

 **TANDEM** COMPUTERS

A Highly Integrated, Fault-Tolerant Minicomputer: The NonStop CLX

Daniel E. Lenoski

Technical Report 87.5
November 1987
Part Number 11640



**A Highly Integrated, Fault-Tolerant
Minicomputer: The NonStop CLX**

Daniel E. Lenoski

Technical Report 87.5
November 1987
Part Number 11640



Table of Contents

| | |
|---|---|
| Abstract | 1 |
| System Organization | 1 |
| Processor Organization | 2 |
| CPU Architecture | 3 |
| CPU Chip Internal Architecture. | 4 |
| CPU Chip Internal Timing | 4 |
| Data Integrity Features | 5 |

A HIGHLY INTEGRATED, FAULT-TOLERANT MINICOMPUTER: THE NONSTOP CLX

Daniel E. Leñoski

Tandem Computers Incorporated
19333 Vallco Parkway
Cupertino, CA 95014

Abstract

The NonStop CLX, a highly integrated version of Tandem Computers' fault-tolerant NonStop architecture, is described. An overview of the system block diagram and a detailed description of the CLX processor is given. The processor is based on a custom CMOS chip-set developed using silicon compilation techniques. The CPU micro architecture is a hybrid of traditional minicomputer and high performance micro-processor architectures. This merging leads to a number of novel structures including a single static RAM array that is used as writeable control store, data cache and page table cache. The processor includes a high degree of fault checking in order to assure data integrity and fault-tolerant operation.

System Organization

The block diagram for the CLX™ is shown in Figure 1. The structure is that of a private memory multiprocessor and is similar to that of earlier Tandem NonStop™ systems (TNS) [1].

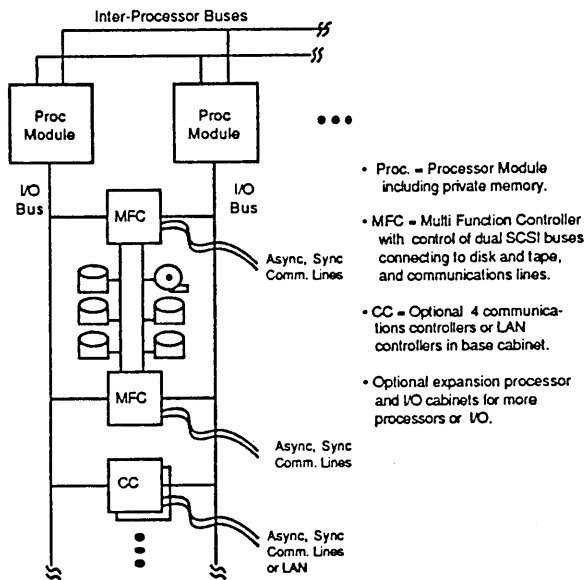


Figure 1. System Block Diagram

Each CPU communicates to other CPU's over two high speed Interprocessor buses (IPB's). Each bus operates synchronously on 16 bit wide data providing a peak bandwidth of 20 Mbytes / sec. The two buses transfer independently providing a total of 40 Mbyte / sec of bandwidth to the maximum of six processors in a CLX system.

CPU's communicate with I/O devices either through a local I/O bus or through the IPB to another CPU and its I/O bus. Each CPU contains a single asynchronous, burst multiplexed I/O bus which transfers data at a maximum rate of 3.7 Mbyte / sec. to a maximum of 16 controllers. I/O controllers are dual-ported and can be driven by either of the CPU's to which they are attached.

The CLX utilizes a Multi-Function Controller (MFC) based on a Motorola 68010 to control dual Small-Computer-System-Interfaces (SCSI) containing up to five disk drives and one tape drive. The MFC also contains 2 asynchronous and 1 synchronous communication lines together with a dedicated maintenance communication line. The MFC runs it's own real-time operating system kernel that coordinates the independent disk, tape, comm. and maintenance tasks.

Not shown in Figure 1 are the system maintenance buses. These buses allow maintenance and diagnostic information to travel between the system control panel, the processors, the multifunction controllers and the power and environmental monitors.

