Circuit Implementation of a 300-MHz 64-bit Second-generation CMOS Alpha CPU

by William J. Bowhill, Shane L. Bell, Bradley J. Benschneider, Andrew J. Black, Sharon M. Britton, Ruben W. Castelino, Dale R. Donchin, John H. Edmondson, Harry R. Fair, III, Paul E. Gronowski, Anil K. Jain, Patricia L. Kroesen, Marc E. Lamere, Bruce J. Loughlin, Shekhar Mehta, Robert O. Mueller, Ronald P. Preston, Sribalan Santhanam, Timothy A. Shedd, Michael J. Smith, Stephen C. Thierauf

ABSTRACT

A 300-MHz, custom 64-bit VLSI, second-generation Alpha CPU chip has been developed. The chip was designed in a 0.5-um CMOS technology using four levels of metal. The die size is 16.5 mm by 18.1 mm, contains 9.3 million transistors, operates at 3.3 V, and supports 3.3-V/5.0-V interfaces. Power dissipation is 50 W. It contains an 8-KB instruction cache; an 8-KB data cache; and a 96-KB unified second-level cache. The chip can issue four instructions per cycle and delivers 1,200 mips/600 MFLOPS (peak). Several noteworthy circuit and implementation techniques were used to attain the target operating frequency.

INTRODUCTION

The Alpha 21164 chip is a 300-megahertz (MHz), quad-issue, custom very large-scale integration (VLSI) implementation of the Alpha architecture that delivers peak performance of 1,200 million instructions per second (mips)/600 million floating-point operations per second (MFLOPS). The chip is designed in a 0.5-micrometer (um) complementary metal-oxide semiconductor (CMOS) technology using four levels of metal. The die measures 16.5 millimeters (mm) by 18.1 mm and contains 9.3 million transistors. It operates at 3.3 volts (V) and supports 3.3-V and 5.0-V interfaces. The chip dissipates 50 watts (W) at 300 MHz (internal clock frequency). Switching noise on the power supplies is controlled by an on-chip distributed coupling capacitance between power and ground of 160 nanofarads (nF). The chip contains an 8-kilobyte (KB), first-level (L1) instruction cache; an 8-KB L1 data cache; and a 96-KB second-level (L2) unified data and instruction cache.

This paper focuses on the circuit implementation of the Alpha 21164 CPU. Space does not permit a description of the complete design process utilized throughout the project. Instead, some of the significant circuit design challenges encountered during the project are discussed. The paper begins with an introductory overview of the chip microarchitecture. It continues with a description of the floorplan and the physical layout of the chip. The next section discusses the clock distribution and latch design. This is followed by an overview of the circuit design strategy and some specific circuit design examples. The paper concludes with information about design (physical and electrical) verification and CAD tools.

MICROARCHITECTURE OVERVIEW

The Alpha 21164 chip is a completely new implementation of the Alpha architecture. Figure 1 shows a block diagram of the Alpha 21164 chip. The microprocessor consists of five functional units: the instruction fetch, decode, and branch unit (I-box); the integer execution unit (E-box); the memory management unit (M-box); the cache control and bus interface unit (C-box); and the floating point unit (F-box). The chip contains three on-chip caches: the instruction cache (I-cache); the data cache (D-cache); and the second-level (unified data and instruction) cache (S-cache). The microprocessor uses a seven-stage pipeline for integer and memory operations and a nine-stage pipeline for floating-point operations. Pipeline stages are referred to as S0 to S8 for the remainder of this paper.

[Figure 1 (Block Diagram of the Alpha 21164 Functional Units) is not available in ASCII format.]

The I-box fetches instructions from the 8-KB virtual, direct-mapped, physical L1 I-cache. It decodes and issues instructions to the E-box, M-box, and F-box. It maintains state for all pipeline stages to track outstanding register writes. The I-box can decode and issue up to four instructions per cycle. The E-box contains two 64-bit pipelines. The first pipeline contains an adder and a Boolean logic unit. The second pipeline contains an adder, a Boolean logic unit, a shifter, and a multiplier. Most integer instructions execute in one cycle. The F-box contains a floating-point multiply pipeline and a floating-point add pipeline. The M-box executes all load and store instructions. The 8-KB, direct-mapped, write-through, L1 D-cache has two read ports, which allow two load instructions to execute in parallel. The C-box processes memory accesses sent by the M-box and manages all cache coherence protocol functions. It also controls the L2 unified cache and an optional external (off-chip) backup cache. A detailed discussion of the chip microarchitecture is presented in another paper in this issue.[1]

FLOORPLAN OVERVIEW

With 9.3 million transistors and a die size very close to the manufacturing limit, floorplanning of the Alpha 21164 chip was a critical activity during development. Power and ground distribution, as well as clock routing and signal busing, were also major factors in developing a viable floorplan. The floorplan was developed early in the project and was regularly monitored and kept up-to-date. Chip area estimates were made using several different methods. The area estimates of the caches were determined by creating trial layout structures. Major sections of the integer and floating-point execution units were estimated using data from previous designs and trial layout. The control logic areas were estimated using data from logic and layout synthesis CAD tools.

Figure 2 is a photomicrograph of the chip with overlays showing the major functional units and pin locations. The major functional units were positioned to minimize critical signal propagation delays and the length of global buses. The I-box, E-box, and M-box were aligned to allow the routing of the data between these sections in a common data path. Since the C-box interfaces with most of the other functional units, it was situated along the bottom of the chip to accommodate different signal routing needs. The address pins for the memory interface were placed at the bottom of the chip to keep them in close proximity to the C-box and M-box, thereby reducing routing and latency. Due to the significant data routing between the S-cache, C-box, and pins, the placement of these sections was critical. Both the 128-bit-wide S-cache data array and the 128-bit-wide C-box data path were split into two halves, each containing the upper and lower 64 bits. This minimized the routing between the C-box, the S-cache, and the data pins along the two sides of the chip.

[Figure 2 (Floorplan of the Alpha 21164 Chip) is not available in ASCII format.]

