the processor continues to drive the serial clock output, but no further initialization bits are read.

Boot-Time Modes

The boot-time serial mode stream is defined in Table 5. Bit 0 is the bit presented to the processor when $\overline{\text{Vccok}}$ is asserted; bit 255 is the last.

Power Management

CP0 is also used to control the power management for the R4640. This is the standby mode and it can be used to reduce the power consumption of the internal core of the CPU. The standby mode is entered by executing the WAIT instruction with the SysAD bus idle and is exited by any interrupt.

Standby Mode Operation

The R4640 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 is known as "Standby Mode".

Entering Standby Mode

Executing the WAIT instruction enables interrupts and enters Standby mode. When the WAIT instruction finishes the W pipe-stage, if the SysAd bus is currently idle, the internal clocks will shut down, thus freezing the pipeline. The PLL, internal timer, and some of the input pins (Int[5:0]*, NMI*, ExtReq*, Reset*, and ColdReset*) will continue to run. If the conditions are not correct when the

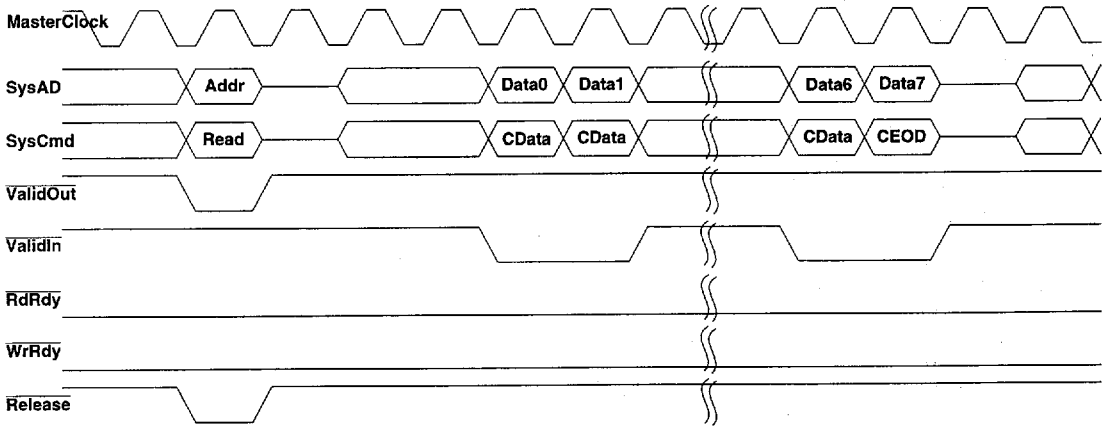


Figure 5: R4640 Block Read Request

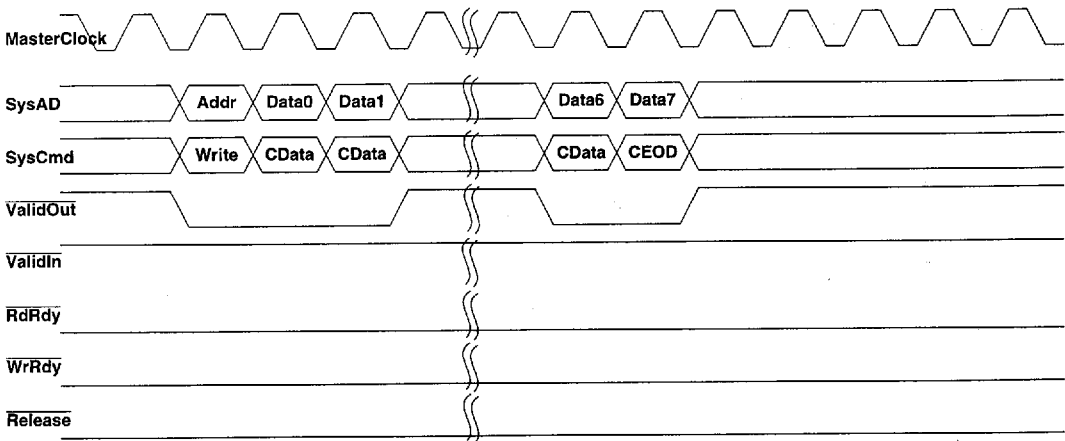


Figure 6: R4640 Block Write Request

WAIT instruction finishes the W pipe-stage (i.e. the SysAd bus is not idle), the WAIT is treated as a NOP.