A fully populated single cabinet system contains two CPU boards with optional expansion memory, six I/O controllers, five 145 Mbyte disk drives and one cartridge tape drive. Fault-Tolerant (dual) power supplies and cooling fans are also provided within the cabinet. The entire system is designed to operate within the power, noise and size requirements of an office environment. Expansion beyond a single cabinet is handled with additional I/O or CPU cabinets.

The message based Guardian 90™ operating system [2] provides all processes with a seamless interface to all CPU's and peripherals. All resources are viewed in a uniform way regardless of their physical position within the system. Networking software extends this transparency between systems. Enhanced diagnostic software together with careful mechanical and electrical design of each customer replaceable unit allows 98% of all component failures to be serviced by the user.

The system is optimized for On-Line Transaction Processing.

The best measure of performance of such an OLTP system is Transactions Per Second (TPS) [3]. For a specific benchmark environment, the TPS rating gives a true measure of system performance that is independent of processor type and includes the performance of the I/O subsystem, the operating system and application code. The most widely used transaction benchmark is *debit-credit* (ET1). Each transaction in the ET1 benchmark represents a typical bank teller account update. The CLX system has demonstrated a performance of 2.5 TPS per processor on this benchmark (with less than 2 second response time for 90% of all transactions). This rating includes the use of a full SQL relational database and transaction logging. For OLTP applications it has also been demonstrated that the NonStop architecture shows linear performance growth in excess of 32 processors[4]. For the CLX this implies a 1-6 processor system can deliver between 2.5 to 15 TPS.

Fault-Tolerance is provided through module redundancy and fail-fast module operation. Module redundancy implies that each unit must be replicated so that the system can continue operation in the face of a failure of one module. The system as a whole will only fail if there is a second failure within the window of time it takes to repair an initial failure. This increases the mean time to failure of the system orders of magnitude greater than any individual module. Note that redundancy does not imply replicated units need to be idle. In the case of NonStop CPU's, checkpointing [5] allows CPU's to work independently, but still have all the necessary state to recover in the event of a failure of a primary CPU. Likewise, under normal conditions IPB's transfer data simultaneously and mirrored disks perform independent reads.

Fail-fast or fail-safe operation is required of each module so that the integrity of data is not compromised by an undetected faulty module. This implies that replicated components must fail in such a way as to stop operating. This assures that faulty outputs do not corrupt the module's back-up or both sections of a redundant system interface.

Processor Organization

The TNS processor architecture is optimized to support OLTP and a message based OS environment. In this environment operations such as interprocessor bus transfers and block moves are more important than in typical processors. Conversely, operations such as floating point arithmetic are not as critical to overall performance.

The CLX supports the entire instruction set of the TNS architecture[6]. The architecture defines a stack based CISC processor together with its interprocessor and I/O buses. The definition includes an explicit register stack used to accelerate stack computations and to hold array indices. Sixteen and thirty-two bit operations with thirty-two bit addressing are defined by the instruction set. The machine includes complex instructions that support OS operations, block operations, and decimal and floating point data types.