The D-cache and the I-cache were arranged to allow 128-bit-wide data access to the S-cache. The I-cache and D-cache were placed above and below the S-cache, respectively, and centered within the chip as much as possible. The S-cache was implemented with metal 4 read and write buses spanning the entire height. This provided the access needed to both the top and bottom of the S-cache for routing to the I-cache and D-cache.

The D-cache supports two loads per cycle, requiring a dual-ported read design. The D-cache was implemented as two single-ported caches containing identical data instead of one dual-ported cache. The major consideration that led to this decision was the ability to share the single-ported design with the I-cache. Sharing the design also reduced the overall analysis and verification required.

Interconnect routing was another important part of the floorplanning process. Four metal layers were available for routing. The lower metal layers, metal 1 and metal 2, were used for local transistor hookup and signal routing. The upper metal layers, metal 3 and metal 4, were used primarily for clock, power, and ground distribution. When necessary, the upper metal layers were also used to route critical signals and long buses. The metal orientations were chosen to accommodate both the cache structures and the data paths of the functional units. With reference to Figure 2, metal 2 and metal 4 lines were arranged to run vertically and metal 1 and metal 3 were arranged to run horizontally. Most of the global routing was done by hand. Local cell routing was done by hand with some assistance from auto-routing CAD tools.

The upper metal layers are organized as a fine-pitched regular grid structure placed over the entire chip. The typical drawn line width used in this grid is 12 um. Power and ground lines are alternated with a single clock line interspersed every few pairs. A limited number of critical signals and buses are also routed in metal 3 and metal 4. In the pad ring, metal 4 is used to route power and ground into the chip from the pads, and input and output signal wires are routed circumferentially in metal 3.

The fine pitch of the power grid allowed its placement and hookup to be deferred until most of the cell layout was complete. Therefore, individual cell layout was done independently of the main grid. The local power, ground, and clock lines in cells were connected to vertically routed metal 2 lines. These metal 2 lines were designed to be long enough to span two or more of the metal 3 grid lines, which allowed the eventual connection to the grid.

Cell hookup to the grid was automatically generated using an in-house CAD tool that accepted nodal layout extracts of the power, ground, and clock nodes. It connected the grids by placing a maximum number of contacts between metal 4 and metal 3, and metal 3 and metal 2. All contacts placed by the tool met the process design rules.

CLOCK DISTRIBUTION SCHEME AND LATCH DESIGN

The Alpha 21164 chip enhanced the clock distribution mechanism and verification techniques that were developed during the design of the first-generation Alpha microprocessor chip, the DECchip 21064.[2] A global, single-wire clock (CLK) is distributed over the chip. The design quality of the clock signal was critical to meeting the fast cycle time goal of the Alpha 21164 chip. (Note that the high state of the clock signal is identified as the A-phase, and the low state of the clock signal is considered the B-phase.)

Figure 3 is a circuit schematic of the clock generation circuitry. The chip receives a 2X differential emitter-coupled logic (ECL) oscillator clock signal (up to 600 MHz) that is divided by two and then routed to the center of the chip. It is then buffered up through 6 inverter levels and fanned-out to 24 inverters through a balanced tree to drive the PRE_CLK signal in metal 3 to the right and left banks of the final stage clock drivers. These final buffers are located between the second-level caches and the outside edge of the execution core. The final clock driver has 4 more levels of inverter buffering before it drives the single-wire CLK signal that is formed using a metal 3 and metal 4 grid. Figure 4 is a schematic showing the placement of the CLK grid.

[Figure 3 (Schematic of Clock Distribution System) is not available in ASCII format.]

[Figure 4 (Metal 3 and Metal 4 Clock Grid) is not available in ASCII format.]

There are 44 final clock drivers per side. The placement of these drivers was based on local clock loading. There are also 12 conditional clock drivers per side, with each activating a single S-cache subarray. Each clock buffer is surrounded by decoupling capacitance to reduce switching noise. The last clock driver inverter has a total gate width of 58 centimeters (cm). It drives a load of 3.75 nF of gate and interconnect capacitance. The clock distribution system consumes about 20 W, approximately 40 percent of the total chip power.

The two banks of final clock drivers, coupled with the size of the clock network, complicated the clock skew analysis. The precise method of calculating clock skew requires a SPICE model of the entire network from input oscillator signal through the drivers and the entire metal grid.[3] This approach was not possible due to the size of the network. Instead, the three main components of clock skew were quantified: (1) PRE_CLK driver and PRE_CLK RC delay variations, (2) final CLK driver transistor delay variations, and (3) RC delay through the metal grid. A SPICE model of the PRE_CLK drivers and network was simulated to identify the differences in arrival time of the PRE_CLK signal at the input to the final drivers.[3] The simulations showed that any delay variations developed in this portion were equalized by strapping the inputs and outputs of the final driver inverters. The net result was that the PRE_CLK network contributes only 10 picoseconds (ps) to the total skew.

The next source of variability affecting clock skew was the difference in transistor characteristics among the final stage drivers. There are actually two causes of this variation: layout-related effects and systemic intra-die process variations. Although the layout-related effects can be controlled in the design, the process variations cannot. To limit channel length variations that result from layout differences, a joint effort was undertaken with the process development group to define a modular block for the driver layout. This block was then repeated as many times as necessary to achieve the required total driver size while keeping the polysilicon spacing, density, and orientation constant.

Since it was not possible to model the metal 3 and metal 4 grid with the drivers in SPICE, extensive RC delay simulation of the clock grid alone was done based on resistance and capacitance values extracted from layout.[3] The drive impedance of each clock driver was modeled as an equivalent resistance in the network. The skew between the first and last latch receiving the CLK signal is 90 ps. The instruction and execution units all see the CLK signal within 65 ps, which is well within the design goal. Figure 5 shows the clock RC delay as a function of the X and Y locations on the chip.

[Figure 5 (Clock RC Delay) is not available in ASCII format.]

The development of standard latches was an important aspect of the Alpha 21164 implementation process. The primary goal of the latch design was to produce a fast circuit that would use the device area efficiently and that could be used in a wide variety of instances. To minimize the chip verification effort, a standard latch library was developed early in the design process. This library set the standards for latch usage and allowed designers to utilize latches that had already been verified over a range of operating conditions and process corners.