Once the CPU is in Standby Mode, any interrupt, including the internally generated timer interrupt, will cause the CPU to exit Standby Mode.

Mode bit	Description
0	Reserved (must be zero)
4..1	Writeback data rate: 32-bit 0 → W 1 → WWx 2 → WWxx 3 → WxWx 4 → WWxxx 5 → WWxxxx 6 → WxxWxx 7 → WWxxxxx 8 → WxxxWxxx 9-15 reserved
7..5	Clock multiplier: 0 → 2 1 → 3 2 → 4 3 → 5 4 → 6 5 → 7 6 → 8 7 reserved
8	0 → Little endian 1 → Big endian
10..9	00 → R4000 compatible 01 → reserved 10 → pipelined writes 11 → write re-issue
11	Disable the timer interrupt on Int[5]
12	Must be 1
14..13	Output driver strength: 10 → 100% strength (fastest) 11 → 83% strength 00 → 67% strength 01 → 50% strength (slowest)
255..15	Must be zero

Table 5: Boot time mode stream

Thermal Considerations

The R4640 utilizes special packaging techniques to improve the thermal properties of high-speed processors. The R4640 is packaged using cavity up packaging in a 128-pin PQFP package for devices with low peak power. The R4640 also utilizes the MQAD package for higher power consumption devices (the "MU" package), which is an all-aluminum package with the die attached to a normal copper lead frame mounted to the aluminum casing.

Due to the heat-spreading effect of the aluminum, the MQAD package allows for an efficient thermal transfer between the die and the case. The aluminum offers less internal resistance from one end of the package to the other, reducing the temperature gradient across the package and therefore presenting a greater area for convection and conduction to the PCB for a given temperature. Even nominal amounts of airflow will dramatically reduce the junction temperature of the die, resulting in cooler operation. The MQAD package is pin and socket compatible with the 128-pin PQFP package.

The R4640 is guaranteed in a case temperature range of 0° to +85° C. The type of package, speed (power) of the device, and airflow conditions affect the equivalent ambient temperature conditions that will meet this specification.

The equivalent allowable ambient temperature, T_A , can be calculated using the thermal resistance from case to ambient (θ_{CA}) of the given package. The following equation relates ambient and case temperatures:

$$T_A = T_C - P * \theta_{CA}$$

where P is the maximum power consumption at hot temperature, calculated by using the maximum ICC specification for the device.

Typical values for θ_{CA} at various airflows are shown in Table 6.

		θ_{CA}					
		0	200	400	600	800	1000
Airflow (ft/min)		41	30	26	23	22	21
128 PQFP		21	13	10	9	8	7
128 MQAD							

Table 6: Thermal Resistance (θ_{CA}) at Various Airflows

Note that the R4640 implements advanced power management to substantially reduce the average power dissipation of the device. This operation is described in the *IDT79R4640/ IDT79R4640 RISC Processor Hardware User's Manual*.

DATA SHEET REVISION HISTORY

Changes to version dated December 1995:

Features:

- Added 32-bit bus interface info
- Deleted items from low-power operation descriptions.

Hardware Overview:

- Added detailed descriptions of features.
- Changed Boot Time Mode Stream table values for mode bit 12.

DC Electrical Characteristics:

- The C_{IN} and C_{OUT} values have been changed.

AC Electrical Characteristics:

- In System Interface Parameters tables (R4640 and RV4640), Data Setup and Data Hold minimums changed.

Valid Combinations::

- List of valid combinations has been corrected.

PIN DESCRIPTION

The following is a list of interface, interrupt, and miscellaneous pins available on the R4640. Pin names ending with an asterisk (*) identify pins that are active when low.

