Toshiba TC86R10000 MIPS RISC Microprocessor

Toshiba R10000 Hits Power-Performance Max

Toshiba is the leading silicon vendor licensed to develop, manufacture and sell MIPS RISC or Reduced Instruction Set Computing products. Starting with the R3000, Toshiba has long been a foundry for MIPS RISC products including the R4000, R4400, R4600 and R8000 processors. As a key supplier of RISC technology for the computer system and embedded control markets, Toshiba now adds the R10000 superscalar processor to its family of high performance RISC products.

The R10000 is a 64-bit processor designed for demanding, high performance applications. It provides the highest integer and floating point performance available in a microprocessor, and runs on both UNIX and Windows NT operating systems.

The R10000 is ideal for applications and products which depend on floating point functions such as those that require single and double precision, scalar and irregular vector computing. It is uniquely suited for graphics rendering, modeling, visualization, and simulation products where real-time and near-real-time response is essential.

Designers and developers can deploy the R10000 and quickly realize its performance advantages. To protect and extend your investment in product development, the R10000 is fully compatible with previous-generation MIPS RISC processors. Even without optimization for the R10000, applications can realize performance gains of up to three times. Products designed to leverage the power and performance potential of the R10000 will be industry leaders in speed and function.

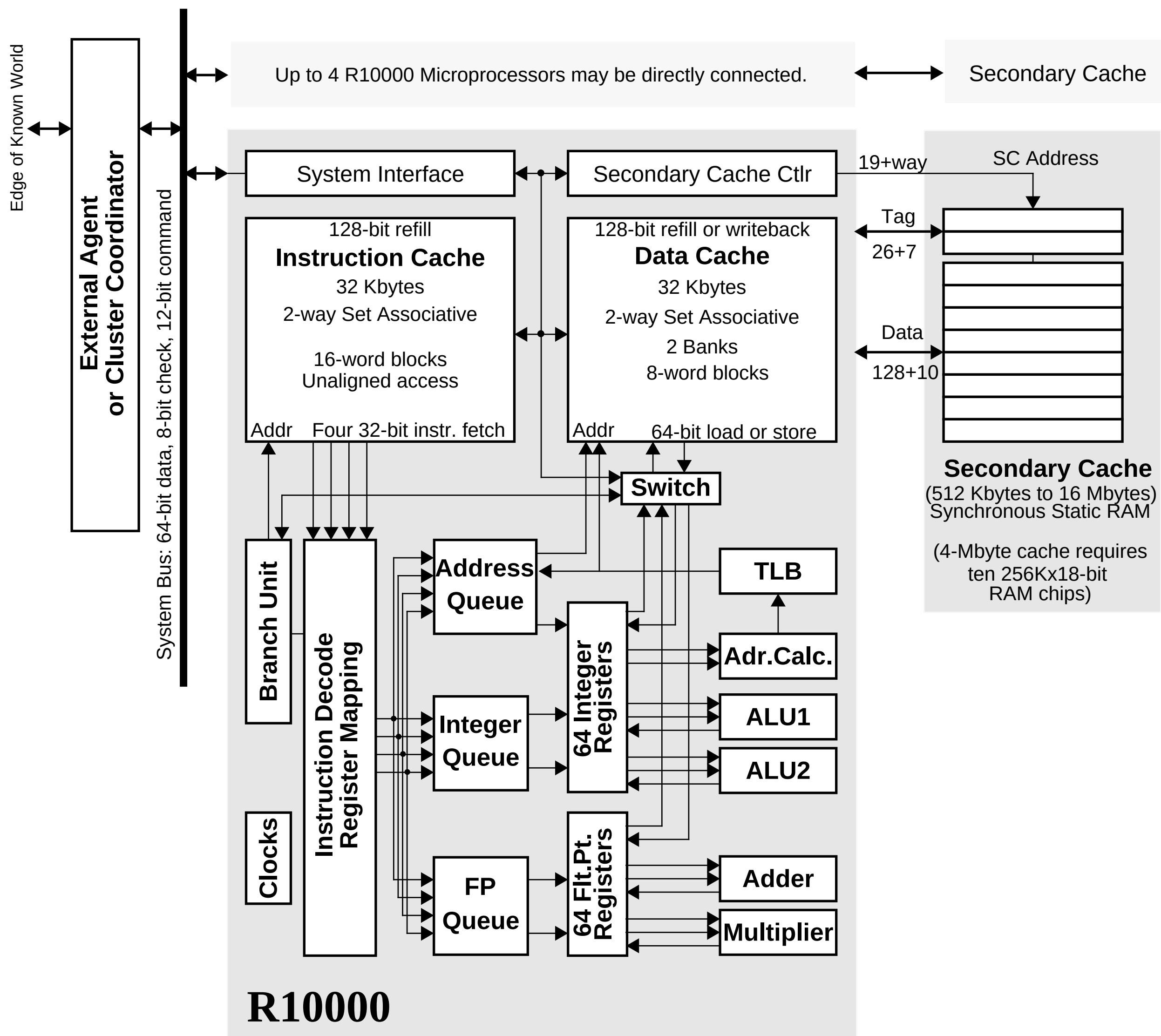


Figure 1. Block Diagram of the R10000 Processor

Next Generation Capabilities Give Immediate Benefits

Although the R10000 is fully compatible with earlier versions of MIPS RISC processors, it differs from these processors in many ways. As expected of next-generation products, the best features are retained while additional, more powerful attributes are incorporated. The R10000 is no exception. System designers have considerably more options and performance potential when implementing the R10000.