The Alpha 21164 chip uses level-sensitive, transmission gate latches as shown in Figure 6. Two basic types of latches were developed: A-latches (Figures 6a and 6c) and B-latches (Figures 6b and 6d). The A-latches are open when CLK is high, and the B-latches are open when CLK is low. The latch input inverter can be replaced by a logic gate (shown in Figures 6c and 6d), thus reducing gate delays in other logic. This style of latch is very fast and area-efficient, yet it does have an inherent race-through problem. It was estimated that the use of this latch style yields a 10 percent improvement in speed over the 21064 microprocessor.

[Figure 6 (Alpha 21164 Standard Latch Examples) is not available in ASCII format.]

The additional skew in the clock, resulting from the local clock buffer delay, increases the possibility that data could race through a pair of latches during the transition of the clock. Although the overall skew of the internal clock is low, this was not considered sufficient to avoid race conditions. Two significant steps were taken to guarantee that no race could exist between latches. First, the buffered clock inside the latch was sized to minimize the additional skew resulting from its delay. Second, rules and verification tools were developed to make sure that the design includes at least one additional gate delay between all latches, thus guaranteeing a race-free design. Designers had the option of designating these gates as logic functions or simple inverters. The delay did not affect critical speed paths, since critical paths tended to have more than one delay between latches.

CIRCUIT DESIGN STRATEGY

Due to the complexity of the Alpha 21164 chip and the large size of the design team, a comprehensive design methodology was developed. A design guide was created to provide a consistent set of rules and methods for the development of circuit schematics and layout. This document helped ensure that all designers worked under the same design assumptions. In addition, it relieved time-consuming analysis of each circuit by providing guidelines and "rules of thumb" that guaranteed correct operation and minimized the possibility of reliability problems.

Guidelines for common circuit structures such as complementary, cascode, dynamic, and static circuits were created by characterizing their behavior over all process corners. Adequate noise margins were ensured by specifying operating envelopes for such design parameters as device size, stack height, and beta ratio. Reliability guidelines were specified for electromigration, hot carrier effects, and substrate charge injection. Most circuits were designed within the rules specified in the guide; however, a few circuit designs violated the rules. These designs were allowed only when performance and area advantages would be gained. These exceptions were carefully verified for functionality and reliability.

An extensive suite of in-house CAD tools was used to aid and structure the design process. In all cases, the tools supplemented the design process and automated repetitive work. Engineering judgment and iterative use of the software were required to create the final production schematics. Tools that aided schematic generation included a schematic editor, a logic synthesis tool, and a device-sizing tool. Post-schematic tools included a latching methodology checker, a circuit verifier that highlighted design methodology violations, and a timing verifier that analyzed potential critical speed paths. The use of the design tools varied across the chip, based on the degree of customized logic required. For example, the I-box did not rely heavily on the synthesis tools because of the need for optimized circuit structures. However, the C-box used the synthesis tools extensively to produce baseline schematics, which were then modified by hand as necessary.

CIRCUIT DESIGN EXAMPLES

The designers of the Alpha 21164 chip were faced with a number of implementation challenges. The most significant challenge was to design a chip that could run at 300 MHz, 50 percent faster than the previous Alpha implementation.[2] Device scaling, process development, and architectural improvements delivered part, but not all, of the required speedup. The additional improvement had to be obtained using circuit design techniques. Other challenges included a much more complicated microarchitecture and the reduction in latency of a number of instructions from the previous implementation. Finally, the large physical size of the chip also led to challenges in circuit design and power management.

The following sections describe several circuit design challenges encountered during the implementation of the Alpha 21164 chip.

I-box Design -- Issue Stage Dynamic Dirty/Bypass Logic

The issue stage of the I-box coordinates the release of instructions into the E-box, F-box, and M-box pipelines. The deep pipelines and sophisticated memory management unit along with the high clock frequency presented significant challenges to the implementation team. The Alpha 21164 microarchitecture allows up to 37 instructions to be in progress at the same time (7 integer operates, 9 floating operates, and 21 loads that missed). Superscalar issue of 4 instructions requires that 8 operands and 4 new destinations must be checked against these 37 outstanding instructions in every cycle. In addition, 44 bypass paths are built into the E-box and F-box pipelines in the Alpha 21164 chip. Each of the 8 operands must be checked against several of these bypass paths to ensure that the most up-to-date data is forwarded to the issuing instruction.

The register comparisons were implemented using domino logic. As each instruction is issued, its destination register address is decoded into a 31-bit mask that is entered into a shift register that mimics the appropriate execution pipeline. Checks are performed for stalls and bypasses by selecting the appropriate masks from each level of the shift register and comparing them to the register addresses of the new instructions. Integer and floating-point instructions are handled in separate 31-bit-wide data paths.

Decoding the register addresses allows a logical OR of several destinations to create "dirty" bit masks, greatly reducing the required number of comparators. This reduction in comparators more than compensates for the additional logic involved in carrying the decoded register addresses for all pipe stages (31 bits versus 6 bits for encoded register numbers). With this scheme, all stall calculations are performed using only 38 comparators. Bypass detection is performed in a manner similar to the stall generation using an additional 44 comparators, one for each E-box and F-box bypass path.

The implementation of the comparators requires three domino stages (see Figure 7). The first stage is a two-input dynamic multiplexer that selects the operand/destination decode field for the new instruction or the field of the previous cycle's instruction if a stall was detected. The dirty bit mask is created in a similar dynamic OR structure. The second domino stage is a bit-wise AND function of the operand/destination decode mask and the dirty bit mask followed by a zero detector (logical OR of the 31 bits). A transmission gate forms a second AND function in this stage that qualifies the detected register conflict with an instruction valid signal. The third domino stage is used to further qualify the detected conflict with instruction type decode information and to start a logical OR of the 38 conflict outputs into a single stall wire. In the case of bypasses, the third domino stage is used to priority-encode the bypasses so that only the most up-to-date data is bypassed.

[Figure 7 (Domino Logic for Issue Scoreboard) is not available in ASCII format.]