Pin Name	Type	Description
System Bus Interface		
ExtRqst*	Input	External request Signals that the system interface needs to submit 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(31:0)	Input/Output	System address/data bus A 32-bit address and data bus for communication between the processor and an external agent.
SysADC(3:0)	Input/Output	System address/data check bus A 4-bit bus containing parity check bits for the SysAD bus during data bus 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 system command/data identifier bus parity For the R4640 this signal is unused on input and zero on output.
Clock/Control interface		
MasterClock	Input	Master 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.

5

Pin Name	Type	Description
Interrupt interface		
Int*(5:0)	Input	Interrupt Six general processor interrupts, bit-wise OR' d with bits 5:0 of the interrupt register.
NMI*	Input	Non-maskable interrupt Non-maskable interrupt, OR'd with bit 6 of the interrupt register.
Initialization interface		
Vccok	Input	Vcc is OK When asserted, this signal indicates to the R4640 that the 3.3V (5.0V) power supply has been above 3.0V (4.5V) 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 MasterClock.
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 MasterClock.
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.

AC ELECTRICAL CHARACTERISTICS — COMMERCIAL TEMPERATURE RANGE—R4640 $(V_{CC}=5.0V \pm 5\%; T_{CASE} = 0^{\circ}C \text{ to } +85^{\circ}C)$ **Clock Parameters—R4640**

Parameter	Symbol	Test Conditions	R4640 80MHz		R4640 100MHz		R4640 133MHz		Units
			Min	Max	Min	Max	Min	Max	
Pipeline clock frequency	PClk		50	80	50	100	50	133	MHz
MasterClock HIGH	t_{MCHIGH}	Transition $\leq 5ns$	6	—	4	—	3	—	ns
MasterClock LOW	t_{MCLOW}	Transition $\leq 5ns$	6	—	4	—	3	—	ns
MasterClock Frequency	—	—	25	40	25	50	25	67	MHz
MasterClock Period	t_{MCP}	—	25	40	20	40	15	40	ns
Clock Jitter for MasterClock	$t_{jitterIn}$	—	—	± 250	—	± 250	—	± 250	ps
MasterClock Rise Time	t_{MCRise}	—	—	5	—	5	—	4	ns
MasterClock Fall Time	t_{MCFall}	—	—	5	—	5	—	4	ns
ModeClock Period	$t_{ModeCKP}$	—	—	256* t_{MCP}	—	256* t_{MCP}	—	256* t_{MCP}	ns

System Interface Parameters—R4640

Parameter	Symbol	Test Conditions	R4640 80MHz		R4640 100MHz		R4640 133MHz		Units
			Min	Max	Min	Max	Min	Max	
Data Output	$t_{DM} = \text{Min}$ $t_{DO} = \text{Max}$	mode _{14..13} = 10 (fastest)	1.0	11	1.0	9	1.0	9	ns
		mode _{14..13} = 01 (slowest)	2.0	15	2.0	12	2.0	12	ns
Data Setup	t_{DS}	$t_{rise} = 5ns$ $t_{fall} = 5ns$	6.5	—	5.5	—	4.5	—	ns
Data Hold	t_{DH}		3	—	2	—	1.5	—	ns

Boot Time Interface Parameters—R4640

Parameter	Symbol	Test Conditions	R4640 80MHz		R4640 100MHz		R4640 133MHz		Units
			Min	Max	Min	Max	Min	Max	
Mode Data Setup	t_{DS}	—	3	—	3	—	3	—	Master Clock Cycle
Mode Data Hold	t_{DH}	—	0	—	0	—	0	—	Master Clock Cycle

