TMS 99000 SYSTEM BRIEF

A THIRD GENERATION OF MICROPROCESSOR COMPONENTS, SOFTWARE, AND SUPPORT

The new TMS 99000 microprocessor family has harnessed advanced design concepts into a memory-intensive architecture, which represents true third generation features and performance. The TMS 99000 is a third generation descendent of the TMS 99000 16-bit microprocessor, sharing the same advanced memory-to-memory architecture of the TMS 9900. With TMS 99000 instruction set as a superset of the TMS 9900, full object code compatibility is maintained. Speed of the TMS 99000 microprocessors is from 5 to 12 times faster than the first generation TMS 9900, 2 times faster than the second generation TMS 9995, and up to 3 times faster than other currently available processors.

However with today's computing needs, raw processor performance alone does not determine microprocessor selection. Memory costs, software adaptability, and a clearly defined migration path among cost/performance options all contribute to design decisions. The TMS 99000 family was conceived to facilitate a wide spectrum of cost/performance options – from small systems to large, multiprocessor environments.

Cost/performance flexibility is attained by providing a path for migrating functions from software to higher performance hardware designs. Additionally, a machine-cycle efficient memory interface allows a choice of very fast (<60 ns) memory devices for maximum performance or optionally, slower, less expensive memory devices. Thus, through a selection of software-to-hardware migration and memory options, flexibility in configuration for optimum system cost and performance is attained. This capability enables designers to address the dynamic marketplace problems of the '80s.

APPLICATION SPECIFIC CPU'S

Flexibility within the TMS 99000 family of CPU's is enhanced through different versions of the same architecture with functions added to the same base instruction set. Through these added functions, a TMS 99000 CPU is tailored to specific tasks. The first members of the TMS 99000 family to be introduced are a base line processor (TMS 99105), a floating point processor (TMS 99110), and a high level language processor, which contains commonly used utilities of the PASCAL language (TMS 99120). Future processors can include support for functions such as linked lists, business applications, and scientific applications. With this approach, the TMS 99000 family of processors can satisfy the requirements of a broad price/performance spectrum while maintaining upward software compatibility.

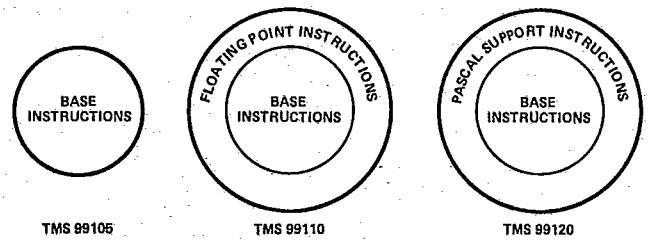


FIGURE 1 – TMS 99000 FAMILY PROCESSORS

Key features of the TMS 99000 family of microprocessors include:

- Unique VLSI memory-to-memory architecture
- Performance
  - 167 ns machine state cycle time
  - Single machine state cycle memory access
- 85 instructions
- 256K byte memory addressability
- Instruction privileging
- 16 hardware prioritized vectored interrupts
- Serial or parallel I/O transfers
  - I/O data bit, byte, and word addressable
- Attached processor and computer interface
- 16 bus status codes
- DMA interface
- Single 5-volt operation
- 40-pin package
- N-channel silicon gate SMOS technology
- On-chip clock generator

Peripheral Family

The TMS 99000 family includes members of the powerful 9900 family of peripherals: data communications devices, a floppy disk controller, an IEEE-488 bus interface, video timers and controllers, a powerful color video display processor, and a wide range of interface support circuits.

The 99000 family of peripherals will be expanding to meet the emerging system requirements of the '80s. Devices in definition and development include man-machine interface, speech processing, data communications peripherals such as a high-speed multiprotocol processor, and mass storage support devices such as Winchester disk controller chip sets. Together the TMS 99000 family of peripherals will comprise a powerful family of VLSI support devices, which will in turn be supported with TI's component software. Component software modules will be migrated to the silicon of the peripheral function achieving an unsurpassed level of functionality.

Development Tools

The TMS 99000 family is today supported by an unmatched existing base of software and hardware development systems and software support tools. The powerful Advanced Microprocessor Prototyping Laboratory (AMPL) has seen wide use in development of software and hardware for the TMS 9900, TMS 9940, and the TMS 9995 microprocessors. The AMPL development systems allow the system designer to utilize the efficiencies inherent in higher-level languages such as Microprocessor PASCAL, thus increasing programmer productivity. AMPL system options range from low-cost single user systems to cost efficient multi-user hard disk based systems. A photo of the AMPL development labs and options available is shown in Figure 2.

The powerful library of component software was conceived for the TMS 9900 microprocessor. Again due to maintaining full object code compatibility with the TMS 9900, the TMS 99000 can efficiently use this existing software base.

Component software consists of software application modules, which are tied together through a real-time executive "bus" just as peripheral devices are tied together through a common hardware bus. The modularity of component software modules allows the system designer to select only the software functions needed for a specific application without the burden of purchasing a large application program which may contain more functions and consume more memory space than necessary. This modularity also supports eventual migration of the software function to a pure silicon solution, i.e., hardware.

Customer Support

To help system designers take full advantage of the technological capabilities offered by Texas Instruments, a number of Technology Centers have been strategically located through the United States. These Regional Technology Centers (RTCs) provide training in the use of TI products — from microprocessors to speech. Courses and seminars are provided in systems design, software development, and technological overviews. In addition, design services for customer applications may be contracted.

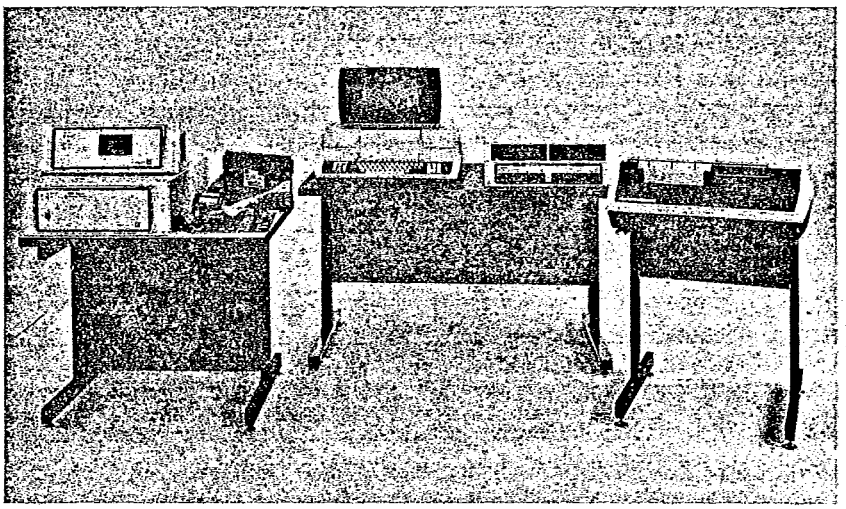


FIGURE 2 — AMPL MICROPROCESSOR PROTOTYPING LABS

THE TMS 99000 MICROPROCESSOR FAMILY — FUNCTIONAL EXTENSIBILITY

As previously discussed, a predefined path for migrating software functions to hardware solutions has been designed into the TMS 99000 family CPU's. This migration path, illustrated in Figure 3, includes: an innovative concept called macrostore, attached processors, and attached computers. Each progressive step in the migration path can result in improving system performance. The impact of these migration steps is represented graphically in Figure 4. Note that a function execution is segmented into four definable steps: invocation, parameter passage, execution, and completion processing.

The migration of the software emulation routine to on-chip macrostore can result in significantly reduced execution time. Execution time is further reduced by performing the function in an attached processor, which essentially executes functions in line with the code execution of the host TMS 99000 CPU. Attached computers remove the effective function execution time completely by allowing concurrent processing with the host TMS 99000. Concurrency is possible due to the private memory capability of the attached computer thus allowing the host TMS 99000 continued access to the memory bus.