Special attention was given to several circuit design issues when the domino logic was implemented. Careful preplanning of the routing provided large lateral spacing on the dynamic lines to reduce coupling. Noise margins were protected by ensuring that all dynamic inputs were driven from local inverters with a common ground reference. Charge-share problems in the large second domino stage (31-bit-wide AND-OR function) were minimized due to the fact that only a single bit will be set in the new instruction's operand decode bit mask, which is used as the upper input in the 31 AND stacks. Therefore, only a single internal node may charge-share with the large output capacitance.

Another critical concern in such a large dynamic structure was power consumption. The logic was implemented in such a way as to minimize the number of nodes that discharge each phase. To minimize short-circuit currents, the second and third domino stages are precharged by means of matched delay signals. These self-timed precharged lines also help to minimize clock loading since CLK is used to precharge only the first stage.

E-box Design -- Bypass Logic

The E-box presented a number of interesting circuit challenges. The Alpha 21164 implementation contains two integer pipelines, as compared to one in the 21064. This significantly increased the circuit design complexity associated with providing result bypassing from all functional units.

The E-box bypass logic is responsible for supplying input operand data to the functional units in both integer pipelines. Input operand data can be supplied from the register file or bypassed from the output of any pipeline stage in the E-box (Figure 8). Functional operations are performed in pipeline stage 4 (S4), and register file writes occur in stage 6 (S6). Without bypass logic, instructions that require data from the pipeline would have to be stalled until the data reaches S6 and is written into the register file. These stalls would impact the integer performance severely. Therefore, the ability to bypass operand data from pipeline stages S4 through S6 was critical to obtaining high integer performance.

[Figure 8 (E-box Pipeline) is not available in ASCII format.]

Four 64-bit dual-rail operand buses are used to bypass data. Two buses in each pipeline are used to supply A and B operand data to the functional units. The buses are controlled by the BYPASS_ENABLE_L signals generated in the I-box and are driven during the B-phase (see Figure 9). A typical operand bus driver is shown as well as the shifter operand bus driver. The shifter driver is unique because it has byte zap (set byte to zero) logic capability.

[Figure 9 (E-box Bypass Bus) is not available in ASCII format.]

Data is read from the operand buses during the early portion of the A-phase by operand bus receivers located at the input of each functional unit. The receiver is a dynamic gate structure that can be configured to receive one or more inputs and generate a logical function output. The adder uses the logical function capability to generate propagate and kill signals.

The operand bus is precharged by a delayed A-phase clock. This delay allows the dynamic bus receiver gate to act as a latch and eliminates the need for a true B-latch (see Figure 9). During the beginning of the A-phase, operand data propagates through the receiver and is captured by the receiver gate output latch node before the delayed A-phase clock precharges the operand bus. Once the operand bus is precharged, the latch node is decoupled from the operand bus.

E-box Design -- 64-bit Shifter

The E-box shifter executes all 64-bit shift, extract, insert, and zap (set to zero) instructions on both little and big endian data types using a 128-bit right-only shifter. All shift instructions take one cycle to execute, an improvement of one cycle relative to the 21064 design.

The data path portion of the shifter logic uses dynamic and cascode circuitry to read the operand buses, to present the data to either the low or high 64 bits of data, and to sign-extend the high 64 bits, when necessary, in the A-phase. In the B-phase, the input data is shifted, a byte zap is performed when necessary, and the result is driven onto the result bus. The result can be bypassed onto an operand bus. Right shifts are performed by loading the A-operand data into the low 64 bits and shifting based on the value of the B-operand; left shifts are performed by loading the A-operand data into the high 64 bits and shifting based on the two's complement of the value of the B-operand. The shifter array is implemented as a differential dynamic gate. The layout uses metal 1 for the input data, metal 2 for the output value, and metal 3 for the shift amount.

The chief improvement in this design over the 21064 design is the single-phase generation of the 65 shift enable signals and byte zap mask. The shift enable generation is accomplished by combining the shift requirements of the extract and insert instructions with the B-operand decode logic for normal shifts.

An 8-bit shifter is used to implement the byte zap mask to achieve the single-phase goal. The 8-bit zap mask shifter is built using differential dynamic logic. Its control resembles that of the 64-bit shifter, employing cascode data input circuitry and dynamic decode logic. The shift amount is determined from the B-operand or bits in the instruction based on the opcode.

Cache Design -- Power Savings

Special design considerations were given to the three caches on the Alpha 21164 chip because they comprise, by far, the largest number of devices and have the greatest impact on yield. Since the caches are accessed frequently, the power consumption of the caches was also a cause for concern.

The 8-KB I-cache includes two pairs of fuse-programmable redundant rows to offset any yield loss. The D-cache leverages the I-cache design by combining two of these caches to form a single, dual-read-ported, 8-KB data cache. The D-cache employs the same row redundancy scheme as the I-cache. The Alpha 21164 chip also contains the S-cache, which is a large, second-level cache for both data and instructions. The S-cache data array is organized into 24 banks of 4 KB each. Twelve banks are placed on the left and right sides of the chip. Figure 10 shows the arrangement of the banks on the right side. Each bank of both the tag and data arrays implements row redundancy. The S-cache data array also implements column redundancy.

[Figure 10 (Schematic of Right Half of L2 Cache Data Arrays) is not available in ASCII format.]

Pipeline processing of the S-cache allows the inclusion of power-saving features. The S-cache operates in a four-stage pipeline: two stages for tag lookup and modification, and two for data access and transfer. Address decoding during the tag lookup results in the clocking of only 2 of the 8 banks in each of the 3 sets (6 of 24 in the whole cache). The bit lines and sense amplifiers in the disabled 18 banks are frozen in the precharge mode, consuming minimal power.

Hit signals from the tag-lookup logic control the word lines and sense amplifiers of the six enabled banks. Therefore, of the six banks enabled, only the two banks for the set that hit are activated and discharged. This design results in an estimated power savings of 10 W.

System Clock Design -- Synchronization

The Alpha 21164 chip is designed to accommodate multiprocessor systems using a synchronous bus. This requires the synchronization of the Alpha 21164 chip's generated reference clock (SYS_CLK) to the systems-generated reference clock (REF_CLK). To achieve the maximum system performance, this must be done with as little error as possible.