DC ELECTRICAL CHARACTERISTICS — COMMERCIAL TEMPERATURE RANGE—R4640

 $(V_{CC} = 5.0 \pm 5\%, T_{CASE} = 0^\circ\text{C to } +85^\circ\text{C})$

Parameter	R4640 80MHz		R4640 100MHz		R4640 133MHz		Conditions
	Minimum	Maximum	Minimum	Maximum	Minimum	Maximum	
V_{OL}	—	0.1V	—	0.1V	—	0.1V	$ I_{OUT} = 20\mu\text{A}$
V_{OH}	$V_{CC} - 0.1\text{V}$	—	$V_{CC} - 0.1\text{V}$	—	$V_{CC} - 0.1\text{V}$	—	
V_{OL}	—	0.4V	—	0.4V	—	0.4V	$ I_{OUT} = 4\text{mA}$
V_{OH}	3.5V	—	2.4V	—	2.4V	—	
V_{IL}	-0.5V	0.8V	-0.5V	$0.2V_{CC}$	-0.5V	$0.2V_{CC}$	—
V_{IH}	2.0V	$V_{CC} + 0.5\text{V}$	2.0V	$V_{CC} + 0.5\text{V}$	2.0V	$V_{CC} + 0.5\text{V}$	—
I_{IN}	—	$\pm 10\mu\text{A}$	—	$\pm 10\mu\text{A}$	—	$\pm 10\mu\text{A}$	$0 \leq V_{IN} \leq V_{CC}$
C_{IN}	—	15pF	—	15pF	—	15pF	—
C_{OUT}	—	15pF	—	15pF	—	15pF	—
I/O_{LEAK}	—	20 μA	—	20 μA	—	20 μA	Input/Output Leakage

Power Consumption—R4640

Parameter	R4640 80MHz		R4640 100MHz		R4640 133MHz		Conditions	
	Typical	Max	Typical	Max	Typical	Max		
System Condition:	80/40MHz		100/50MHz		133/44MHz		—	
I_{CC}	standby	—	50 mA	—	75 mA	—	100 mA	$C_L = 0\text{pF}$
		—	125 mA	—	150 mA	—	200 mA	$C_L = 50\text{pF}$
	active	575 mA	800 mA	700 mA	1000 mA	950 mA	1350 mA	$C_L = 0\text{pF}$ No SysAd activity
		625 mA	1000 mA	750 mA	1200 mA	1000 mA	1550 mA	$C_L = 50\text{pF}$ R4x00 compatible writes, $T_C = 25^\circ\text{C}$
		625 mA	1100 mA	750 mA	1350 mA	1000 mA	1650 mA	$C_L = 50\text{pF}$ Pipelined writes or write re-issue, $T_C = 25^\circ\text{C}$

AC ELECTRICAL CHARACTERISTICS — COMMERCIAL TEMPERATURE RANGE—RV4640 $(V_{CC}=3.3V \pm 5\%; T_{CASE} = 0^{\circ}C \text{ to } +85^{\circ}C)$ **Clock Parameters—RV4640**

Parameter	Symbol	Test Conditions	RV4640 80MHz		RV4640 100MHz		RV4640 133MHz		Units
			Min	Max	Min	Max	Min	Max	
Pipeline clock frequency	PClk		50	80	50	100	50	133	MHz
MasterClock HIGH	t_{MCHIGH}	Transition $\leq 5ns$	6	—	4	—	3	—	ns
MasterClock LOW	t_{MCLOW}	Transition $\leq 5ns$	6	—	4	—	3	—	ns
MasterClock Frequency	—	—	20	40	25	50	25	67	MHz
MasterClock Period	t_{MCP}	—	25	40	20	40	15	40	ns
Clock Jitter for MasterClock	$t_{Jitterin}$	—	—	± 250	—	± 250	—	± 250	ps
MasterClock Rise Time	t_{MCRise}	—	—	5	—	5	—	4	ns
MasterClock Fall Time	t_{MCFall}	—	—	5	—	5	—	4	ns
ModeClock Period	$t_{ModeCKP}$	—	—	$256^* t_{MCP}$	—	$256^* t_{MCP}$	—	$256^* t_{MCP}$	ns

Note: Operation of the RV4640 is only guaranteed with the Phase Lock Loop enabled.

