DECSYSTEM-10

DUAL-PROCESSOR SYSTEMS

TABLE OF CONTENTS

DECSYSTEM-10 DUAL-PROCESSOR SYSTEMS

I. ANNOUNCEMENT

Digital Equipment Corporation announces the development and support of a high throughput, dual-processor, DECsystem-10 system. The highly advanced and field-proven DECsystem-10 operating system now supports dual processor configurations.

Technical Leadership

In keeping with its technical leadership in large computer operating systems, DECsystem-10 Dual Processor software provides maximum performance with minimum overhead. For those upgrading single processor systems to dual-processor capability, the Dual Processor software requires no user inconvenience or incompatibilities. Operator re-training is minimal.

Hardware required to accomplish expansion consists of a KA10 processor and a memory port for each memory. No bus switches, doorbells, DECtapes, or other equipment is required. More memory may be desirable in order to provide core space for the additional jobs that the Dual Processor system will handle.

The DECsystem-10 Dual Processor monitor is not significantly larger - approximately 1K word monitor expansion will be required in the average installation.

II. SYSTEM PERFORMANCE

More Processor Power

The DECsystem-10 Dual Processor software offers a significant increase in throughput for compute bound systems. Increased processing power has been measured at over 1.9 times a single processor system in a highly compute bound job mix.

A compute bound work load might typically be found in Computation Centers and in various laboratory applications both with and without on-line, real-time data handling requirements. Also, commercial firms offering timesharing services may be recording extended periods of low null time and thus find themselves operating in a compute bound mode.

III. CONFIGURATIONS

Easy Expansion

Configuring a DECsystem-10 Dual Processor system requires only the addition of a second KA10 processor plus implementation of a memory port for each core memory module. No other hardware is required!

Cost Effectiveness

In order to take full advantage of the efficiency of the software and of the performance of the second processor, sufficient core memory must be provided. An adequate amount of core quarantees two or more runnable jobs will reside in core simultaneously.

Of course, it is not always possible to provide the ideal configuration for a variety of reasons which are sometimes technical but most often economic. Nevertheless, one can make some reasonable estimates concerning the performance to be expected when the amount of core is limiting and/or the mix consists of I/O bound jobs.

Performance Goals

The graph to the right has as the horizontal axis the loading that would occur on a single processor with a typical job mix. For those considering expansion of a single processor system, this parameter (null time) can be measured and is known.

The exponential curves indicate forecasted performance increase for several different numbers of runnable jobs which can simultaneously fit in core.

The "average" job size and the amount of core available determine this number. Four examples are provided in this booklet.