In other designs, this synchronization is achieved using an on-chip phase-locked loop (PLL).[4] However, the on-chip noise environment of the Alpha 21164 could cause excessive PLL jitter. Jitter can reduce the width of a clock phase and create a pulse too narrow to clock on-chip logic. This uncertainty would dictate slowing the clock frequency, thus reducing system performance.

The design challenge was to find a low-risk digital solution that would meet the high-frequency performance requirements of the Alpha 21164. To meet this challenge, a state machine PLL (SMPLL) was designed. This all-digital approach has much better noise immunity than a traditional PLL, but it does introduce a quantizing error, or skew, into the system clock timing. This skew can complicate system timing but has minimal impact on CPU performance, since it allows the Alpha 21164 chip to run at the highest possible clock frequency.

Figure 11 shows a functional block diagram of the SMPLL. The Alpha 21164 generates a system bus clock (SYS_CLK) by dividing the internal CLK by a preprogrammed amount. This SYS_CLK is then aligned to the system-generated reference clock (REF_CLK). To do this, the frequency of REF_CLK must be slightly lower than that of SYS_CLK. A phase detector compares the arrival of the rising edge of REF_CLK with the rising edge of SYS_CLK. If the edges are coincident, the SMPLL stretches SYS_CLK by the period of the chip oscillator. Thus, the rising edge of REF_CLK always leads the rising edge of SYS_CLK. However, because SYS_CLK is slightly faster than REF_CLK, the rising edge of SYS_CLK will eventually catch up to REF_CLK. When this happens, the phase detector once again stretches SYS_CLK, and the process of catching up starts anew.

[Figure 11 (SMPLL Block Diagram) is not available in ASCII format.]

The SMPLL design takes advantage of the on-chip clock divider circuitry by suppressing the divide for a single count whenever a phase alignment is required. This scheme adjusts the phase alignment in increments of 1.67 nanoseconds (ns) (assuming a 600-MHz input clock) and allows the rising edge of the REF_CLK, measured at the input pin of the Alpha 21164, to coincide with the SYS_CLK to within 1.67 ns.

PHYSICAL AND ELECTRICAL VERIFICATION

The ability to verify the layout of a 9.3-million-transistor VLSI chip, both physically and electrically, without hampering its performance or impacting its development schedule, was a primary concern from the outset of the project. Many new techniques were developed to accomplish this task. Some of the more significant advances are discussed in the following section.

Physical Layout Verification

The size and complexity of the Alpha 21164 dictated the use of physical assembly methods that did not require the CAD tool suite to verify the complete chip layout database in one pass.

Full-custom designs like the Alpha 21164 chip are composed of large blocks of random logic that are not easily divisible into highly repetitive instantiations of common cells. Because of the relatively few instances of repetitive structures, there was no need to design using a deep cell hierarchy. Similar to previous, large, full-custom designs, the Alpha 21164 floorplan divided the chip along major box boundaries.[5] This partitioning reduced the device count per partition, allowing each to be verified independently.

The Alpha 21164 cache partitions, containing 7.2 million of the 9.3 million total devices, are, in themselves, very large and difficult to verify. Accordingly, all three caches were designed and assembled hierarchically. Specifically, each cache bank contains several references to the same precharge, decoder, control, and random access memory (RAM) array logic and layout, which are then instantiated to form the overall cache. Subdividing the cache partitions into major hierarchical blocks reduced the device count per block. In addition, since each bank was identical, only the devices within one bank needed to be verified.

Although the hierarchical method is typically used in semicustom designs, it was new for Digital's full-custom microprocessors. Prior to the development of the Alpha 21164, the caches were designed and verified without an established hierarchy, as was the rest of the chip. Digital's CAD tools handled large databases without hierarchy; the layout verification methods were trusted; and the percentage of duplicated circuitry was small. Consequently, there had been no prior compelling need to design with deep hierarchies.

Toward the end of the chip development, using a considerable amount of computer resources, all three Alpha 21164 cache layouts were also verified without hierarchy to prove the new hierarchical method and CAD tools. The large size of the Alpha 21164 made it the prime candidate for verifying new hierarchical verification tools (which were run concurrently with the traditional ones). Table 1 compares the processing time of the S-cache for both the nonhierarchical and the hierarchical verification methods. The hierarchical approach resulted in a significant improvement in CPU time.

Table 1 Alpha 21164 S-cache Verification Compute Time

Operation	Nonhierarchical Processing	Hierarchical Processing
Netlist extraction	11 hours	6 hours
Netlist comparison	6 hours	30 minutes
Geometric verification	18 hours	10 minutes

Capacitive Coupling and Carrier Injection Verification

Since capacitive coupling between adjacent signals can have a disastrous effect on the logical functionality and long-term reliability of a design, it was a major concern throughout the project. When adjacent nodes switch, coupling between them can result in their logic state being degraded or lost by adding or removing charge to or from the coupled node. For static cases, coupling results in a loss in performance, since the node recovers state if the chip cycle time is slowed. For dynamic nodes, however, state may be lost, leading to a logic failure that occurs regardless of cycle time.

Interconnect coupling capacitance can also lead to voltage excursions above the power supply voltage (VDD) and below ground (VSS) on signals in the chip. For the case of an excursion below VSS, the n-type source/drains connected to the signal become forward biased, injecting minority carriers (electrons) into the substrate. If these minority carriers are collected by N-diffusions connected to dynamic nodes, the charge stored on the dynamic node can be corrupted, as shown in Figure 12. Similarly, excursions above VDD forward bias p-type source/drains, which can also lead to data corruption.

[Figure 12 (Dynamic Node Corruption Caused by Minority Carrier Injection) is not available in ASCII format.]

An extensive set of CAD tools was used to identify potential coupling and charge injection problems. In the case of injection checks, a circuit wirelist of the chip was extracted from the layout that included X-Y location coordinates for all transistors. An electrical analysis, using capacitances extracted from layout, was then run to identify all nodes that made voltage excursions outside the power supply voltages and that were potential minority carrier injectors. Once these nodes were identified, the CAD tool, which referenced the coordinates from the extracted wirelist, checked all circuitry in the vicinity of the injectors to ensure that there were no dynamic nodes present that could be corrupted. When a potential corruption problem was found, a layout fix was implemented to eliminate the coupling causing the injection. If the coupling could not be reduced or eliminated, a diffusion collector tied to a power rail was placed between the injector and the dynamic node (Figure 13).