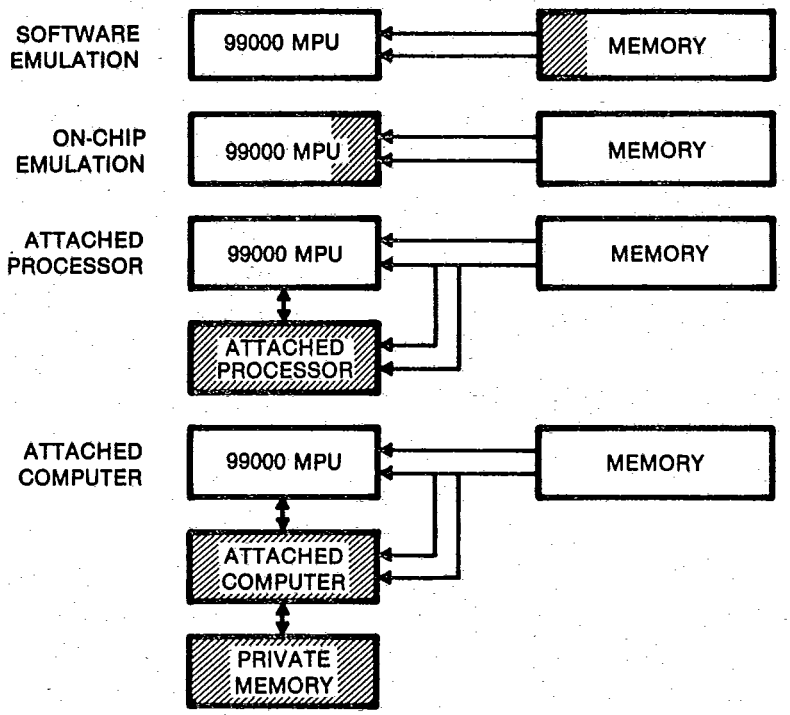


FIGURE 3 – FUNCTION MIGRATION PATHS

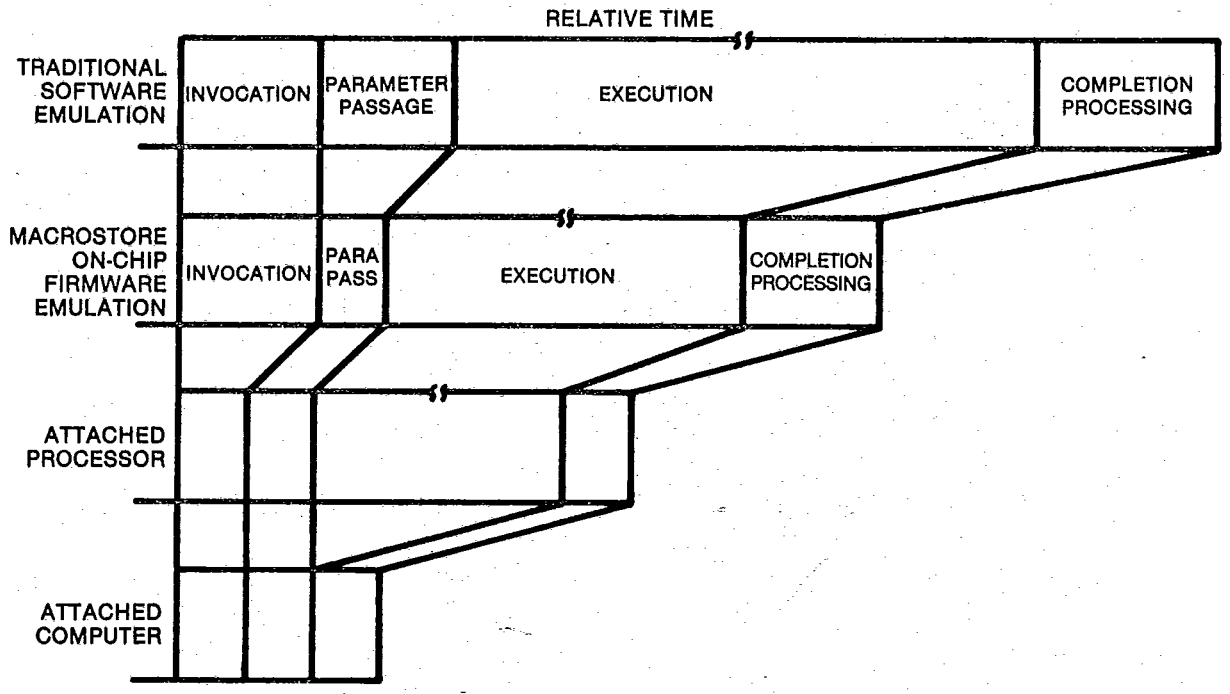


FIGURE 4 – PERFORMANCE IMPACT OF FUNCTION MIGRATION

During normal instruction, fetch, and execute sequences the TMS 99000 family processors perform a decision process shown in Figure 5.

When each instruction is fetched from main memory, several paths of execution may be followed. If the opcode is a standard instruction, i.e., a native instruction of the processor, the instruction is executed through normal on-chip sequences of microcode. If the opcode is not standard, i.e., an illegal or otherwise undefined opcode, the TMS 99000 will test the attached processor/computer interface to determine if an attached processor/computer is waiting to perform the function. If the attached processor or computer is present, the attached processor or computer is invoked; and the TMS 99000 either waits or continues depending on whether an attached processor or an attached computer is present. If no external processors are present, a reference is made to the separate macrostore memory space to determine if an emulation routine resides for that specific opcode. If the emulation routine is found in macrostore, execution control is transferred to this separate memory space to complete execution of the function or instruction — much like a subroutine call. If macrostore emulation is not present, an interrupt is asserted allowing either for emulation of the function or instruction in main memory or handling the non-standard opcode as an opcode violation.

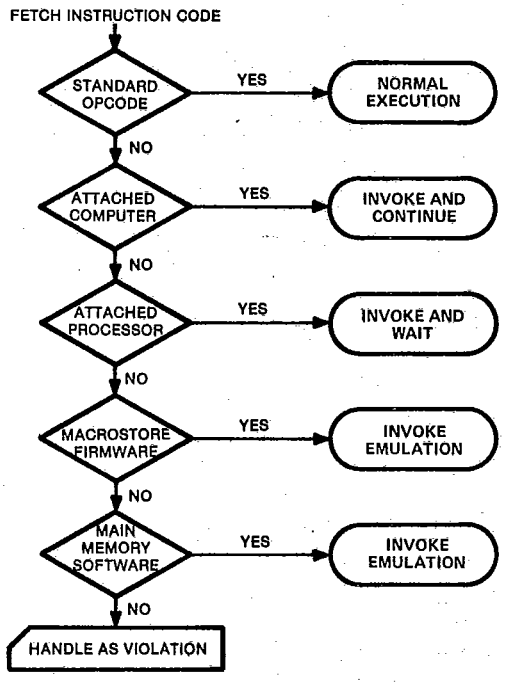
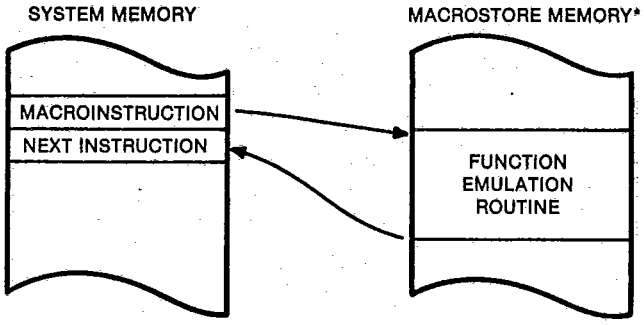


FIGURE 5 – STRUCTURED CONTROL FLOW

The TMS 99000 family of CPU's performs this control flow with no impact to instruction execution performance. All control decisions are performed during the normal instruction decode time.

Macrostore

As previously discussed, macrostore is a logically separate memory address space, which provides for emulation of added functions and instructions. Consisting of 64K bytes, macrostore enables the system designer to tailor system functionality to specific system requirements. New functions and instructions are implemented in normal TMS 99000 machine codes for ease of implementation and efficient use of memory. Figure 6 illustrates how macrostore emulation occurs in the logically distinct address space. During the decision process previously discussed, control is transferred to provide instruction fetch and execute cycles out of the macrostore memory space. The emulation routine, implemented with standard TMS 99000 instructions, is exited in a manner similar to a return from subroutine. Execution is continued out of main memory at the location specified by the program counter.