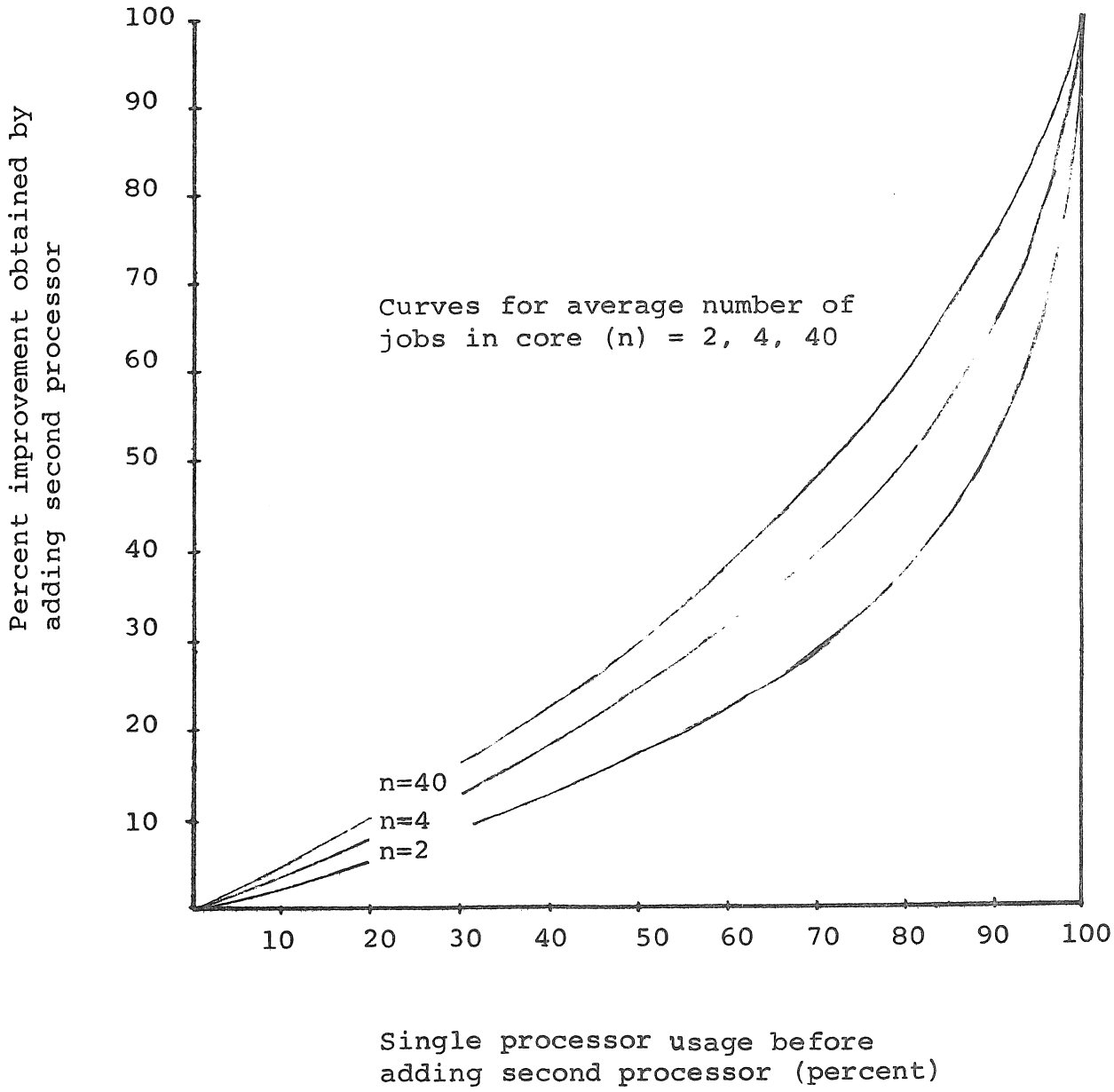
IV. NEW SYSTEM CONTROL

New Console Commands

New console commands and monitor calls let the user assign a particular job to a particular processor or to both processors.

THROUGHPUT IMPROVEMENT TO BE EXPECTED

BY ADDING A SECOND PROCESSOR



Using these control functions, a DECsystem-10 Dual Processor system can be tailored to favor one or more jobs as required. For example, a highly compute bound job could be assigned to the Slave processor and could proceed virtually without interruption, thereby permitting assignment of maximum computing power to that job.

V. REAL-TIME PROCESSING

Compatibility

Real-time processing can take place exactly as in the single processor system; that is, jobs may be locked in core and run under control of the primary processor's interrupt system. Linkage of the user's program to the interrupt system is accomplished using the RTTRP (Real-Time Trap) monitor call.

Faster Response

With a DECsystem-10 Dual Processor system, users have the further advantage that real-time hardware may be interfaced to the secondary processor and controlled using the monitor call TRPSET (Trapset). TRPSET allows the user to assign a hardware interrupt level to his real-time device and then link this level to his code. Even the peripherals provided as standard equipment with the secondary processor can be utilized with TRPSET. System response to the needs of the real-time application is maximized.

VI. RELIABILITY

Processor

The DECsystem-10 Dual Processor system requires no extra hardware beyond the second processor and memory ports. Should the secondary processor fail, the system remains operable as a single processor system without reloading. Should the primary processor fail, it is possible to move the I/O cables to the secondary, thereby establishing a working single processor system. The likelihood of either or both of the processors failing is small.

Memory

If other system units, such as a memory module, should fail

(a Dual Processor system neither increases or decreases this possibility significantly), switches provided as standard equipment can be set to remove the defective module from the bus. The system can then be restarted.

I/O Devices

If I/O devices become faulty, the failure may still not be critical. Magtapes, DECtapes, and user consoles are examples of devices which will not cause the system to be significantly less useful should a failure occur.

There are, however, cases where a failure results in the system going down and one is confronted with the various inconveniences, loss of data, loss of money, or other conditions.

Many system planners believe the most practical remedy is to fix the problem and proceed. Proper preventive maintenance and quick solution by service personnel to a problem can minimize the frequency of occurrence of failures and the severeness of their effect.

Bus Switches

In order to provide utmost reliability, DEC can provide both manual and programmable I/O bus, memory bus, and channel bus switch hardware. If switch hardware is provided, it is assumed by the Dual Processor software that the switches are thrown to their correct positions.