Where previous MIPS processor performance was described as “almost one instruction per cycle”, the superscalar R10000 can complete as many as five instructions per cycle. It has five independent execution units or pipelines, each of which are optimized for specific classes of MIPS-IV instructions. With its increase in instruction throughput comes more sophisticated mechanisms for handling resource conflicts and on-chip competition. For example, the R10000 uses dynamic instruction scheduling, out-of-order execution, and speculative branching. Perhaps more importantly, however, the R10000 has separate integer and floating point units, each with dedicated queues, for maximum computational performance.

To reduce bottlenecks and delays, the R10000 has a large primary cache. Its non-blocking mechanism and 32KB of instruction and data cache ensure that the CPU will not sit idle waiting for data. There is also an independent secondary cache controller for 512KB to 16MB of synchronous cache to offer system designers a full range of options for product design.

The result is a best-of-class processor which incorporates both high-speed execution and conflict anticipation/resolution features. That means that next-generation applications and products will face fewer bottlenecks and less performance degradation. Products will perform both faster and with more functionality; as the R10000 achieves greater on-chip integration, software resources once required to perform or compensate for these functions are free to be applied to broader functional and performance enhancements..

Superscalar Architecture Offers Greatest Throughput

As a superscalar device, the R10000 has five separate execution pipelines of seven stages each.The R10000, in contrast to previous generation processors, fetches and decodes four instructions in parallel each cycle or stage. Quite simply, the R10000 executes more instructions per cycle, which increases its performance and the systems’ throughput. Another way that the R10000 execution performance is improved is through the use of dedicated pipelines and instruction queues, each optimized appropriately.