* A LOGICALLY DISTINCT MEMORY SPACE

FIGURE 6 – EMULATION IN MACROSTORE

Of the 64K byte macrostore logical address spaces, the first 4K bytes are reserved for on-chip integration of macrostore memory. All TMS 99000 processors contain 32 bytes of macrostore RAM on-chip. The TMS 99110 and TMS 99120 processors contain an additional 1K bytes of macrostore ROM. In the TMS 99110, this ROM contains emulation routines for single precision floating point instructions. In the TMS 99120, this ROM contains run-time support for the PASCAL high-level langauge. The remaining 60K address space allows off-chip expansion as shown in the address map of Figure 7.

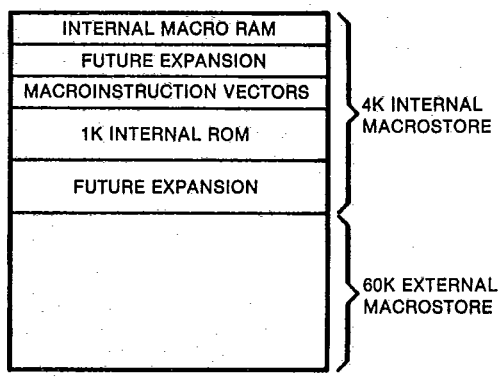


FIGURE 7 – MACROSTORE ADDRESS SPACE

Macrostore functions and instructions are powerful tools for system designers. To the system user, the implementation of instructions or functions in processor microcode or in macrostore emulation is virtually indistinguishable. To the system software designer, macrostore implemented instructions require no special conventions as they may be coded exactly as a subroutine. However, because this "subroutine" resides in a separate logical address space, the function is transparent and not subject to modification by the system user. The viewpoints perceived by both the system designer and system user are illustrated in Figure 8.

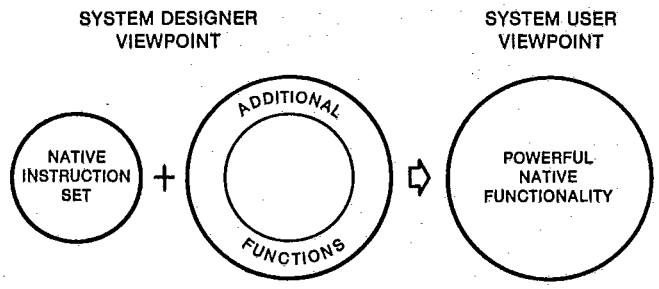


FIGURE 8 – MACROSTORE VIEWPOINTS

Attached Processors and Computers

The next step in functional migration is the migration of the function to a dedicated processor or computer. The distinction between attached computer and attached processor is the manner in which the host TMS 99000 reacts after invocation of a process in the attached element. Attached processors essentially replace in-line execution of code by the host TMS 99000 and, as such, halt the host until function execution is complete. Thus the environment is sequential in nature.

Attached computers, on the other hand, usually have the facilities of a private memory system thus allowing them to release the host TMS 99000 after function invocation. This means concurrent execution of the attached computer with the host TMS 99000. Once the task of the attached computer is complete, the attached computer signals the host TMS 99000 completion through an interrupt, DMA or the attached processor interface. Both attached computers and attached processors utilize a common interface with a TMS 99000 family processor as shown in Figure 9.

The ability to configure an attached hardware processor element to operate concurrently or sequentially provides the means to configure the system that best meets the system and function requirements. Some functions are best implemented in a sequential manner such as floating point processors. Some functions, such as communications channel controllers, are more efficiently implemented as concurrent processes.

The invocation stage for attached processors as well as attached computers provides for a full transfer of the TMS 99000 state vector to the attached processor or computer. The exchange of only three internal TMS 99000 registers makes this capability possible. In conventional register-based machines, the overhead of transferring the internal register set of the host processor would have a serious performance impact. The transfer of the entire state vector is critical to maintaining system integrity of both concurrent and sequential processes.

- MEMORY BUS
- ATTACHED PROCESSOR PRESENT (APP)
- INTERRUPT PENDING (INTP)
- BUS STATUS CODES

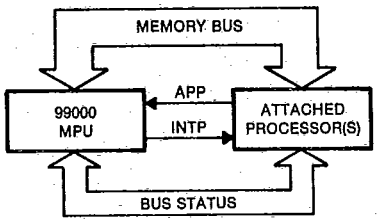


FIGURE 9 – ATTACHED PROCESSOR/COMPUTER INTERFACE

The TMS 99000 microprocessor family provides a structured path for system upgrades and cost reductions. A range of cost and performance options can be considered while maintaining the same level of functionality. As an example of this, consider the implementation of a floating point instruction set. At the first level of implementation, this could be an emulation routine in main memory entered through a system level interrupt when the floating point opcodes are fetched and decoded. The second level is the migration of floating point instruction emulation to macrostore, on or off-chip. This was done on the TMS 99110 floating point microprocessor. A floating point attached processor provides the best performance (with specialized hardware) for floating point instructions. During such a migration scenario no modification to user software need to be made as the floating point opcodes would remain the same. To the system user, the floating point instruction set would appear identical, regardless of the method of implementation.

MEMORY INTERFACE

Another level of cost/performance efficiency, which has been designed into the TMS 99000 microprocessor family, is the ability to configure a memory system consistent with the overall system cost and performance goals. This is a result of an efficient memory interface, which allows the use of fast memory for critical functions or slower, less expensive memory for main memory store. Appreciable performance gains through the use of high-speed memory allow cache memory techniques to become a viable option for configuring high performance systems while reducing the overall memory costs.

The memory cycle time of the TMS 99000 microprocessor can be as fast as 167 nanoseconds with a minimum memory access time of approximately 60 nanoseconds. For slower memories any number of additional memory cycles may be inserted. The result of this memory cycle time is that overall memory bus bandwidth is increased.

Another element contributing to memory interface efficiency is an intelligent instruction prefetch mechanism, which practically eliminates "dead" cycles on the memory bus. Thus internal, or ALU cycles, are peformed concurrently with memory bus activity. This prefetch method, illustrated in Figure 10, is termed intelligent because the actual program counter location is fetched for branch and jump instructions but not necessarily the next sequential location. The prefetch queue is one instruction deep to eliminate complex queue management, which adversely impacts performance.

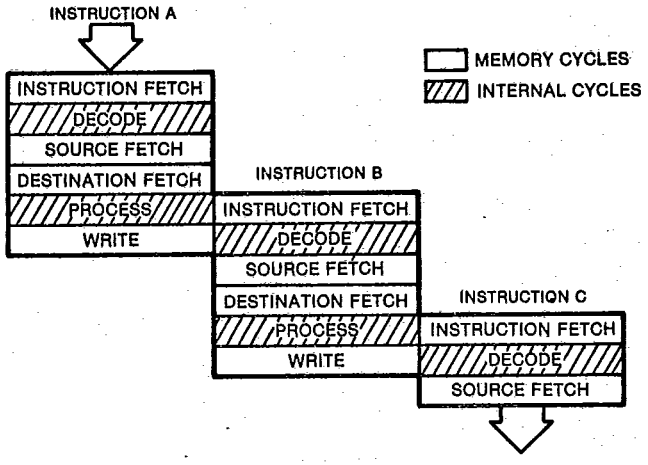


FIGURE 10 – INTELLIGENT INSTRUCTION PREFETCH

TMS 99000 MEMORY SYSTEMS

The ability of the TMS 99000 family of microprocessors to access memory at very fast cycle times provides the system designer wide latitude in configuring the memory subsystem to support cost/performance goals of the entire system. Because memory costs are predominant in most systems, this capability is essential in meeting design goals.

The result of this efficient memory interface is that TMS 99000 systems support hierarchial memory systems. Figure 11 illustrates the structure of this memory hierarchy. This hierarchy supports a large variation in processor to backing-store mass storage transfer capabilities. In addition, this hierarchy provides growth in memory address reach, supporting memory systems ranging from less than 64K bytes up to 16M bytes of memory.