[Figure 13 (Dynamic Node Protected from Minority Carrier Injection) is not available in ASCII format.]

Antenna-induced Device Damage Analysis

During the metal etch process, when interconnect is being formed from a blanket layer of metal, stray charge from the etch plasma can be captured by the visible metal. The charge is collected on any polysilicon gate capacitors attached to the node. If enough charge is collected, the gate voltage may rise high enough for tunneling into the gate oxide to occur. This new concern, called antenna-induced device damage, can cause breakdown of the gate oxide, transistor threshold voltage shifts, and long-term reliability problems.

Antenna-induced device damage can be prevented if an alternate path is provided for the collected charge. A diode connection on the antenna node, such as a diffusion connection in either the well or the substrate, acts as such a path. Although all nodes in the Alpha 21164 chip have a diode connection, this connection may not be present at the first or second metalization steps, thereby allowing damage to occur. The magnitude of the damage is dependent on the antenna ratio, defined as the ratio between the area of the visible metal layer being processed and the area of the gates attached to that node through lower-level connecting layers.

To analyze the chip, a special computer-based layout design rule check was developed. This check extracted partial node layout as it would appear during each metalization-patterning etch step and filtered all nodes that did not have a diode shunt connection. For these nodes, antenna ratios were computed and compared to their corresponding ratio limit. To reduce the antenna ratio of a failing node, the antenna metal was broken into sections and metal jumpers, which were placed in the next-higher adjacent metal layer, to connect the sections into a single node. This reduced the charge-collecting area for the section of interconnect that had the polysilicon gate attached and, as a result, reduced the antenna ratio. If this approach was not feasible or did not reduce the antenna ratio adequately, a diffusion diode was attached to the offending antenna to shunt the charge away.

Electromigration Reliability Analysis

The methods and algorithms used to perform the electromigration (EM) analysis on the Alpha 21164 chip have greatly improved since previously reported.[5] The chief enhancements are the analysis of unidirectional and bidirectional current flow, the addition of thermal heating models, and the introduction of statistical electromigration budgeting.

From a design perspective, one of the main improvements in EM analysis was the introduction of unidirectional and bidirectional current flow limits. Unidirectional current is the flow of current in one direction, for example in wires connecting devices to power or ground. The segment of wire connecting a complementary logic gate to its load is considered bidirectional since the current flows toward the load to charge its capacitance and flows back to the driver as the capacitance is discharged. The bidirectional behavior of current has been shown to improve EM reliability by at least a factor of two. This is a tremendous benefit as nearly all on-chip signal wiring is bidirectional. (Power supply metal is not and must be treated accordingly.)

The most stringent EM requirement is meeting the traditional average current density limit of 2.0 milliampere/um**2. Statistical electromigration budgeting (SEB) was used for the first time during design verification to assess the impact of allowing small portions of the Alpha 21164 design to exceed the fixed EM average current limits.[6] Statistical parameters characterizing EM risk for the 0.5-um CMOS interconnect process were combined with the average node currents and layout geometry to compute the magnitude of the EM risk of all design rule violations taken together. Only those violations that added significant risk were required to be fixed. This reduced design verification time and retained performance advantages while ensuring that the Alpha 21164 design met its chip-level reliability goals.

CONCLUSION

The implementation details of the Alpha 21164 microprocessor have been described. The custom VLSI chip contains 9.3 million transistors, including a 96-KB second-level cache, in an area of 299 mm**2. The chip implements the Alpha instruction set architecture and can issue up to four instructions at a time. It reaches a peak execution rate of 1.2 billion instructions per second (bips) and 600 MFLOPS. The Alpha 21164 is the fastest and highest-performance microprocessor designed to date in the industry.[7]

The chip achieved its performance goal of 300-MHz operation in a 0.5-um CMOS technology by employing a fine-pitch, low-resistance power grid; a low-skew clock distribution network; fast latches; and high-speed circuit techniques. Extensive verification of the functionality, electrical circuits, and physical layout was performed to ensure the functionality and reliability of the design. The chip operates from a 3.3-V supply and dissipates 50 W. It is easily air-cooled using conventional technology. First-pass silicon was functional and booted three operating systems running on a number of different system platforms.

ACKNOWLEDGMENTS

We would like to acknowledge the contributions of many people who helped make this chip possible. These include William Herrick and Paul Rubinfeld for management and project support; Alan Cave and Radenko Cvijetic for invaluable CAD assistance; and Larry Bair, Narain Arora, Len Gruber, and Bjorn Zetterlund for device and technology modeling. Designers include Randy Allmon, Roy Badeau, Pete Bannon, Todd Benninghoff, Randel Blake-Campos, Derek Brasili, Kevin Broch, Todd Broch, Mike Charnoky, Beth Cooper, Dan Dever, Rob Dupcak, Tim Fischer, Frank Fox, Rich Fromm, Bruce Gieseke, Mary Gowan, Charles Hightower, Jim Keller, John Kowaleski, Tim Mast, Anthony Murphy, John Mylius, Andy Olesin, Tung Pham, Nate Raughley, Don Priore, Vidya Rajagopalan, Steve Strickland, Chandra Somanathan, Jon White, Gil Wolrich, and the authors of this paper. Custom layout was done by Picco Aires, Sandy Carroll, Jeff Ceparski, Danielle DeMarse, Gina Franceschi-Bean, Mark Gaetz, Natasha Geagan, Jerry Heath, Susan Lowell, Tom McDermott, Karen McFadden, Rich Matthew, Stephanie Miller, Sue Moore, Brian Mulhollen, Dave Olson, Marie Riley, Avraham Shenvald, Chad Stark, Marc Tareila, Lang Tran, and Greg Williams.

REFERENCES AND NOTE

1. J. Edmondson et al., "Internal Organization of the Alpha 21164, a 300-MHz 64-bit Quad-issue CMOS RISC Microprocessor," Digital Technical Journal, vol. 7, no. 1 (this issue): 119-135.