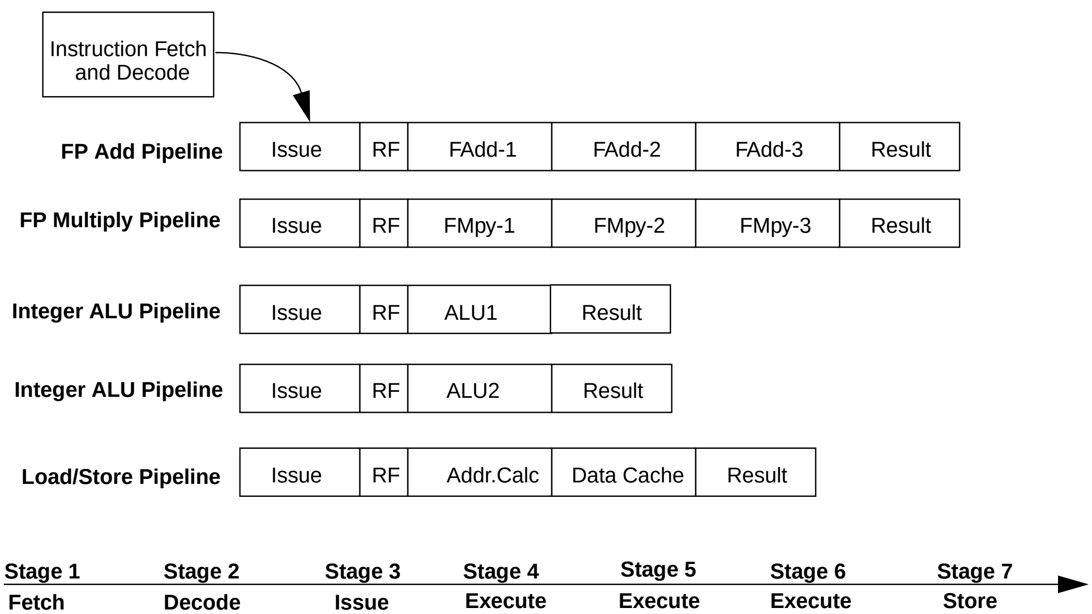


Figure 2. Superscalar Pipeline Architecture in the R10000

Instruction queues dynamically dispatch instructions into the appropriate pipeline, allowing the processor to perform at its maximum throughput without stalling because of conflicts or data dependencies. The R10000 has two integer arithmetic logic units (or ALUs) with a dedicated integer instruction queue. More complex computations are performed by three iterative units. A load/store pipeline is fed by an address queue. The floating point queue issues instructions to the floating point adder and multiplier pipelines. The fetch pipeline reads and decodes instructions from the instruction cache.

By using the MIPS-IV instruction set, the R10000 offers additional instructions to designers for anticipating data requirements and handling data dependencies. For example, the R10000 implements a prefetch instruction which allows the compiler to anticipate the need for a given block of instructions and take progressive actions. Upon execution, the block is placed in closer proximity to the processor (in secondary cache or instruction buffers). Overall product performance is enhanced by avoiding slower memory access times whenever possible.

Non-sequential Architecture Breaks Barriers

In addition to the advantages of the various functional units and prefetch instructions, the R10000 is optimized for demanding scientific and database applications where speed and high throughput are essential. With its aggressive superscalar ANDES (**A**rchitecture with **N**on-Sequential **D**ynamic **E**xecution **S**cheduling) architecture, the R10000 microprocessor breaks many of the bottlenecks that can render typical processors needlessly idle. The R10000 operates at peak efficiency by reordering instructions, allowing out-of-order execution, and predicting branches.The R10000's capacity to self-manage the flow of instructions into the pipeline ensures that applications perform faster with less overhead. See Figure 3 below.

In a typical pipeline processor, instructions are executed sequentially because each may be dependent upon the previous instruction's result. Execution of subsequent instructions—and the system software—cannot proceed until the operands are valid. Appropriately, this condition is known as a "hazard". The R10000, in contrast, allows out of order execution so that any instruction can be executed when its operands become available *regardless of its place in the sequence of instructions*. This reduced dependency on compilers and code scheduling benefits existing applications with older code. By using a mapping scheme of physical and logical registers, the R10000 is able to retain the intended sequence while exploiting non-sequential execution. (Instructions graduate in their original sequence.) Ultimately, the pipeline operates more efficiently, and system applications benefit from higher throughput and less degradation.

Another method of optimizing pipeline throughput is branch prediction. Typically, about one in six instructions is a conditional branch. Older processors sometimes stalled until they determined whether to take a branch. Worse, some falter upon an inaccurate determination and issued instructions are aborted. Because the R10000 processes five instructions per cycle, the probability of encountering a branch is much greater. It is, therefore, optimized to achieve its full potential. The R10000 predicts the outcome of every branch and speculatively executes the branch based on the prediction. It can speculate up to four branches deep (by appending a branch predictor bit to instructions during decode). Shadow copies of the mapping tables are kept every time a prediction is made, allowing the R10000 processor to recover from a mispredicted branch in a single cycle. This prediction scheme combined with the recovery mechanism dramatically increase throughput without the jeopardies of previous branch handling schemes.