System Interface Parameters—RV4640

Parameter	Symbol	Test Conditions	RV4640 80MHz		RV4640 100MHz		RV4640 133MHz		Units
			Min	Max	Min	Max	Min	Max	
Data Output	$t_{DM} = \text{Min}$ $t_{DO} = \text{Max}$	mode _{14..13} = 10 (fastest)	1.0	11	1.0	9	1.0	9	ns
		mode _{14..13} = 01 (slowest)	2.0	15	2.0	12	2.0	12	ns
Data Output Hold	t_{DOH}	mode _{14..13} = 10 (fastest)	1.0	—	1.0	—	1.0	—	ns
Data Setup	t_{DS}	$t_{rise} = 3ns$	6.5	—	5.5	—	4.5	—	ns
Data Hold	t_{DH}	$t_{fall} = 3ns$	3	—	2	—	1.5	—	ns

Note: * 25pf loading on external output signals, fastest settings

Boot Time Interface Parameters—RV4640

Parameter	Symbol	Test Conditions	RV4640 80MHz		RV4640 100MHz		RV4640 133MHz		Units
			Min	Max	Min	Max	Min	Max	
Mode Data Setup	t_{DS}	—	3	—	3	—	3	—	Master Clock Cycle
Mode Data Hold	t_{DH}	—	0	—	0	—	0	—	Master Clock Cycle

DC ELECTRICAL CHARACTERISTICS — COMMERCIAL TEMPERATURE RANGE—RV4640

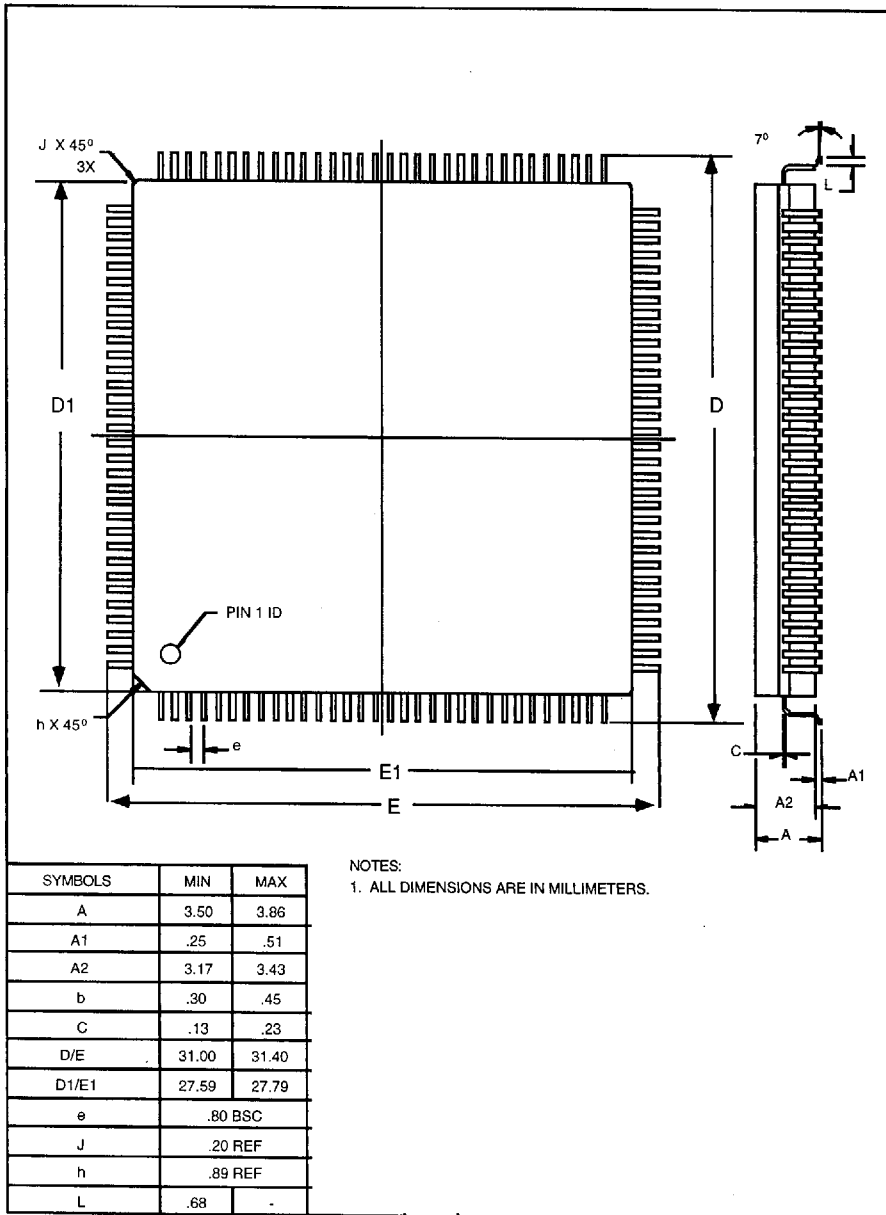
 $(V_{CC} = 3.3 \pm 5\%, T_{CASE} = 0^\circ\text{C to } +85^\circ\text{C})$