2. D. Dobberpuhl et al., "A 200-MHz 64-bit Dual-issue CMOS Microprocessor," Digital Technical Journal, vol. 4, no. 4 (Special Issue 1992): 35-51.

3. SPICE is a general-purpose circuit simulator program developed by Lawrence Nagel and Ellis Cohen of the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley.

4. K. Kurita et al., "PLL-Based BiCMOS On-Chip Clock Generator for Very High-Speed Microprocessor," IEEE Journal of Solid State Circuits, vol. 26, no. 4 (April 1991): 585-589.

5. D. Donchin et al., "The NVAX CPU Chip: Design Challenges, Methods, and CAD Tools," Digital Technical Journal, vol. 4, no. 3 (Summer 1992): 24-37.

6. J. Kitchin, "Statistical Electromigration Risk Budgeting for Reliable Design and Verification in a 300MHz Microprocessor," Digest of Technical Papers, VLSI Circuits Symposium, 1995.

7. "Digital Leads the Pack with 21164," Microprocessor

Report, vol. 8, no. 12 (September 1994): 1, 6-10.

BIOGRAPHIES

William J. Bowhill

William Bowhill is a consultant engineer in Digital Semiconductor's High Performance CPU Group. He co-led the implementation of the Alpha 21164 CPU and represented the design organization during the development of the 0.5-um CMOS process in which the chip is fabricated. His previous responsibilities include technical contributions to both the VAX 6000 Model 400 and Model 600 chip sets. Before joining Digital in 1985, he worked for Standard Telecommunications Laboratories, Harlow, England, where he designed VLSI chips for telecommunication applications. Bill received a B.Eng. (honors) in electronic engineering from the University of Liverpool in 1981.

Shane L. Bell

Shane Bell joined Digital after receiving a B.S. in computer systems engineering from the University of Massachusetts at Amherst in 1993. As a hardware engineer in Digital Semiconductor, he worked on the integer execution unit of the Alpha 21164 CPU. He is currently involved in the design of another high-performance microprocessor. Shane is a member of Eta Kappa Nu, Tau Beta Pi, and IEEE.

Bradley J. Benschneider

Brad Benschneider is a principal hardware engineer in Digital Semiconductor. He was responsible for designing various sections of the memory management unit on the 21164, as well as defining the latching methodology for the chip. He is currently leading the implementation effort of the memory management unit for the next-generation Alpha CPU. Since joining Digital in 1987, he has contributed to several custom chip designs in the VAX 6000 family and the early Alpha implementations. He received a B.S.E.E. from the University of Cincinnati, has one patent, and has co-authored four papers.

Andrew J. Black

Andy Black is a senior hardware engineer in Digital's Palo Alto Design Center, where he is designing the bus interface unit for the StrongARM PDA microprocessor chip. During his work on the Alpha 21164 CPU, he was a member of the design team for the memory management unit and contributed to the chip's clock design. Andy joined Digital in 1992 after working for International Solar Electric Technology. He received a B.S.E.E. from Pennsylvania State University and an M.S.E.E. from the University of Southern California. Andy is a member of IEEE, Tau

Beta Pi, and Eta Kappa Nu.

Sharon M. Britton

Sharon Britton is a principal hardware engineer in Digital Semiconductor. She received a B.S.E.E. from Boston University in 1983 and an M.S.E.E. from MIT in 1990. She joined Digital in 1983 to work on the design and development of optical disk drive controllers. Since joining Digital Semiconductor in 1990, Sharon has contributed to the design of the floating-point unit on the 21064 CPU chip and led the implementation of the load/store unit for the Alpha 21164 CPU. She is currently a member of the design team working on the instruction issue unit for the next-generation Alpha chip.

Ruben W. Castelino

Before receiving a B.S.E.E. from the University of Cincinnati in 1988, Ruben Castelino was a co-op student at Digital working on a chip set for the VAX 6000 Model 200. Currently a senior hardware engineer in Digital Semiconductor, he was a codesigner of the cache control and bus interface unit for the Alpha 21164 CPU. Prior to that, he worked on the instruction fetch, decode, and branch unit for the NVAX chip and performed implementation work for the NVAX virtual instruction cache. Ruben is currently a codesigner of the cache control and bus interface unit for a new Alpha microprocessor.

Dale R. Donchin

Dale Donchin is an engineering manager and technical contributor in Digital Semiconductor. He designed several circuits related to the clock and cache and contributed to and led CAD tool use for the Alpha 21164 CPU. He is presently performing these duties for the development of the next-generation Alpha microprocessor. Dale joined Digital in 1978 and was previously a development manager in the RSX Operating System Group. Dale holds a B.S.E.E. (1976, honors) and an M.S.E.E. (1978) from Rutgers University College of Engineering and is a member of IEEE and ACM.

John H. Edmondson

John Edmondson is a consultant engineer in Digital Semiconductor. He was the architecture leader of the design team for the Alpha 21164 microprocessor. Previous to that work, he was a member of the design team for the VAX 6000 Model 600 microprocessor. Prior to joining Digital in 1987, John worked at Canaan Computer Corporation and Massachusetts General Hospital. John received a B.S.E.E. from the Massachusetts Institute of Technology in 1979.

Harry R. Fair, III

Harry Fair is a senior hardware engineer in Digital Semiconductor's Advanced Development Group and is currently working on the design of the instruction issue unit for a high-performance microprocessor. Harry came to Digital in 1985 as a co-op student and worked on the VAX 6000 Model 400 chip set. He joined Digital after receiving a B.S.E.E. from Purdue University in 1989. Since then he has contributed to the NVAX and NVAX+ microprocessor designs and most recently was a member of the design teams for the integer execution unit and memory management unit of the Alpha 21164. Harry is a member of IEEE.

Paul E. Gronowski

Paul Gronowski joined Digital in 1984 after receiving a B.S. degree in electrical engineering from the University of Cincinnati. During the past 10 years with Digital Semiconductor, he has contributed to the design of several high-performance microprocessors. For the Alpha 21164 CPU, he was responsible for the integer execution unit and led the physical chip verification effort. He is currently responsible for the technical design and management of the next-generation processor. He is the co-author of several ISSCC papers and holds one patent.