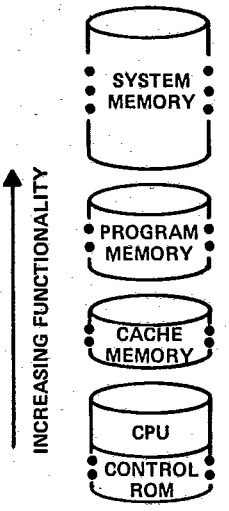


FIGURE 11 – MEMORY HIERARCHY

The TMS 99000 family provides for cost effective growth in physical main memory size through functional segmentation, paging, and use of memory mappers. By use of the status bit output pin, up to 128K bytes of physical memory may be addressed in both the main memory and macrostore memory space for a total of 256K bytes of physical memory. Additional physical memory may be addressed through use of the bus status code outputs to select either a code segment or a data segment, each segment being 64K bytes. These segments may also be paged with the status bit output to provide 128K bytes of physical memory within each segment. Figure 12 illustrates the manner in which large physical addresses may be accessed by the TMS 99000 processor. Note that through the addressing methodologies just discussed, the TMS 99000 family microprocessors can address up to 384K bytes of physical memory.

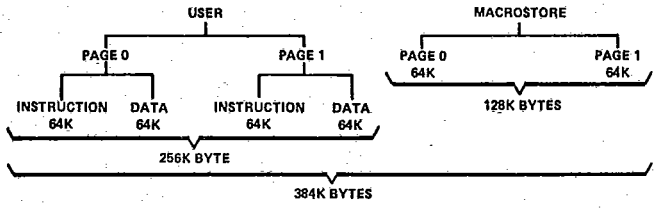


FIGURE 12 – TMS 99000 EXTENDED ADDRESSING

For systems requiring up to 16M bytes of physical memory, a TIM99610 memory mapper may be used. The TIM99610 memory mapper provides sixteen, 12-bit mapping registers which expand the TMS 99000 processors four highest order address lines to twelve. This provides a mapping resolution of 4K byte map boundaries for maximum flexibility. Figure 13 illustrates a TMS 99000 microprocessor system utilizing a TIM99610 memory mapper in a 16M byte memory system.

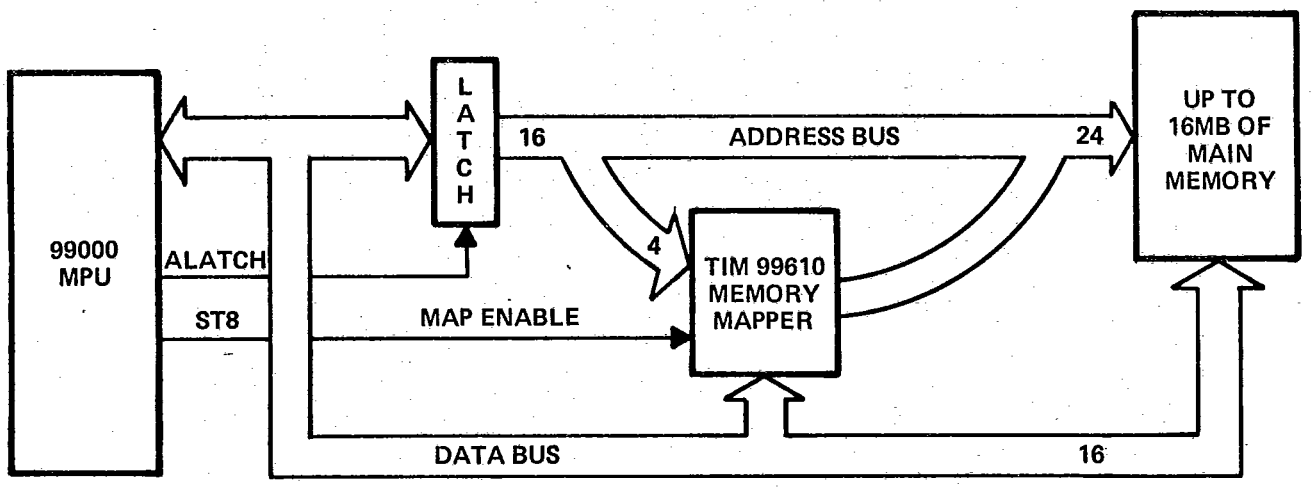


FIGURE 13 – 16M BYTE MEMORY INTERFACE USING A TIM99610 MEMORY MAPPER

Because time spent in memory transfers is the major factor affecting system performance, it is desirable to achieve the fastest memory cycle time possible. In large memory systems, it is often inconsistent with system cost goals to populate large amounts of high-speed memory. With the TMS 99000 it is not necessary to utilize all high-speed memory to obtain the performance goals. Typically 80 percent of an application's processing time is spent in 20 percent of the code. Through use of cache memory and functional partitioning of code segments, a performance advantage is obtained at the established system cost goal. When caching or partitioning techniques are utilized, an effective memory cycle time which falls between the cycle time of the high speed and slow speed memory is obtained. Figure 14 illustrates a bar graph of effective memory cycle times for the TMS 99000 with a main memory cycle time of 333 nanoseconds and TMS 99000 effective cycle times utilizing zero wait-state cache with hit ratios of .9 and .6 respectively. For comparison, the zero wait-state memory cycle times are provided for the Motorola 68000 and the Intel 8086-10 microprocessors. (Note: Hit ratio is the percentage of memory cycles which occur in the cache or high-speed memory.) An illustration of a TMS 99000 system with a cache memory system is shown in Figure 15.

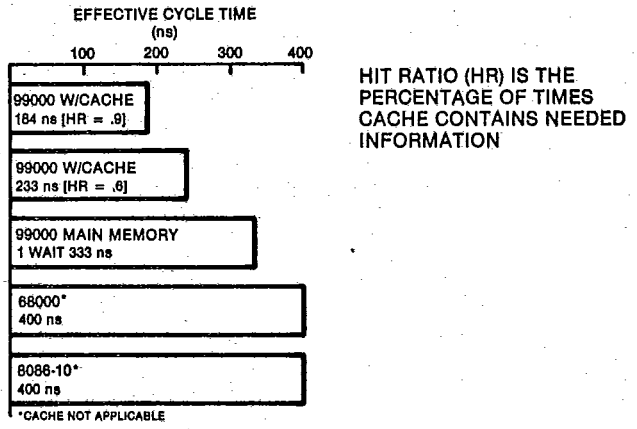


FIGURE 14 – EFFECTIVE MEMORY CYCLE TIME COMPARISONS

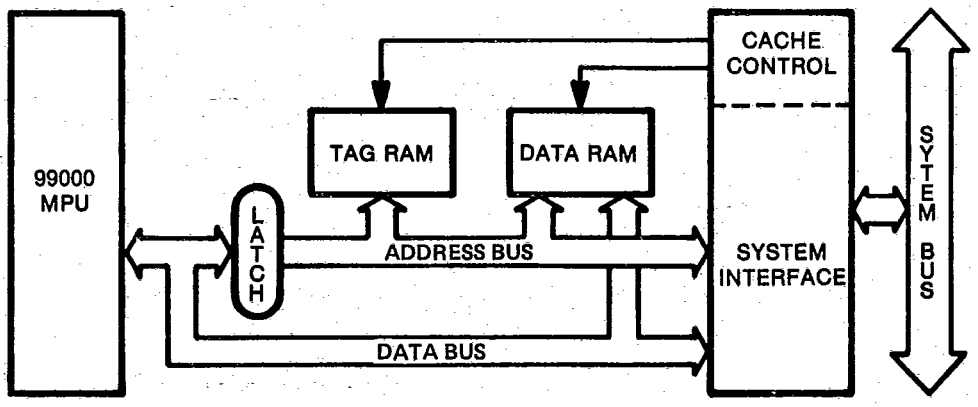


FIGURE 15 – CACHE MEMORY SYSTEM

TMS 99000 PERFORMANCE

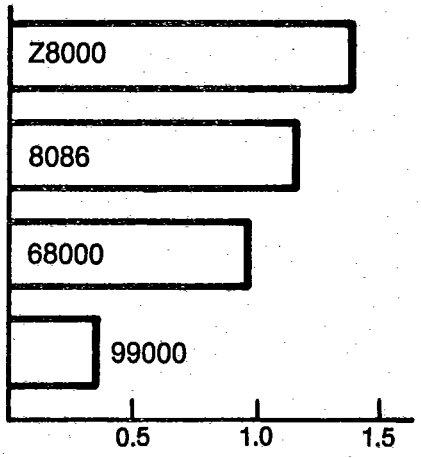
The performance of a microprocessor is a function of the memory cycle time and the memory cycle efficiency. Memory cycle efficiency is a result of accomplishing a given function in the least number of memory accesses. This can be thought of as the information content of each memory access. Because memory cycle times dominate the instruction execution time, the processor that performs a function in the least number of machine states will invariably have the performance advantage. This is especially true when memory cycle times are equal. Thus, as technology improves all microprocessors will lean toward an ultimate in memory cycle times, (in the sub 100 nanosecond range). The processor designed to reduce the number of memory accesses, and thus the number of machine states, will show superior performance. Table 1 contains a comparative listing of common instructions and the number of states required to execute the instruction. The TMS 99000 family of microprocessors is shown as well as state counts and instruction times for the Motorola 68000-8, the Intel 8086-8, and the Zilog Z8000. The results outlined in boxes exhibited the fastest execution time. Note that the TMS 99000 had the highest execution speed for six of the eight instruction statements. A closer examination shows that the TMS 99000 requires fewer machine states for those same instructions.