Computational Power Speeds High Tech Applications

The R10000 provides the best floating point and integer performance available in a microprocessor today. Database applications as well as decision support and technical applications can benefit from performance improvements of up to three-times current performance. Such substantial gains are possible without benefit of R10000 optimization; those applications that actually design for the R10000's computational power can realize even greater improvements.

The R10000 contains five execution units which operate independently. Two integer arithmetic logic units (ALUs) perform standard add, subtract, and logical operations in one cycle. ALU1 handles all branch and shift operations, while ALU2 handles all multiply and divide operations using iterative algorithms. (See Figure 4 on page 4)

The R10000 contains two primary floating point units (FPUs): adder unit and a multiply unit. Two secondary floating point units handle long-latency operations (such as divide and square root) to reduce overall system latency. The common floating point multiply-add operation is computed using separate multiply and add operations. By eliminating an instruction fetch and decode in this frequent computation, performance is improved. FPU speed is also increased by the floating point queue which determines data dependencies and dynamically issues appropriate instructions.

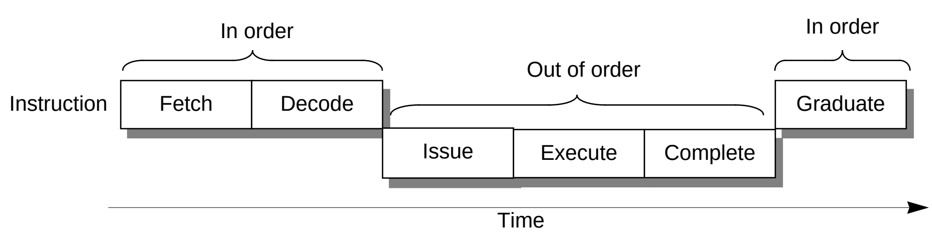


Figure 3. Dynamic Scheduling

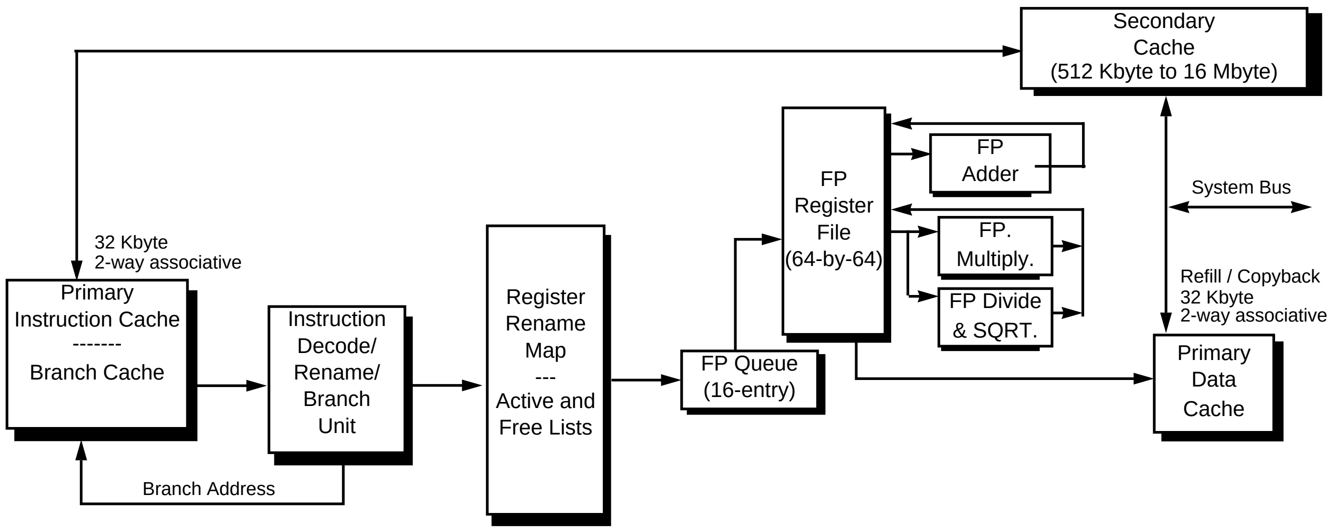


Figure 4. Logical Diagram of FP Operations

Table 1. Latencies and Repeat Rates for User Instructions

Instruction Type	Execution Unit	Latency	Repeat Rate	Comment
Integer Instructions				
Add/Sub/Logical/Set	ALU 1/2	1	1	
MF/MT HI/LO	ALU 1/2	1	1	
Shift/LUI	ALU 1	1	1	
Cond. Branch Evaluation	ALU 1	1	1	
Cond. Move	ALU 1	1	1	
MULT	ALU 2	5/6	6	Latency relative to Lo/Hi
MULTU	ALU 2	6/7	7	Latency relative to Lo/Hi
DMULT	ALU 2	9/10	10	Latency relative to Lo/Hi
DMULTU	ALU 2	10/11	11	Latency relative to Lo/Hi