Parameter	RV4640 80MHz		RV4640 100MHz		RV4640 133MHz		Conditions
	Minimum	Maximum	Minimum	Maximum	Minimum	Maximum	
V_{OL}	—	0.1V	—	0.1V	—	0.1V	$ I_{OUT} = 20\mu\text{A}$
V_{OH}	$V_{CC} - 0.1\text{V}$	—	$V_{CC} - 0.1\text{V}$	—	$V_{CC} - 0.1\text{V}$	—	
V_{OL}	—	0.4V	—	0.4V	—	0.4V	$ I_{OUT} = 4\text{mA}$
V_{OH}	2.4V	—	2.4V	—	2.4V	—	
V_{IL}	-0.5V	$0.2V_{CC}$	-0.5V	$0.2V_{CC}$	-0.5V	$0.2V_{CC}$	—
V_{IH}	$0.7V_{CC}$	$V_{CC} + 0.5\text{V}$	$0.7V_{CC}$	$V_{CC} + 0.5\text{V}$	$0.7V_{CC}$	$V_{CC} + 0.5\text{V}$	—
V_{OHC}	—	—	—	—	—	—	—
V_{ILC}	—	—	—	—	—	—	—
V_{IHC}	—	—	—	—	—	—	—
C_{IN}	—	15pF	—	15pF	—	15pF	—
C_{OUT}	—	15pF	—	15pF	—	15pF	—
I/O_{LEAK}	—	20 μA	—	20 μA	—	20 μA	Input/Output Leakage

POWER CONSUMPTION—RV4640

Parameter	RV4640 80MHz		RV4640 100MHz		RV4640 133MHz		Conditions	
	Typical	Maximum	Typical	Maximum	Typical	Maximum		
System Condition:	80/40MHz		100/50MHz		133/44MHz		—	
I_{CC}	standby	—	40 mA	—	50 mA	—	60 mA	$C_L = 0\text{pF}$
		—	90mA	—	100 mA	—	110 mA	$C_L = 50\text{pF}$
	active	375 mA	575mA	475 mA	700 mA	625 mA	925 mA	$C_L = 0\text{pF}$ No SysAd activity
		400 mA	700 mA	525 mA	825 mA	650 mA	1050 mA	$C_L = 50\text{pF}$ R4x00 compatible writes, $T_C = 25^\circ\text{C}$
		400 mA	775 mA	525 mA	825 mA	650 mA	1125 mA	$C_L = 50\text{pF}$ Pipelined writes or Write re-issue, $T_C = 25^\circ\text{C}$