As a result of reduced state counts and fast machine state time the TMS 99000 exhibits superior performance when compared with competitive processors. The benchmark chosen was found in the Intel Benchmark Report #AFN01551A. The result of including the TMS 99000 in these benchmarks is shown in Figure 16. For all cases the TMS 99000 family processor exhibited superior performance for both zero-and one-wait states. The one-wait state times are given for the TMS 99000 to allow comparison on an equal memory cycle time basis with the MC68000.

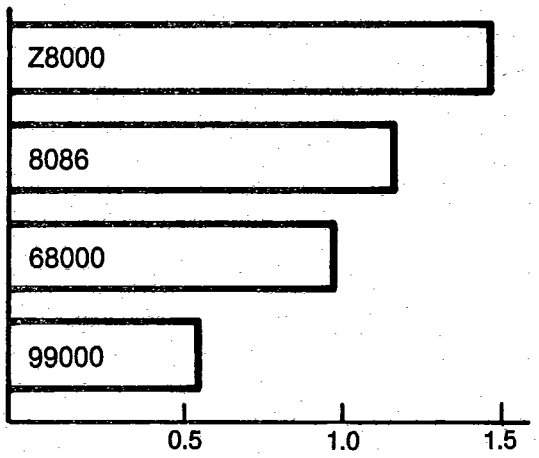
TABLE 1 — COMPARATIVE INSTRUCTION EXECUTION

PROCESSOR	99000 STATE TIME = 166		68000-8 STATE TIME = 250		8086-8 STATE TIME = 125		Z-8001 STATE TIME = 250	
	STATES	REAL TIME	STATES	REAL TIME	STATES	REAL TIME	STATES	REAL TIME
MOV (R, R)	3	(500)	2	(500)	2	(250)	3	(750)
MOV (SYM, SYM)	5	(833)	10	(2500)	29	(3625)	20	(5000)
MOV (*R+, *R+)	9	(1500)	6	(1500)	18	(2250)	20	(5000)
ADD (R, R)	4	(667)	2	(500)	3	(375)	4	(1000)
ADD (SYM, R)	5	(833)	6	(1500)	15	(1875)	9	(2250)
JUMP (REL)	3	(500)	5	(1250)	16	(2000)	6	(1500)
MPYS (R, R)	25	(4167)	35	(8750)	128	(16000)	70	(17500)
DIVS (R, R)	34	(5667)	79	(19750)	165	(20625)	95	(23750)

* BENCHMARK FROM INTEL (*AFN015328)

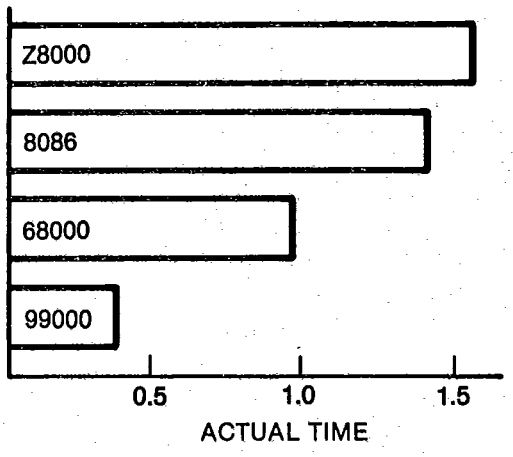


AUTOMATED PARTS INSPECTION(S)

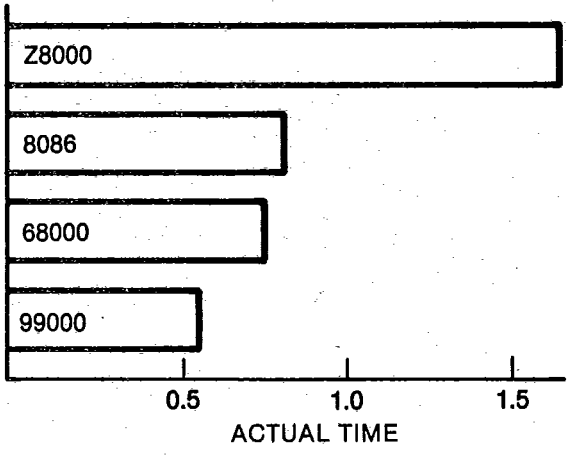


BUBBLE SORT (ms)

* BENCHMARK FROM INTEL (*AFN015328)



XY TRANSFORMATION(S)



BLOCK TRANSLATION (ms)

FIGURE 16 – TMS 99000 BENCHMARKS

THE TMS 99000 FAMILY OF ADVANCED 16-BIT MICROPROCESSORS

In addition to sharing an advanced memory-to-memory architecture, each member of the TMS 99000 family of microprocessors shares the same base instruction set as well as memory, I/O, interrupt and DMA interface. Each TMS 99000 is distinguished by instruction set enhancements, which are implemented through customized programming of the on-chip macrostore emulation memory.

Instruction Set

The instruction set is a superset of the proven TMS 9900 microprocessor and consists of 85 instructions as shown in Table 2. Complete object code compatibility has been maintained with the TMS 9900 providing an existing and proven software base for the TMS 99000 family. Enhancements to the base TMS 9900 instructions set are the addition of double precision arithmetic, double precision shifts, signed integer multiply and divide, memory bit test, and stack instructions supporting recursive program environments. Additionally, the TMS 99110 floating point and the TMS 99120 PASCAL kernal processor contain additional instructions, which are a result of macrostore and microcode programmations. These include the external memory mapper control instructions. The memory mapper control instructions are designed to facilitate control of a TIM99610 memory mapper circuit, allowing memory expansion up to 16M bytes. (See Figure 12)