DIV/DIVU	ALU 2	34/35	35	Latency relative to Lo/Hi
DDIV/DDIVU	ALU 2	66/67	67	Latency relative to Lo/Hi
Load (not include loads to CP1)	Load/Store	2	1	Assuming cache hit
Store	Load/Store	-	1	Assuming cache hit
Floating-Point Instructions				
MTC1/DMTC1	ALU 1	3	1	
Add/Sub/Abs/Neg/Round/Trunc/Ceil/Floor/C.cond	FADD	2	1	
CVT.S.W/CVT.S.L	FADD	4	2	Repeat rate is on average
CVT (others)	FADD	2	1	
Mul	FMPY	2	1	
MFC1/DMFC1	FMPY	2	1	
Cond. Move/Move	FMPY	2	1	
DIV.S/RECIP.S	FMPY	12	14	
DIV.D/RECIP.D	FMPY	19	21	
SQRT.S	FMPY	18	20	
SQRT.D	FMPY	33	35	
RSQRT.S	FMPY	30	20	
RSQRT.D	FMPY	52	35	
MADD	FADD+FMPY	2/4	1	Latency is 2 only if the result is used as the operand specified by *fr* of another MADD
LWC1/LDC1/LWXC1/LDXC1	LoadStore	3	1	Assuming cache hit

Optimized Memory Provides Fastest Performance

Memory access times lag behind the data rate requirements of high-speed processors, and this latency can degrade performance and increase system cost. The memory hierarchy of the R10000 processor mitigates this by using large set-associative caches and higher bandwidth cache refills to reduce the negative impact of loads, stores and instruction fetches.

To speed data flow, the R10000 has a large on-chip primary cache with 32KB for instructions and 32KB for data. Ideal for large database applications, the R10000 implements non-blocking caches so the processor can proceed while waiting for data. Two-way interleaving reduces latency by allowing overlapping memory access from two memory banks.

To offer greater speed and flexibility to system designers, the R10000 features an on-chip secondary cache controller for supporting 512KB to 16MB of synchronous secondary cache. Unlike shared bus systems, the R10000's dedicated secondary cache bus allows continued cache accesses to occur immediately following a cache miss. Typically, processors bog down after an on-chip cache miss as a shared bus performs other system functions. The R10000, however, continues to transfer data via its dedicated bus at over 3GB/sec.