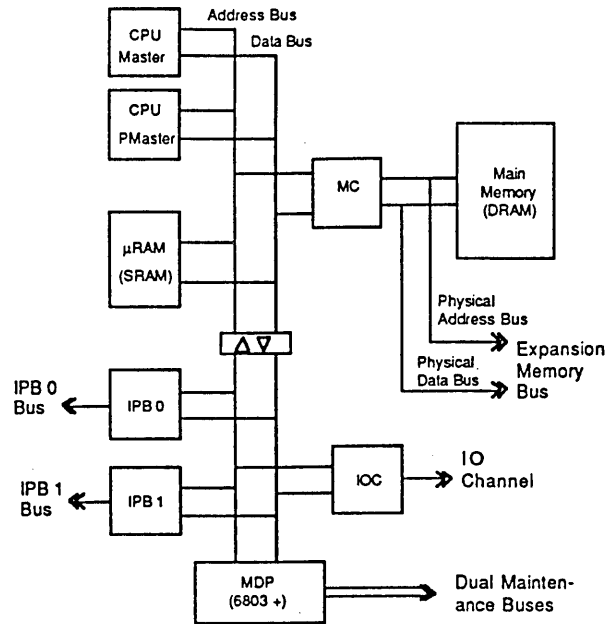


Figure 2. Processor Block Diagram

The processor block diagram is shown in Figure 2. The processor logic is implemented primarily within six custom CMOS chips. The chip set is fabricated in a 2μ drawn gate, n-well CMOS process and was designed using a silicon compiler supplied by Silicon Compiler Systems Corporation [7]. Some of the relevant statistics of chip set are shown in Table 1. The four chip types were implemented, inclusive of definition through to functionally correct silicon in the laboratory, by 6 engineers in 15 months.

Table 1. CPU Chip-Set Statistics

| | CPU | MC | IOC | IPB |
|-------------------|--------|--------|--------|--------|
| Vertical (mils) | 495 | 352 | 309 | 300 |
| Horizontal (mils) | 496 | 367 | 311 | 300 |
| Transistor Sites | 78,301 | 29,408 | 14,785 | 19,565 |
| Transistors | 60,469 | 25,653 | 12,177 | 18,872 |
| Power Dissipation | 2.0 W | 1.3 W | 0.9 W | 0.9 W |
| I/O pins | 110 | 95 | 71 | 95 |
| Power pins | 15 | 13 | 11 | 10 |
| Ground pins | 15 | 12 | 11 | 10 |

The two CPU chips are identical and run in lock-step to form a totally self-checked module. These chips include the complete CPU function, and work together with a single bank of static RAM (μRAM) that functions as the microcode control store, page table cache and data/instruction cache. The μRAM is realized with thirty 16k x 4, 35ns SRAM parts organized as two banks of 16k by 60 bit words. This provides for 14k words of microcode and scratch pad memory, 2k words of page table cache with two entries per word, and 16k words of instruction / data cache with four bytes per word.

The MC chip includes the control and ECC logic (SEC/DED) to interface to the 4 Mbytes of on-board dynamic memory and 8 Mbytes of expansion memory. This chip also includes FIFO's to buffer data to and from main memory using nibble mode accesses. The part also features a wrap around mode for high-speed memory to memory block transfers.

There is one IPB chip per interprocessor bus. Each chip contains a sixteen word in-queue and a sixteen word out-queue. These queues work with on-chip state machines to permit the sending and receiving of interprocessor message packets asynchronous to processor execution.

The IOC chip contains the data latches and control logic to control a burst-multiplexed, asynchronous I/O bus. The I/O bus is primarily controlled directly by the microengine, but can handle DMA transfer polling and selection without microcode intervention. It also includes priority encoding logic to aid in servicing I/O interrupts.

The final component of the processor is a maintenance and diagnostic processor (Motorola 6803). This processor provides overall control of the main processor, and a diagnostic and error reporting path for the main processor through the maintenance buses.

CPU Architecture

The CLX CPU architecture is a hybrid of minicomputer and microcomputer architectures. The CPU chip's external interface is similar to a microprocessor. It contains one address bus, one data bus and one status bus along with miscellaneous signals such as an interrupt request, memory wait controls and tri-state input controls. A closer look, however, reveals that the address bus is only 18 bits wide, and the data bus is 60 bits wide. This structure is actually more akin to that found in minicomputer architectures. The CLX CPU external interface is the merging of many buses that would normally be separate in a minicomputer architecture. In particular, an external cycle on the CLX can have the following meanings:

- Microcode control store access.
- Instruction or Data cache access.
- Page Table cache (TLB) access.
- Main Memory access.
- Microcode Scratch Pad Memory access.
- Special Module (IPB, IOC, MDP) access.