TABLE 2 – TMS 99000 MICROPROCESSOR FAMILY INSTRUCTION SET

MNEMONIC	DESCRIPTION	MNEMONIC	DESCRIPTION
A	Add Word (16-bit)	LWPI	Load Workspace Pointer Immediate
AB	Add Byte (8-bit)	MOV	Move Word
ABS	Absolute Value	MOVB	Move Byte
AI	Add Immediate	MPY	Multiply
AM	Add Multiple (32-bit)	MPYS	Signed Multiply
ANDI	AND Immediate	NEG	Negate
B	Branch	ORI	OR Immediate
BIND	Branch Indirect	RSET	External Reset
BL	Branch and Link	RTWP	Return to Workspace Pointer
BLWP	Branch and Load Workspace Pointer	S	Subtract (16-bit)
BLSK	Branch Immediate and Push Link to Stack	SB	Subtract Byte (8-bit)
C	Compare Word	SBO	Set Bit to One (I/O)
CB	Compare Byte	SBZ	Set Bit to Zero (I/O)
CI	Compare Immediate	SETO	Set Word to Ones
CKOF	External Clock Off	SLA	Shift Left Arithmetic
CKON	External Clock On	SLAM	Shift Left Arithmetic Multiple (32-bit)
CLR	Clear Word	SM	Subtract Multiple (32-bit)
COC	Compare Ones Corresponding	SRA	Shift Right Arithmetic
CZC	Compare Zeroes Corresponding	SRAM	Shift Right Arithmetic Multiple (32-bit)
DEC	Decrement	SRC	Shift Right with Carry
DECT	Decrement by Two	SRL	Shift Right Logical
DIV	Divide	SOC	Set Ones Corresponding Word
DIVS	Signed Divide	SOCB	Set Ones Corresponding Byte
IDLE	Idle Processor	STCR	Store Communications Register Unit (I/O)
INC	Increment	STST	Store Status
INCT	Increment by Two	STWP	Store Workspace Pointer
INV	Invert	SWPB	Swap Bytes
JXX	Jump (1-unconditional, 12-conditional)	SZC	Set Zeroes Corresponding Word
LDD‡	Long Distance Destination (Memory Mapper Control)	SZCB	Set Zeroes Corresponding Byte
LDCR	Load Communications Register Unit (I/O)	TB	Test Bit (I/O)
LDS‡	Long Distance Source (Memory Mapper Control)	TCMB	Test and Clear Memory Bit
LI	Load Workspace Register Immediate	TMB	Text Memory Bit
LIMI	Load Interrupt Mask Immediate	TSMB	Test and Set Memory Bit
LMF‡	Load Map File (Memory Mapper Control)	X	Execute
LREX	Load External	XOP	Extended Operation (software context switch)
LST	Load Status Register	XOR	Exclusive-OR
LWP	Load Workspace Pointer		

‡ These instructions implemented on TMS 99110 and TMS 99120 only.
NOTE: Floating Point (TMS 99110) and Rx instructions (TMS 99120) are not listed.

Memory Interface

The TMS 99000 memory interface consists of a multiplexed address and data bus, and a control bus for controlling transfers to and from memory. The bus bandwidth of the memory interface accounts for the superior performance of the TMS 99000.

I/O Interface

Both bit, byte, and word input/output transfers may be performed by the TMS 99000 processor. The interface shares many of the same pins as the memory interface, e.g., the address and data bus. Input/output transfers are similar to memory transfers in that the bit, byte, or word address is output on the address bus. Whether a transfer is a bit, byte, or word is determined by the I/O address specified in the instruction. Figure 17 illustrates the partitioning of I/O addresses between bit serial and byte/word parallel transfers. Note that the first 16K addresses are for bit transfers and the last 16K addresses are for byte/word parallel transfers. Byte/word selection is accomplished at instruction coding. The transfer rate is 1.7M bits, bytes, or words per second providing very fast I/O transfers. Wait-states may be inserted to match transfer rate to the capabilities of the I/O device.

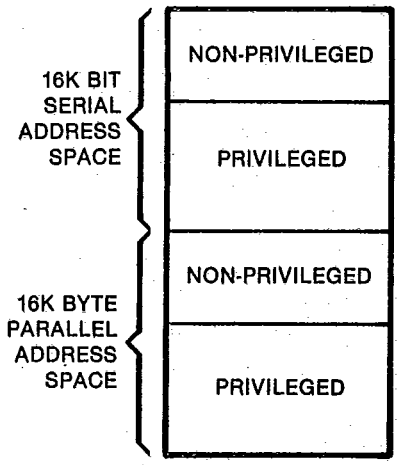


FIGURE 17 – I/O ADDRESS MAP

Interrupt Structure

The TMS 99000 family of microprocessors offer 16 levels of vectored, prioritized interrupts. With advanced memory-to-memory architecture, interrupt context switches are fast and efficient, since a context switch only involves a fetch and store of the workspace pointer, status register, and program counter to redefine the state vector.

Interrupt levels are numbered from level 0 to level 15 with level 0 (RESET) assigned highest priority and level 15 the lowest. Table 3 lists the interrupt levels and their function. Note that level 2 is used for system level interrupts: illegal instruction, privilege violation, and arithmetic fault.

TABLE 3 – INTERRUPT LEVELS

PRIORITY LEVEL	SOURCE AND ASSIGNMENT
LEVEL 0 (HIGHEST PRIORITY)	EXTERNAL : RESET SIGNAL
NMI	NON-MASKABLE INTERRUPT : USER DEFINED
LEVEL 1	EXTERNAL : USER DEFINED
LEVEL 2	INTERNAL : ILLEGAL INSTRUCTION
	INTERNAL : PRIVILEGE VIOLATION
	INTERNAL : ARITHMETIC FAULT
	EXTERNAL : USER DEFINED
LEVEL 3 | LEVEL 15	EXTERNAL : USER DEFINED

The illegal instruction interrupt occurs if an undefined opcode is fetched and decoded and the opcode is not used in an attached processor/computer or macrostore. This interrupt level allows emulation in system memory of instructions which will later be performed by an attached processor/computer or emulated in macrostore. This facilitates transportability of software among system configurations.

The privileged interrupt occurs if execution is attempted of a privileged instruction when the processor is not in the privileged mode. The privileged mode is determined by a bit in the status register. The privileged instructions include input/output instructions and instructions that modify the program environment such as the load status register instruction. Table 4 lists those instructions that are privileged. The privileged interrupt supports user/supervisor environments and implementation of system protection mechanisms.

TABLE 4 – PRIVILEGED INSTRUCTIONS

MNEMONIC	DESCRIPTION
CKON	External Clock On
CKOFF	External Clock Off
IDLE	Processor Idle
LIMI	Load Interrupt Mask Immediate
LREX	Load External
RSET	External Reset
LDCR	Load Communications Register Unit
SBO	Set Bit to One
SBZ	Set Bit to Zero
LMF	Load Map File
LDD	Long Distance Destination
LDS	Long Distance Source

The arithmetic fault interrupt occurs as a result of an arithmetic overflow condition during the execution of arithmetic instructions. This interrupt is enabled/disabled through a bit in the status register. The interrupt context switch occurs immediately following the instruction that caused the overflow without the necessity of using "trap on overflow" type instructions. This provides for a completely automatic interrupt, enhancing execution speed for routines that detect the overflow condition.

SUMMARY

The TMS 99000 microprocessor family provides a true system solution for advanced applications of microprocessors. The TMS 99000 microprocessor family is the only microprocessor family that allows the system designer to design native functionality into his system. It is the only system which supports a path of cost/performance migration with the capability of maintaining system functionality. These options in cost and performance are illustrated in Figure 18. Note that options include both function migration and memory system options to provide the most cost-effective system design. The further integration of functions from memory system software to true silicon solutions has taken the first step with the TMS 99000 microprocessor family. Texas Instruments continues the evolution of functional integration through advanced peripherals and processors to provide leadership in microprocessor products for the '80s.