If programmable switches are supplied, the operating system assumes that the CLEAR or RESET position is the proper state for the switch hardware.

Using the DEC supplied software and manual switches, several different monitors can be built and kept in reserve so that if the need arises, the system can be reconfigured, a new monitor loaded, and operation resumed. At certain times users may wish to operate two single processor systems. Thus, the switches become useful for more than solving problems caused by component failure.

VII. COMPATIBILITY

Transparent to User

The DECsystem-10 Dual Processor software is an extension of

the standard DECsystem-10 operating system with dual processor code a conditional assembly feature. To the user, the Dual Processor system is virtually identical to the single processor system. Of course, the added real-time capability and the new console commands and monitor calls, allowing the user to specify the processor(s) on which a job is to run, are extensions but not incompatibilities.

Minimal Operator Training

Additional operator training is minimized for the DECsystem-10 Dual Processor systems. Simplified start/stop procedures are the same as single processor systems. Messages to the operator of relevance to the secondary processor appear on the primary processor's console teletype. Should the operator halt the primary processor, the secondary processor detects that the primary processor is no longer running and discontinues operation until the primary processor is restarted. Once the primary processor is restarted, the secondary processor automatically continues its operation. Thus, the secondary processor needs no operator attention during a system reload.

No Special Hardware Required

No special hardware is needed. Customers may order an interrupt box called a "doorbell" if they wish. Studies will be made in the future to show whether or not any significant improvements would be realized by this technique. The code necessary to support such a device would be minimal. In fact, one customer with an on-line application has already designed and built the doorbell and written the software.

VIII. CONFIGURATIONS SUPPORTED

Dual KA10 Systems - Fully Supported

Dual Processor KA10 systems have been installed at our factory to ensure that full support is available to KA10 customers. The 5.04 and later releases of the DECsystem-10 monitor support two processor systems.

Dual KI10 Systems - Fully Supported

Dual Processor KI10 systems can be supplied and receive full support with system software.

KA10/PDP-6 Systems - Coded but Unsupported

Support of KA10/PDP-6 systems is to be provided by customer consultation with PDP-10 software support personnel. Coding is being provided specifically for the PDP-6, although this coding will remain untested and unsupported.

On the positive side it should be noted that the first Dual Processor system to run with DEC designed software was and is a KA10/PDP-6 system! This system runs a small monitor known as 10/40 4-series. This notation indicating that the system does not include a disk and the monitor version is not our current software.

Those customers considering a dual PDP-10/PDP-6 system can look forward to achieving success. However, DEC software support personnel should be consulted.

IX. DOCUMENTATION

The detailed description of DECsystem-10 Dual Processor software will be published in future updates and editions of the DECsystem-10 manuals. These manuals are:

1. Introduction to DECsystem-10 Software
2. DECsystem-10 Assembly Language Handbook
3. DECsystem-10 Software Notebooks
4. Future editions of DECsystem-10 Users Handbook
5. Multi-Processing Functional Specification Software Notebook

X. THEORY OF OPERATION

DECsystem-10 Dual Processor systems are composed of two CPU's, designated the primary processor and the secondary processor. The primary processor is connected to all of the memory in the system and has all of the system's peripheral I/O equipment connected to its I/O bus. The secondary processor also has access to all of memory; however, there are normally no I/O devices on this processor's I/O bus, although certain devices may be attached in real-time applications. The primary processor performs exactly the same operations as the processor in a single processor system. This includes all I/O operations, swapping, core allocation, resource allocation, and command decoding. The secondary processor also performs

scheduling and execution of user jobs according to the same algorithm used in the single processor system.

The secondary processor executes user jobs and scans the same job queues as the primary processor. However, since the secondary processor cannot do any standard I/O, it looks for any compute bound jobs which are in core and runnable. An interlock has been added to the scheduler to prevent the possibility of both processors trying to execute the same job at the same time. Whenever a job being executed by the secondary processor requests an I/O operation to be performed, the job is stopped and marked for execution by the primary processor only. Thus, the primary and secondary processors run completely asynchronously, both executing the same scheduler, doing the same job accounting, and using the same job queues.