Anil K. Jain

Anil Jain, a consulting engineer in Digital Semiconductor, led the implementation of the external interface unit on the Alpha 21164 microprocessor. Prior to this, he was the project leader for the floating-point unit on the NVAX microprocessor. He also made technical contributions on the CVAX microprocessor and on device modeling of Digital's first CMOS process. Anil received a B.S.E.E. from Punjab Engineering College (1978) and an M.S.E.E. from the University of Cincinnati (1980). He holds three patents.

Patricia L. Kroesen

A principal engineer in Digital Semiconductor, Patricia Kroesen is currently a circuit designer of the cache controller and I/O interface section for the next-generation Alpha microprocessor. In her work on the Alpha 21164, she was an implementer on the floating-point unit and optimized the clock distribution system for the PASS2 release. Since joining Digital in 1988, Patty has also worked on advanced development efforts of bipolar and GaAs chips. She has a B.S.E.E. from the University of Michigan and an M.S.E.E. from Polytechnic Institute. She holds one patent and has co-authored several papers.

Marc E. Lamere

A principal hardware engineer in Digital Semiconductor, Marc Lamere is currently a CMOS circuit designer for the next-generation Alpha microprocessor. In his work on the Alpha

21164, he was responsible for the integer execution unit shifter and other circuit designs as well as the physical and electrical verification of the chip. Marc joined Digital in 1984 as an ECL circuit designer on the VAX 9000 project and helped design custom and semicustom bipolar chips. He holds a B.S.E.E. (1983) from Rensselaer Polytechnic Institute and an M.S.E.E. (1988) from Northeastern University.

Bruce J. Loughlin

Consultant engineer Bruce Loughlin was responsible for the signal integrity design for the Alpha 21164 chip. Since joining Digital in 1975, Bruce has contributed to many projects, including the clock design of the DEC 3000 workstation, corporate FCC shielding strategy, design of the FDDI physical interface, and disk servo writing equipment for the R80 Winchester disks. From 1971 to 1975, Bruce was a member of the Eclipse design team at Data General. Prior to that, he was the vice president of engineering for Data Technology Inc., a company he cofounded in 1961. Bruce holds B.S. degrees in electrical engineering and mechanical engineering and an M.S. in aeronautics and astronautics, all from MIT.

Shekhar Mehta

Shekhar Mehta is a senior hardware engineer in Digital Semiconductor's High Performance Computing Group. He designed the miss address file on the memory subsystem of the Alpha 21164 CPU and was responsible for the electromigration checks of the chip. He is currently leading the design of the caches on a future Alpha microprocessor. Before joining Digital in 1988, Shekhar was an engineer at Larsen & Toubro, Bombay, India. He received an M.S.E.E. from the University of Wisconsin at Madison (1988).

Robert O. Mueller

Rob Mueller joined Digital in 1990 after receiving a B.S. in computer and systems engineering from Rensselaer Polytechnic Institute. As a senior hardware engineer in Digital Semiconductor, he is currently involved in the design and implementation of the pad ring for a new Alpha microprocessor. In his work on the Alpha 21164 chip, Rob contributed to the design, implementation, and electrical verification of the pad ring, the cache control, and the bus interface unit.

Ronald P. Preston

Ronald Preston is a principal engineer in Digital Semiconductor. Since joining Digital in 1988, he has worked on the design of several microprocessors and was the implementation leader for the instruction unit on the Alpha 21164. Ron was also responsible for the architecture and implementation of the issue/bypass/scoreboard logic. Ron is the coauthor of several articles on hot carrier analysis of CMOS circuits. He received a B.S.E.E. in 1984 and an M.S.E.E. in 1988, both from Rensselaer Polytechnic Institute. Ron is a member of Eta Kappa Nu and IEEE.

Sribalan Santhanam

Sri Santhanam received a B.E. in electrical engineering from Anna University, Madras, India, in 1987, and an M.S.E. degree in computer science and engineering from the University of Michigan in 1989. He joined Digital as a design engineer for Digital Semiconductor, responsible for the full-custom design and development of high-performance CMOS VLSI processors. Sri worked on the design of the floating-point unit of the 21064 CPU and the design of the cache control unit of the Alpha 21164 CPU. He is currently a member of the Low Power Alpha Group where he is involved in the design of a low-power microprocessor.

Timothy A. Shedd

Before receiving a B.S.E.E. from Purdue University in 1992, Tim Shedd was a co-op student at Digital working on several VAX CPUs as well as the floating-point unit of the 21064 CPU. He is now a hardware engineer in Digital Semiconductor's Advanced Development Group. Tim contributed to the circuit design of the Alpha 21164 microprocessor's instruction issue unit and is currently working on the memory management unit of the next-generation Alpha CPU. Tim is a member of Tau Beta Pi and Eta Kappa Nu.

Michael J. Smith

A principal engineer in Digital Semiconductor, Michael Smith was a member of the instruction unit design team for the Alpha 21164 microprocessor, responsible for floorplanning, logic, and circuit design. Prior to this, he was involved in the design of two memory controller/bus adapter chips for the VAX 4000 Models 300 and 600. Currently he is a member of the bus interface and instruction unit teams of the next-generation Alpha microprocessor. Michael joined Digital in 1986 after receiving a B.S.E.E. from the Rochester Institute of Technology.

Stephen C. Thierauf

Stephen Thierauf joined Digital in 1976. As a consulting hardware engineer in Digital Semiconductor, Stephen is currently responsible for I/O circuit design, on- and off-chip signal integrity, and I/O modeling for Alpha microprocessors and PCI peripherals. He has designed or led the circuit design for a number of high-performance telecommunication and peripheral chips. Previous responsibilities include I/O circuit design and system-level signal integrity analysis, micropackaging analysis, and micropackaging design for numerous high-performance microprocessors and peripherals. He is a member of IEEE.

TRADEMARKS

Digital is a trademark of Digital Equipment Corporation.

SPICE is a trademark of the University of California at Berkeley.