Table 2. Secondary Cache Interface Signals

Signal Name	Description	Type
	SSRAM[a] Clock Signals	
SCClk(5:0) SCClk*(5:0)	Secondary cache clock Duplicated complementary secondary cache clock outputs.	Output
	SSRAM Address Signals	
SCAAddr(18:0 SCBAddr(18:0)	Secondary cache address bus Duplicated complementary 19-bit bus which specifies the set address of the secondary cache data and tag SSRAM that is to be accessed.	Output
SCTagLSBAddr	Secondary cache tag LSB address Signal that specifies the least significant bit of the address for the secondary cache tag SSRAM.	Output
	SSRAM Data Signals	
SCADWay SCBDWay	Secondary cache data way Duplicated signal that indicates the way of the secondary cache data SSRAM that is to be accessed.	Output
SCData(127:0)	Secondary cache data bus 128-bit bus to read/write cache data from/to secondary cache data SSRAM.	Bidirectional
SCDataChk(9:0)	Secondary cache data check bus A 10-bit bus used to read/write ECC and even parity from/to the secondary cache data SSRAM.	Bidirectional
SCADOE* SCBDOE*	Secondary cache data output enable Duplicated signal that enables the outputs of the secondary cache data SSRAM.	Output
SCADWr* SCBDWr*	Secondary cache data write enable Duplicated signal that enables writing the secondary cache data SSRAM.	Output
SCADCS* SCBDCS*	Secondary cache data chip select Duplicated signal that enables the secondary cache data SSRAM.	Output
	SSRAM Tag Signals	
SCTWay	Secondary cache tag way Signal indicating the way of the secondary cache tag SSRAM to be accessed.	Output
SCTag(25:0)	Secondary cache tag bus A 26-bit bus to read/write cache tags from/to the secondary cache tag SSRAM.	Bidirectional
SCTagChk(6:0)	Secondary cache tag check bus A 7-bit bus used to read/write ECC from/to the secondary cache tag SSRAM.	Bidirectional
SCTOE*	Secondary cache tag output enable A signal that enables the outputs of the secondary cache tag SSRAM.	Output
SCTWr*	Secondary cache tag write enable A signal that enables writing the secondary cache tag SSRAM.	Output
SCTCS*	Secondary cache tag chip select A signal which enables the secondary cache tag SSRAM.	Output

NOTE **a.** All cache static RAM (SRAM) are synchronous SRAM (SSRAM).

Configurability Increases Value

In addition to its dedicated secondary cache bus, the R10000 also supports a split-level bus transaction protocol. This protocol allows additional processor and external requests to be issued without stopping for completion of a prior transaction. To the system designer, the implications are fewer performance problems in complex applications.

Multiprocessing Made Easy

The R10000 can support two multi-processor system configurations. The first has a dedicated external agent interface to each processor; typically handled with an ASIC device as a gateway, this method has ramifications on overall system cost and complexity. The R10000 also supports a cluster bus configuration for "Glueless MP". In this configuration, a *single* external agent can interface to system resources with up to four R10000 CPUs clustered together. This method provides equivalent power at lower cost and complexity.

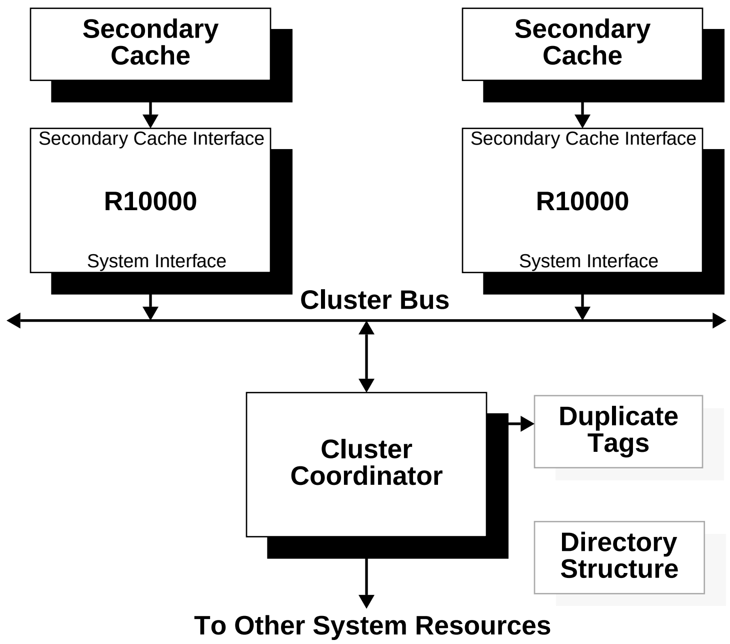


Figure 5. Multiprocessor System Organization Using Cluster Bus

Regardless of the physical configuration, the R10000 will readily meet the hardware requirements of systems with its state-of-the-art packaging. In a 599-pin ceramic land grid array (CLGA) which reduces inductance, output noise from the R10000 is minimized. The entire single-chip package is socketed for easy removal and is 47.5 millimeters square.