XI. PERFORMANCE ANALYSIS

The following examples provide measured performance improvement for a 1055 system as installed in our factory. The hardware configuration is:

2 - KA10 Central Processors
1 - Swapping Drum
2 or more Disk Pack Drives
1 - Line Printer
Compatible Magnetic Tape - 60KC
1 - Card Reader
Core Memory - 64-128K (examples vary in their core requirements)

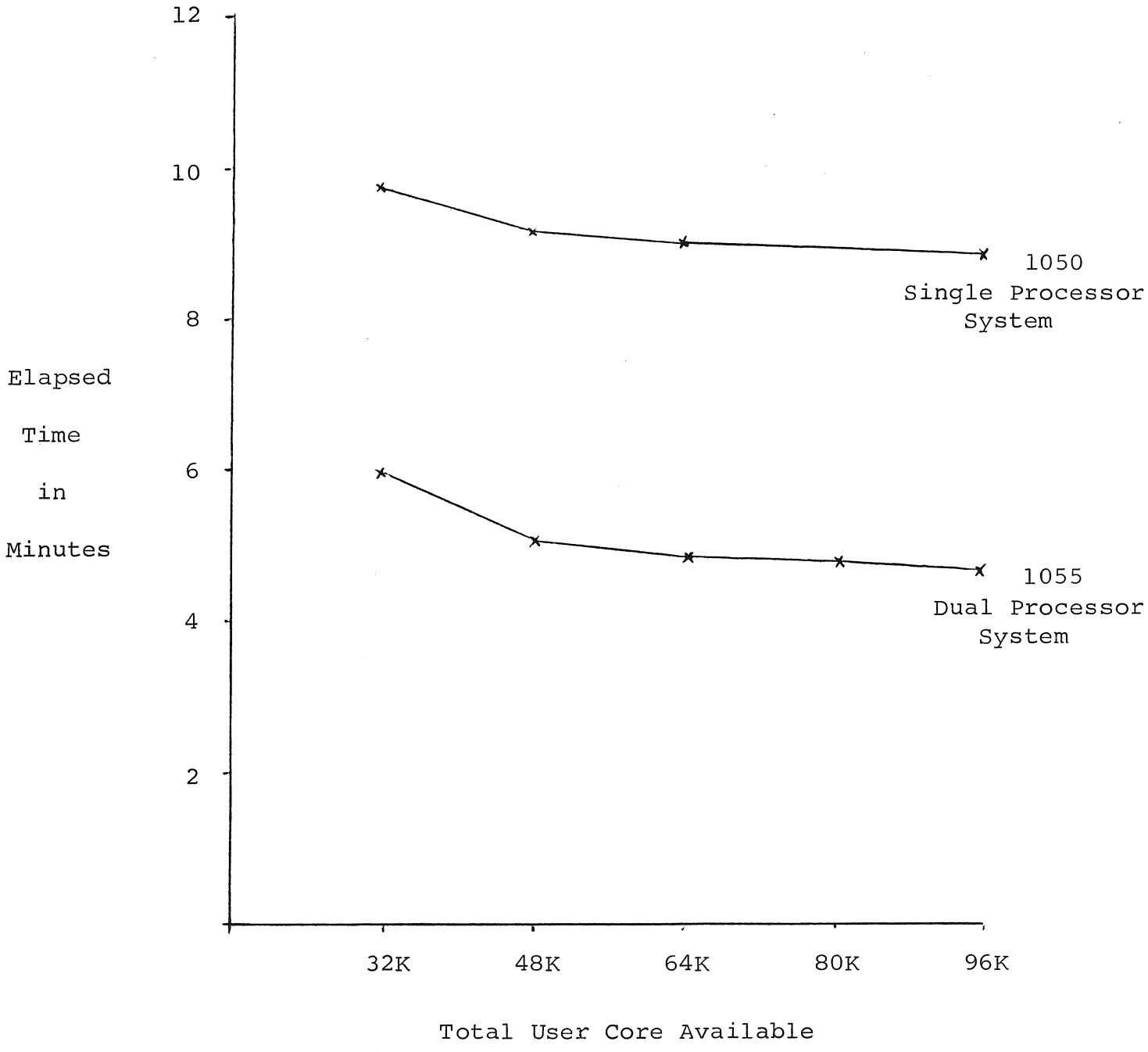
Example 1 - Benchmark A

Consists of eight FORTRAN jobs executed under MPB.

Core Required per Job	
8K words	Jobs 1 and 2 - Two jobs which iteratively compute solutions to differential equations. Rich in sine and cosine computations.
9K words	Jobs 3 and 4 - Two jobs heavy in floating point calculations.

Benchmark A

Throughput improvement at 96K was 1.9:1



Core Required per Job	
11K words	Jobs 5 and 6 - Two jobs which perform effective nuclear charge calculations.
12K words	Jobs 7 and 8 - Two jobs which perform fast Fourier transforms.