This merging of buses reduces the cost of the processor in terms of the number of static RAM parts and their associated support logic, and the pins and packaging of the CPU chip itself. If implemented blindly however, this merging would lead to a significant degradation in performance. In order to reduce the bandwidth required on these buses and thus the performance impact, a variety of techniques were used. These include:

- Utilization of a small on-chip microcode ROM.
- Use of a virtually addressed cache.
- Use of nibble mode DRAM with block operations to the main memory controller.
- Higher-level control operations for special modules.

The on-chip ROM (μ ROM) has the biggest effect on reducing the performance impact of the merged buses. The μ ROM contains 160, 54 bit words of microcode with an identical encoding as external microcode. This ROM is addressed by either the microcode PC (μ PC) or through an explicit index specified in the previous line of microcode. The μ PC addressing is used to implement the inner loops of IPB and IOC transfers, cache filling routines and block memory moves.

The index addressing is used throughout the microcode to implement short common sequences such as instruction prefetch, interrupt testing, and cache loads and stores to the top of the register stack. The explicit indexing acts like a microcode call, but does not modify the μ PC. It simply overlays the microcode that would have been fetched from external control store with a line from the μ ROM. The index specifier of the overlaying line does not conflict with other microfields (unlike a call to the μ ROM). Index addressing is also used in critical instructions where such a micro branch would be costly. The explicit addressing provides for maximal sharing of small amount of μ ROM code without the overhead of a μ Code call and return.

The virtually addressed cache reduces the number of page table accesses, and thus the required band-width to the shared μ RAM. Likewise the use of block-mode commands to the memory controller reduces the number of memory commands needed during cache filling and block moves. Finally, the use of higher-level commands to the IPB and IOC reduce the control transfers needed to receive and transmit data to these devices. The on-chip μ ROM together with these other features reduces the penalty of using a single bus approach from over 50% to less than 12%.

The primary alternative to the μ ROM used on the CLX is an emulation scheme where a set of emulation instructions and a subset of the CISC instructions are implemented entirely by an internal ROM. The μ ROM scheme has two primary benefits when compared with the numerous emulation schemes that have been reported in the literature[8][9]: First, it provides much higher performance when the amount of ROM space is limited relative to the number of instructions that must be implemented. Second, the dispatch of each instruction is to external writeable control store enabling any ROM code errors to be corrected externally (albeit with some performance penalty).

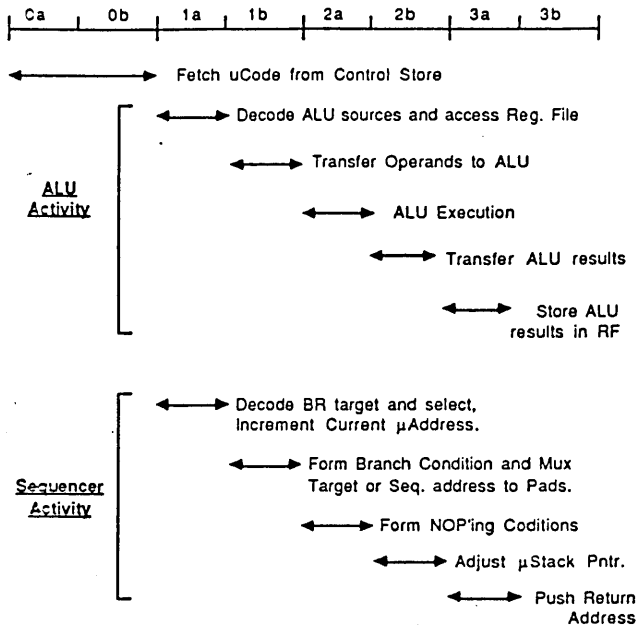


Figure 4. Microcode Pipeline

The macrocode pipeline is controlled by microcode and executes at a maximum rate of one macro instruction per two micro cycles. The general flow of macro instructions is illustrated in Figure 5.