Table 3. Power Interface Signals

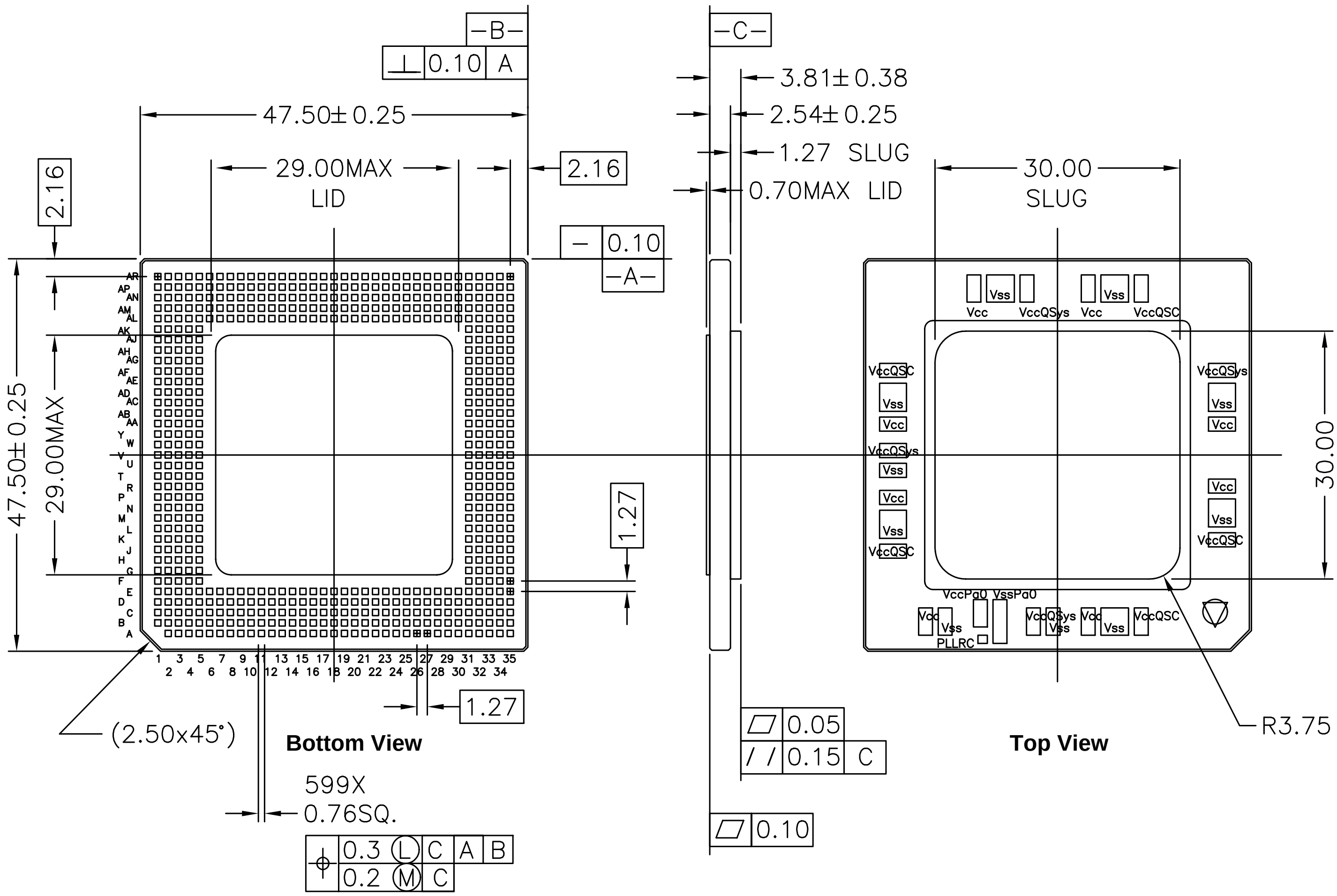
Signal Name	Description	Type
Vcc	**Vcc** core **Vcc** for the core circuits.	Input
VccQSC	**Vcc** output driver secondary cache **Vcc** for the secondary cache interface output drivers.	Input
VccQSys	**Vcc** output driver system **Vcc** for the System interface output drivers.	Input
VrefSC	Voltage reference secondary cache Voltage reference for the secondary cache interface input receivers.	Input
VrefSys	Voltage reference system Voltage reference for the System interface input receivers.	Input
VrefByp	Voltage reference bypass This pin must be tied to **VrefSys** or **Vss**.	Input
Vss	Vss **Vss** for the core circuits and output drivers.	Input
VccPa	**Vcc** PLL analog **Vcc** for the PLL analog circuits.	Input
VssPa	**Vss** PLL analog **Vss** for the PLL analog circuits.	Input
VccPd	**Vcc** PLL digital **Vcc** for the PLL digital circuits.	Input
VssPd	**Vss** PLL digital **Vss** for the PLL digital circuits.	Input
DCOk	DC voltages are OK The external agent asserts these two signals when **Vcc**, **VccQ[SC,Sys]**, **Vref[SC,Sys]**, **Vcc[Pa,Pd]**, and **SysClk** are stable.	Input