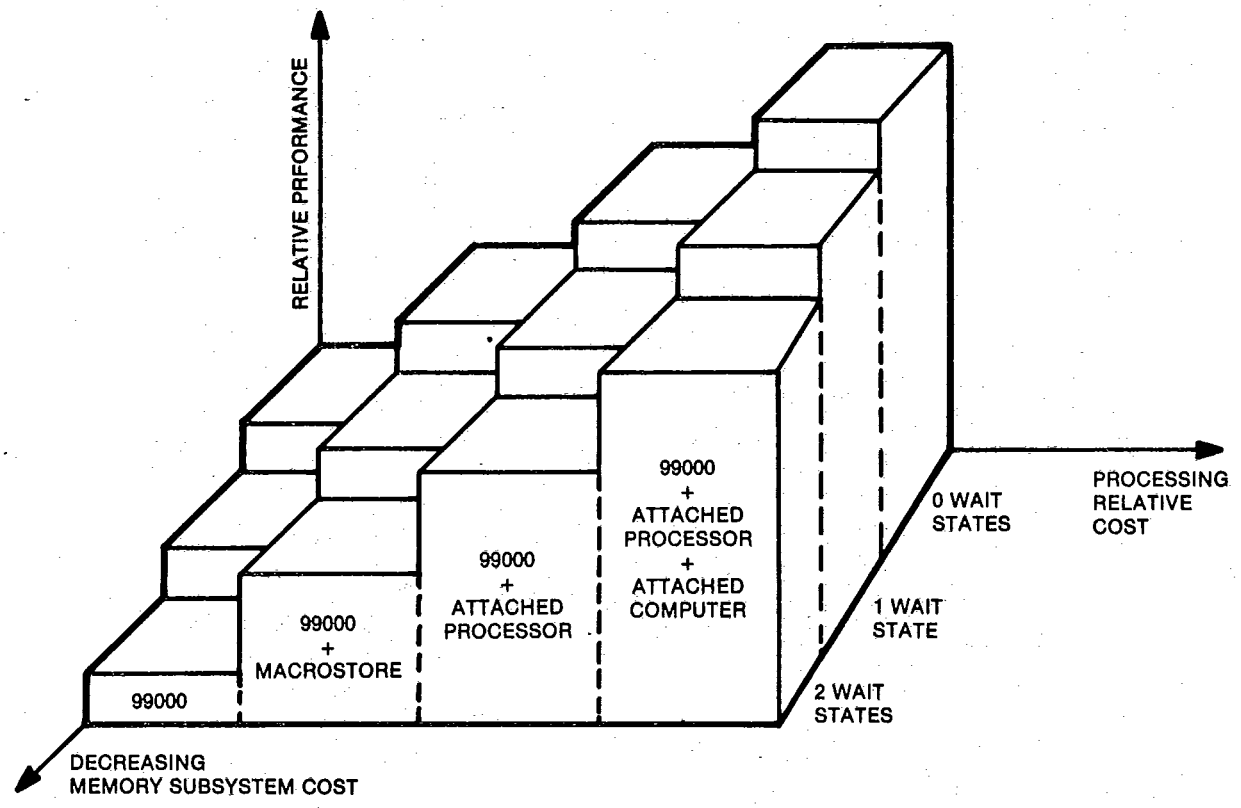


FIGURE 18 – TMS 99000 COST/PERFORMANCE OPTIONS

TI Sales Offices

ALABAMA: Huntsville, 500 Wynn Drive, Suite 514, Huntsville, AL 35805, (205) 837-7530.

ARIZONA: Phoenix, P.O. Box 35160, 8102 N. 23rd Ave., Suite A, Phoenix, AZ 85069, (602) 249-1313.

CALIFORNIA: El Segundo, 831 S. Douglas St., El Segundo, CA 90245, (213) 973-2571; **Irvine,** 17620 Fitch, Irvine, CA 92714, (714) 545-5210; **Sacramento,** 1900 Point West Way, Suite 171, Sacramento, CA 95815, (916) 929-1521; **San Diego,** 4333 View Ridge Ave., Suite B., San Diego, CA 92123, (714) 278-9600; **Sunnyvale,** P.O. Box 9064, 776 Palomar Ave., Sunnyvale, CA 94086, (408) 732-1840; **Woodland Hills,** 21220 Erwin St., Woodland Hills, CA 91364, (213) 704-7759.

COLORADO: Denver, 9725 E. Hampden St., Suite 301, Denver, CO 80231, (303) 695-2800.

CONNECTICUT: Wallingford, 9 Barnes Industrial Park Rd., Barnes Industrial Park, Wallingford, CT 06492, (203) 269-0074.

FLORIDA: Clearwater, 2280 U.S. Hwy. 19 N., Suite 232, Clearwater, FL 33515, (813) 325-1861; **Ft. Lauderdale,** 2765 N.W. 62nd St., Ft. Lauderdale, FL 33909, (305) 973-8502; **Winter Park,** 1850 Lee Rd., Suite 115, Winter Park, FL 32789, (305) 644-3535.

GEORGIA: Atlanta, 3300 Northeast Expy., Building 9, Atlanta, GA 30341, (404) 452-4600.

ILLINOIS: Arlington Heights, 515 W. Algonquin, Arlington Heights, IL 60005, (312) 640-2934.

INDIANA: Ft. Wayne, 2020 Inwood Dr., Ft. Wayne, IN 46805, (219) 424-5174; **Indianapolis,** 2346 S. Lynhurst, Suite J-400, Indianapolis, IN 46241, (317) 248-8555.

MARYLAND: Baltimore 1 Rutherford Pl., 7133 Rutherford Rd., Baltimore, MD 21207, (301) 944-8600.

MASSACHUSETTS: Waltham, 504 Totten Pond Rd., Waltham, MA 02154, (617) 890-7400.

MICHIGAN: Farmington Hills, 3373 W. 12 Mile Rd., Farmington Hills, MI 48018, (313) 553-1500.

MINNESOTA: Edina, 7625 Parklawn, Edina, MN 55435, (612) 830-1600.

MISSOURI: Kansas City, 8080 Ward Pkwy., Kansas City, MO 64114, (816) 523-2500; **St. Louis,** 11861 Westline, Industrial Line Drive, St. Louis, MO 63141, (314) 569-7600.

NEW JERSEY: Clark, 292 Terminal Ave. West, Clark, NJ 07066, (201) 574-9800.

NEW MEXICO: Albuquerque, 5907 Alice NSE, Suite E, Albuquerque, NM 87110, (505) 265-8491.

NEW YORK: East Syracuse, 6700 Old Collamer Rd., East Syracuse, NY 13057, (315) 463-9291; **Endicott,** 112 Nanticoke Ave., P.O. Box 618, Endicott, NY 13760, (607) 754-3900; **Melville,** 1 Huntington Quadrangle, Suite 3C10, P.O. Box 2936, Melville, NY 11747, (516) 454-6600, **Poughkeepsie,** 201 South Ave., Poughkeepsie, NY 12601, (914) 473-2900; **Rochester,** 1210 Jefferson Rd., Rochester, NY 14623, (716) 424-5400.

NORTH CAROLINA: Charlotte, 8 Woodlawn Green, Woodlawn Rd., Charlotte, NC 28210, (704) 527-0930.

OHIO: Beachwood, 23408 Commerce Park Rd., Beachwood, OH 44122, (216) 464-6100, **Dayton,** Kingsley Bldg., 4124 Linden Ave., Dayton, OH 45432, (513) 258-3877.

OKLAHOMA: Tulsa, 3105 E. Skelly Dr., Suite 512, Tulsa, OK 74105, (918) 749-9547.

OREGON: Beaverton, 6700 SW 105th St., Suite 110, Beaverton, OR 97005, (503) 643-6758.

PENNSYLVANIA: Ft. Washington, 575 Virginia Dr., Ft. Washington, PA 19034, (215) 643-6450.

TENNESSEE: Johnson City, P.O. Drawer 1255, Erwin Hwy., Johnson City, TN 37601, (615) 461-2129.

TEXAS: Austin, 12501 Research Bldg., P.O. Box 2909, Austin, TX 78723, (512) 250-7655; **Dallas,** P.O. Box 225012, Dallas, TX 75265, (214) 995-6531; **Houston,** 9000 Southwest Frwy., Suite 400, Houston, TX 77036, (713) 778-6592.

UTAH: Salt Lake City, 3672 West 2100 South, Salt Lake City UT 84120, (801) 973-6310.

VIRGINIA: Fairfax, 3001 Prosperity, Fairfax, VA 22031, (703) 849-1400; **Midlothian,** 13711 Sutter's Mill Circle, Midlothian, VA 23113, (804) 744-1007.

WASHINGTON: Redmond, 2723 152nd Ave., N.E. Bldg 6, (206) 881-3080, Redmond, WA 98052.

CANADA: Ottawa, 436 McClaren St., Ottawa, Canada, K2P0M8, (613) 233-1177; **Richmond Hill,** 280 Centre St. E., Richmond Hill L4C1B1, Ontario, Canada, (416) 884-9181; **St. Laurent,** Ville St. Laurent Quebec, 9460 Trans Canada Hwy., St. Laurent, Quebec, Canada H4S1R7, (514) 334-3635. L

TI Distributors

ALABAMA: Huntsville, Hall-Mark (205) 837-8700.

ARIZONA: Phoenix, Kierulff (602) 243-4101; R.V. Weatherford (602) 272-7144; **Tempe,** Marshall (602) 968-6181; **Tucson,** Kierulff (602) 624-9986.