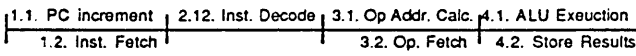


Figure 5. Macroinstruction Pipeline

This macro pipeline is not fully realizable, however, because there is not enough resources to perform the operand calculation and fetch of the instruction in parallel with the instruction address calculation and fetch for the instruction two downstream. One solution to this problem would be to increase the number of microcycles per macropipeline stage, but this would result in a penalty for simple instructions that do not use memory operands. Another solution would be to allow short instructions to omit the operand fetch, but this would require that the microcode for short instructions control the operand fetch part of the macropipeline. This, in turn, would require a deeper microcode pipeline with the undesirable increase in the penalty for pipeline breaks and interlocks. The final solution used in the CLX is to have the contents of the next macro instruction register modify the semantics of certain microcode operations. In particular, when the microcode sequence of the instruction preceding a short instruction specifies the operand calculation and fetch for the short instruction this operation is modified to actually do an instruction calculation and fetch of the instruction two downstream from the short instruction. Since these two operations are similar (both have an address calculation using the DALU and a memory fetch from cache) the only additional logic is a decode of the next instruction register to single out all short instructions.

The overall result is a fast pipelined micro engine which controls the macro pipeline, but whose operations can be altered by the contents of the macro pipeline. This configuration gives high speed execution of complex instructions without penalizing simple instructions. Some sample timing for simple instructions is given in Table 2.

Table 2. Simple Instruction Execution Times.

| | |
|---------------------------|-----------------|
| Register Stack Operations | 2 microcycles |
| Memory to Stack | 3 microcycles |
| Stack to Memory | 5 microcycles |
| Branch Taken/Not Taken | 3/3 microcycles |

Data Integrity Features

As stated earlier, fail-fast operation of hardware modules is essential to NonStop execution to be effective in providing fault-tolerance at the system level. Fail-fast operation requires that faults are detected and that the processor is halted upon detection of a fault. The CLX CPU supports a very high degree of fault coverage using a variety of error checking strategies.

The CPU chip itself is covered by a duplicate and compare scheme. This scheme was chosen because it minimizes the amount of internal logic required for a high degree of coverage, and it maximizes the utilization of existing library elements in the silicon compiler CAD system. The implementation of the CPU's duplicate and compare logic is shown in Figure 6.

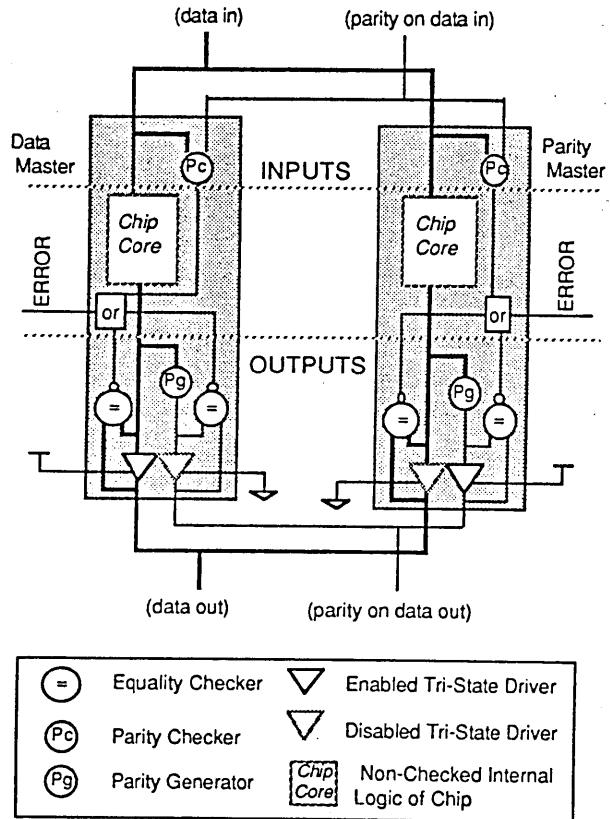


Figure 6. CPU Cross-Coupled Checking.