Toshiba Delivers the R10000

In a world where performance and price are the hallmarks of product success, the R10000 is simply the best choice. To meet the price/performance demands of customers, system developers must be both agile and aggressive. The R10000 meets this need by providing both immediate gains through recompiling and enormous performance improvements with optimization. Ultimately, the R10000 enables developers to provide the fastest and most functional products to the most demanding customers.

With its reputation for high quality, high volume manufacturing, Toshiba offers the latest in RISC technologies combined with excellence in manufacturing. Rigorous production quality control and monitoring coupled with a sophisticated batch tracking system provide Toshiba with the ability to meet the requirements of the most agile and aggressive companies. The variety of semiconductor products Toshiba offers can simplify component purchasing and reduce costs: as a single source for the R10000, D-RAMs, S-RAMs, ASICs, and other devices, Toshiba can meet all your component needs.

Package Outline



NOTES:
1. DIMENSIONS ARE IN MILLIMETERS
2. UNLESS OTHERWISE SPECIFIED, TOLERANCES ARE ±0.13
3. LID AND SLUG ARE CONNECTED TO VSS

RC31670396

TOSHIBA

TAEC REGIONAL SALES OFFICES

CENTRAL

Chicago, IL
TEL: (708) 945-1500
FAX: (708) 945-1044

NORTHWEST

Portland, OR
TEL: (503) 629-0818
FAX: (503) 629-0827

San Jose, CA
TEL: (408) 456-8900
FAX: (408) 456-8910

SOUTHWEST

Dallas, TX
TEL: (214) 480-0470
FAX: (214) 235-4114

Irvine, CA
TEL: (714) 453-0224
FAX: (714) 453-0125

NORTHEAST

Boston, MA
TEL: (617) 224-0074
FAX: (617) 224-1096

Edison, NJ
TEL: (908) 248-8070
FAX: (908) 248-8030

SOUTHEAST

Atlanta, GA
TEL: (770) 368-0203
FAX: (770) 368-0075

Toshiba High-End Processors

Available Products	MIPS ISA	Clock Frequency	Drystone Performance	Power (Watts)	Process (µ)	Volts	On-Chip Cache
TC86R10000YL-20B	I,II,& III	180/200MHz	-	30@200MHz	-	3.3	32kB (inst) / 32kB(data)
TC86R4300	I,II, & III	100MHz	177	1.8	0.3	3.3	16kB (inst) / 8kB (data)
TC86R4400	I,II, & III	150/200/250MHz	-	-	-	3.3	16kB (inst) / 16kB(data)
TC86R4600	I,II, & III	133MHz	-	3.8	0.4	3.3	16kB (inst) / 16kB(data)