PHYSICAL SPECIFICATIONS – 128-Pin MQUAD/ PQFP



5

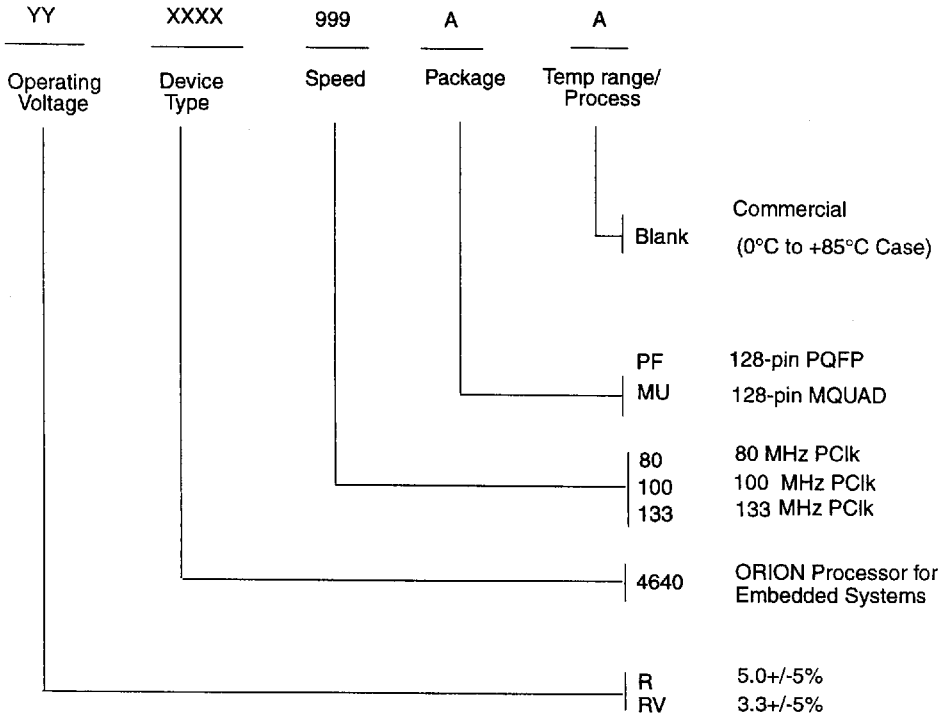
R4640-32 PQFP/MQuad Package Pin-Out

N.C. pins should be left floating for maximum flexibility as well as for compatibility with future designs. An asterisk (*) identifies a pin that is active when low.

Pin	Function	Pin	Function	Pin	Function	Pin	Function
1	N.C.	33	Vcc	65	Vcc	97	Vcc
2	SysCmd2	34	Vss	66	SysAD28	98	Vss
3	Vcc	35	SysAD13	67	ColdReset*	99	SysAD19
4	Vss	36	SysAD14	68	SysAD27	100	ValidIn*
5	SysAD5	37	Vss	69	Vss	101	Vcc
6	WrRdy*	38	Vcc	70	Vcc	102	Vss
7	ModeClock	39	SysAD15	71	N.C.	103	SysAD18
8	SysAD6	40	Vss	72	SysAD26	104	Int0*
9	Vcc	41	Vcc	73	N.C.	105	SysAD17
10	Vss	42	SysADC1	74	Vss	106	Vcc
11	SysCmd3	43	Vss	75	Vcc	107	Vss
12	SysAD7	44	Vcc	76	SysAD25	108	Int1*
13	SysCmd4	45	MasterClock	77	Vss	109	SysAD16
14	Vcc	46	VssP	78	Vcc	110	Int2*
15	Vss	47	VccP	79	SysAD24	111	Vcc
16	SysADC0	48	Vss	80	SysADC2	112	Vss
17	SysCmd5	49	Vss	81	Vss	113	Int3*
18	SysAD8	50	Vss	82	Vcc	114	SysAD0
19	Vcc	51	Vss	83	NMI*	115	Int4*
20	Vss	52	Vss	84	SysAD23	116	Vcc
21	SysCmd6	53	Vss	85	Release*	117	Vss
22	SysAD9	54	SysADC3	86	Vss	118	SysAD1
23	Vcc	55	VccOK	87	Vcc	119	Int5*
24	Vss	56	Vss	88	SysAD22	120	SysAD2
25	SysCmd7	57	Vcc	89	ModeIn	121	Vcc
26	SysAD10	58	SysAD31	90	RdRdy*	122	Vss
27	SysCmd8	59	Vss	91	SysAD21	123	SysCmd0
28	Vcc	60	Vcc	92	Vss	124	SysAD3
29	Vss	61	SysAD30	93	Vcc	125	Vcc
30	SysAD11	62	SysAD29	94	ExtRqst*	126	Vss
31	SysCmdP	63	Reset*	95	SysAD20	127	SysCmd1
32	SysAD12	64	Vss	96	ValidOut*	128	SysAD4

ORDERING INFORMATION

IDT79



5

VALID COMBINATIONS

- IDT79R4640 - 80, 100, 133MHz MQAD package
- IDT79RV4640 - 80, 100, 133MHz PQFP package
- IDT79RV4640 - 80, 100, 133MHz MQAD package