This scheme improves the fault coverage of other duplicate and compare schemes reported in the literature [10][11] by providing for a cross-coupling of data and parity outputs. One chip is designated the data master and drives all data outputs while the other is designated the parity master and drives all parity outputs. This insures that both chips' outputs and checking logic are active, and that latent errors in the checking logic can not lead to an undetected double failure. The parity out of the CPU also functions to cover the address and data lines connecting the CPU to other parts of the processor and the μ RAM.

Within the memory system ECC with encoded address parity is used to provide checking of all memory system data paths. In addition, redundant state machines are used within the MC chip and in the external RAS/CAS generation logic. The state transitions of these machines are encoded into CRC registers whose outputs are compared. The resulting structure contains a high degree of fault coverage for both the data and control sections of main memory.

The IOC and IPB provide for parity protection of the data and control lines that they are interfaced to. In addition, they are protected by end-to-end checksums supported in software that guarantee the integrity of their respective buses, I/O controllers and devices.

Conclusion

The NonStop CLX is a prime example of the benefits of using high density ASIC technologies in the design of high performance minicomputers. The architecture of such machines must blend the structures used on previous board level minicomputers as well as the structures used in VLSI microprocessors in order to be effective. This includes the matching of the pin limits of VLSI to the wide buses utilized on minicomputers, and the use of a high degree of pipelining throughout the design. The machines resulting from this merger can provide low-cost through integration while maintaining or increasing the performance, data-integrity and fault-tolerance of previous machines.

Acknowledgement


The CLX system is the result of the hard work of over 35 hardware and software engineers together with an even greater number of support personnel. This work was greatly aided by the leadership and environment provided by Tandem management.

References

- [1] Tandem Computers Inc., *Introduction to Tandem Computer Systems*, Tandem Part No. 82503, March 1985.
- [2] Tandem Computers Inc., *Guardian Operating System Programmer's Guide*, Tandem Part No. 82357, March 1985.
- [3] Anon et al., "A Measure of Transaction Processing Power," *Datamation*, April 1, 1985, pp. 112-118.

- [4] R. Horst, T. Chou, "The Hardware Architecture and Linear Expansion of Tandem NonStop Systems", *Proceedings of the 12th International Symposium on Computer Architecture*, June 1985.
- [5] J. Bartlett, J. Gray, R. Horst, "Fault Tolerance in Tandem Computer Systems", *The Evolution of Fault-Tolerant Computing, Vol 1*, ed. A. Avizienis et al., May 1987, pp. 55-76.
- [6] Tandem Computers Inc., *System Description Manual*, Tandem Part No. 84017, October 1986.
- [7] S. C. Johnson, "Silicon Compiler Lets System Makers Design Their Own VLSI Chips", *Electronic Design*, Oct. 4, 1984, pp. 168-181.
- [8] R. M. Supnik, "MicroVAX 32, A 32 Bit Microprocessor", *IEEE Journal of Solid-State Circuits*, October 1984, pp. 675-681.
- [9] H.H. Chao et al, "Micro/370: A 32-bit Single-Chip Microprocessor", *IEEE Journal of Solid-State Circuits*, October 1986, pp. 733-740.
- [10] D. Johnson, "The Intel 432: A VLSI Architecture for Fault-Tolerant Computer Systems," *Computer*, August 1984, pp. 40-48.
- [11] D. Ajmera et al, "Bipolar Building Blocks Deliver Supermini Speed to Microcoded Systems", *Electronic Design*, Nov. 15, 1987, pp 230-246.
- [12] J. Bartlett, "A NonStop Kernel", *Proceedings of the Eighth Symposium on Operating System Principles*, Dec. 1981, pp. 22-29.
- [13] R. Horst and S. Metz, "A New System Manages Hundreds of Transactions / Second", *Electronics*, April 19, 1984, pp. 147-151.
- [14] Anon, "Tandem Makes a Good Thing Better", *Electronics*, April 14, 1986, pp. 34-38.

TM Tandem, NonStop, Guardian 90 and CLX are trademarks of Tandem Computers Inc.

Distributed by
 **TANDEM COMPUTERS**
Corporate Information Center
19333 Vallco Parkway MS3-07
Cupertino, CA 95014-2599