CALIFORNIA: Anaheim, R.V. Weatherford (714) 634-9600; **Buena Park,** RPS (213) 744-0355; **Canoga Park,** Marshall (213) 999-5001; **Chatsworth,** JACO (213) 998-2200; **El Monte,** Marshall (213) 686-0141; **Glendale,** R.V. Weatherford (213) 849-3451; **Goleta,** RPS (805) 964-6823; **Irvine,** JACO (714) 540-5600; Marshall (714) 556-6400; **Los Angeles,** Kierulff (213) 725-0325; **Palo Alto,** Kierulff (415) 968-6292; **Pomona,** R.V. Weatherford (714) 623-1261; **Sacramento,** Kierulff (916) 924-8522; **San Diego,** Arrow (714) 565-4800; Kierulff (714) 278-2112; Marshall (714) 578-9600; R.V. Weatherford (714) 695-1700; **Santa Barbara,** R.V. Weatherford (805) 465-8551; **Santa Clara,** United Components (408) 496-6900; **Sunnyvale,** Arrow (408) 745-6600; Marshall (408) 732-1100; Time (408) 734-9888; United Components (408) 496-6900; **Torrance,** Time (213) 320-0880; **Tustin,** Kierulff (714) 731-5711.

COLORADO: Denver, Arrow (303) 758-2100; Diplomat (303) 740-8300; Kierulff (303) 371-6500; **Englewood,** R.V. Weatherford (303) 770-9762.

CONNECTICUT: Orange, Milgray (203) 795-0714; **Wallingford,** Arrow (203) 265-7741; Marshall (203) 265-3822.

FLORIDA: Clearwater, Diplomat (813) 443-4514; **Ft. Lauderdale,** Arrow (305) 973-8502; Diplomat (305) 971-7160; Hall-Mark (305) 971-9280; **Orlando,** Hall-Mark (305) 855-4020; **Palm Bay,** Arrow (305) 725-1480; Diplomat (305) 725-4520; **St. Petersburg,** Kierulff (813) 576-1966; **Winter Park,** Milgray (305) 647-5747.

GEORGIA: Norcross, Arrow (404) 449-8252; Hall-Mark (404) 447-8000; Marshall (404) 923-5750.

ILLINOIS: Bensonville, Diplomat (312) 595-1000; Hall-Mark (312) 860-3800; **Elk Grove Village,** Kierulff (312) 640-0200; **Chicago,** Newark (312) 638-4411; **Schaumburg,** Arrow (312) 893-9420.

INDIANA: Ft. Wayne, Graham (219) 423-3422; **Indianapolis,** Graham (317) 634-8202; Arrow (317) 243-9353.

IOWA: Cedar Rapids, Deeco (319) 365-7551.

KANSAS: Lenexa, Component Specialties (913) 492-3555; **Shawnee Mission,** Hall-Mark (913) 888-4747; **Wichita,** LCOMP (316) 265-9507.

MARYLAND: Baltimore, Arrow (202) 737-1700, (301) 247-5200; Hall-Mark (301) 796-9300; **Columbia,** Diplomat (301) 995-1226; **Rockville,** Milgray (301) 466-6400.

MASSACHUSETTS: Billerica, Kierulff (617) 667-8331; **Burlington,** Marshall (617) 272-8200; **Woburn,** Arrow (617) 933-8130; Time (617) 935-8080.

MICHIGAN: Ann Arbor, Arrow (313) 971-8200; **Oak Park,** Newark (313) 967-0600; **Farmington,** Diplomat (313) 477-3200; **Grand Rapids,** Newark (616) 241-6681.

MINNESOTA: Bloomington, Hall-Mark (612) 854-3223; **Edina,** Arrow (612) 830-1800; Kierulff (612) 835-4388; **Minneapolis,** Diplomat (612) 788-8601.

MISSOURI: Earth City, Hall-Mark (314) 291-5350; **Kansas City,** LCOMP (816) 221-2400; **St. Louis,** Arrow (314) 567-6888; LCOMP (314) 291-6200; Kierulff (314) 569-7630.

NEW HAMPSHIRE: Manchester, Arrow (603) 668-6968.

NEW JERSEY: Camden, General Radio Supply (609) 964-8560; **Cherry Hill,** Hall-Mark (609) 424-0880; **Clifton,** JACO (201) 989-7081; Marshall (201) 340-1900; **Fairfield,** Kierulff (201) 575-6750; **Marlton,** Milgray (609) 424-1300; **Moorestown,** Arrow (609) 235-1900; **Saddlebrook,** Arrow (201) 797-5800; **Totowa,** Diplomat (201) 785-1830.

NEW MEXICO: Albuquerque, Arrow (505) 243-4566; International Electronics (505) 345-8127; United Components (505) 345-9981.

NEW YORK: Endwell, Marshall (607) 754-1570; **Freeport,** Milgray (516) 546-5600, N.J. (800) 645-3986; **Hauppauge,** Arrow (516) 231-1000; JACO (516) 273-5500; **Liverpool,** Arrow (315) 652-1000; Diplomat (315) 652-5000; **Melville,** Diplomat (516) 454-6400; **Rochester,** Arrow (716) 275-0300; Rochester Radio Supply (716) 454-7800; Marshall (716) 235-7620; **Ronkonkoma,** QPL (516) 467-1200.

NORTH CAROLINA: Greensboro, Kierulff (919) 852-6261; **Raleigh,** Arrow (919) 876-3132; Hall-Mark (919) 832-4465; **Winston-Salem,** Arrow (919) 725-8711.

OHIO: Beachwood, Kierulff (216) 464-6100; **Centerville,** Arrow (513) 435-5563; **Cincinnati,** Graham (513) 732-1661; **Columbus,** Hall-Mark (614) 846-1882; **Dayton,** ESCO (513) 226-1133; Marshall (513) 236-8088; **Highland Heights,** Hall-Mark (216) 473-2907; **Solon,** Arrow (216) 248-3990.

OKLAHOMA: Tulsa, Component Specialties (918) 664-2820; Hall-Mark (918) 835-8458; Kierulff (918) 749-9543.

OREGON: Beaverton, Almac/Stroum (503) 641-9070; **Portland,** Kierulff (503) 641-9150.

PENNSYLVANIA: Pittsburgh, Arrow (412) 856-7000.

TEXAS: Austin, Component Specialties (512) 837-8922; Hall-Mark (512) 837-2814; Harrison Equipment (512) 458-3555; Kierulff (512) 250-7037; **Dallas,** Component Specialties (214) 357-6511; Hall-Mark (214) 341-1147; International Electronics (214) 233-9323; Kierulff (214) 343-2400; **El Paso,** International Electronics (915) 778-9761; **Houston,** Component Specialties (713) 771-7237; Hall-Mark (713) 781-6100; Harrison Equipment (713) 652-4700; Kierulff (713) 778-6530.

UTAH: Salt Lake City, Diplomat (801) 486-4134; Kierulff (801) 973-6913.

WASHINGTON: Redmond, United Components (206) 643-7444; **Seattle,** Almac/Stroum (206) 763-2300; Kierulff (206) 575-4420; **Tukwila,** Arrow (206) 575-0907.

WISCONSIN: Oak Creek, Arrow (414) 764-6600; Hall-Mark (414) 761-3000; **Waukesha,** Kierulff (414) 784-8160.

CANADA: Calgary, Cam Gard Supply (403) 287-0520; Future (403) 259-6408; Varah (403) 230-1235; **Downsview,** CESCO (416) 661-0220; **Edmonton,** Cam Gard Supply (403) 426-1805; **Halifax,** Cam Gard Supply (902) 454-8581; **Hamilton,** Varah (416) 561-9311; **Kamloops,** Cam Gard Supply (604) 372-3338; **Moncton,** Cam Gard Supply (506) 855-2200; **Montreal,** CESCO (514) 735-5511; Future (514) 731-7441; **Ottawa,** CESCO (613) 729-5118; Future (613) 820-8313; **Quebec City,** CESCO (418) 687-4231; **Regina,** Cam Gard Supply (306) 525-1317; **Saskatoon,** Cam Gard Supply (306) 652-6424; **Toronto,** Future (416) 663-5563; **Vancouver,** Cam Gard Supply (604) 291-1441; Future (604) 438-5545; Varah (604) 873-3211; **Winnipeg,** Cam Gard Supply (204) 786-8481; Varah (204) 633-6190. AD

Texas Instruments invented the integrated circuit, microprocessor and microcomputer. Being first is our tradition.

TEXAS INSTRUMENTS
INCORPORATED

Printed in U.S.A.

13