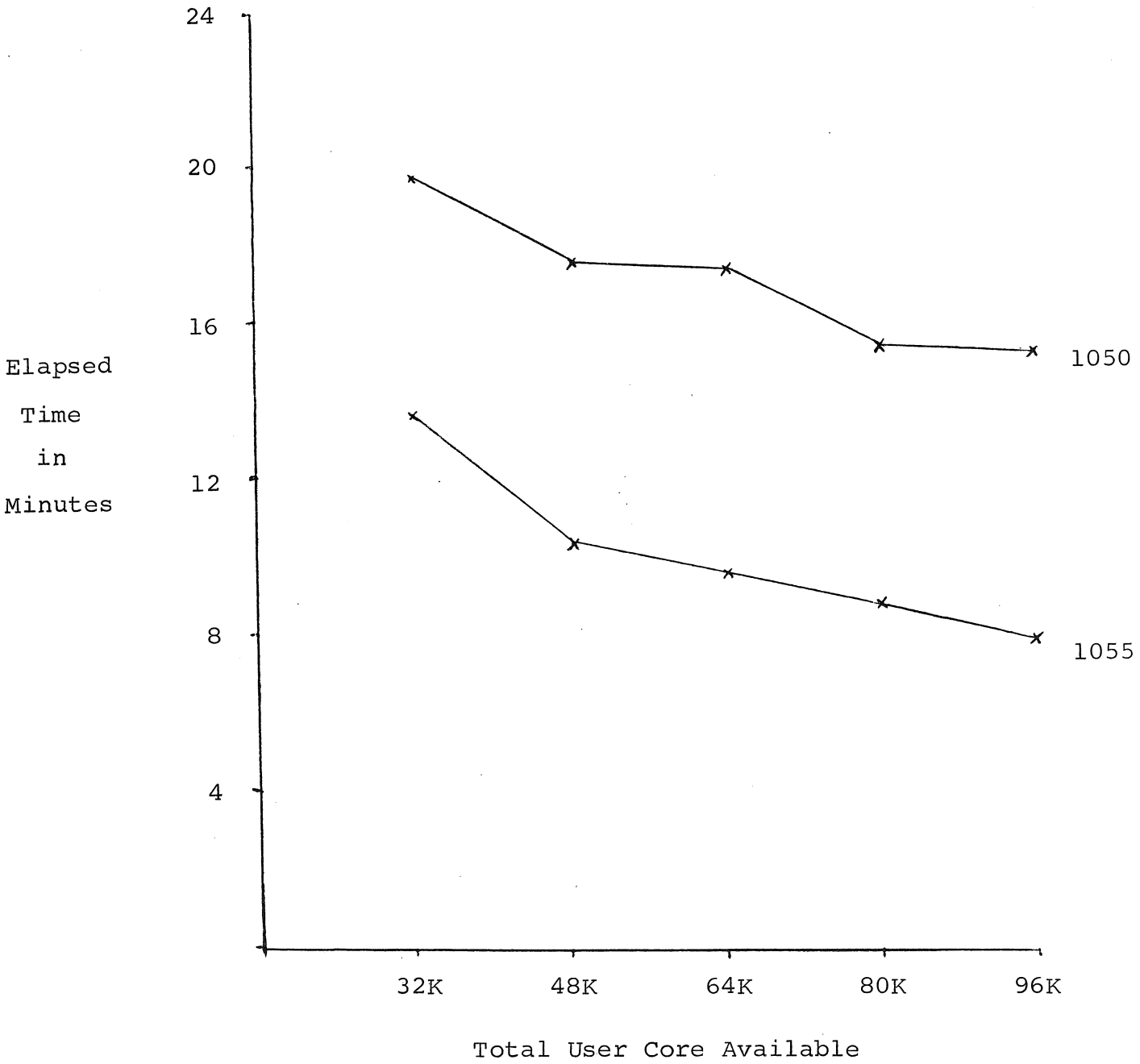
Example 2 - Benchmark B

Consists of twenty-six jobs executed under MPB.

10K words	Jobs 1,2, and 3 - Test speed of single precision vs double precision arithmetic.
7K words	Jobs 4,5, and 6 - Calculates the chi square of data elements.
7K words	Jobs 7,8, and 9 - Calculates a table of differences for the data elements supplied.
13K words	Jobs 10,11, and 12 - Matrix inversion problem.
8K words	Jobs 13,14, and 15 - Real primary numbers raised to double precision power and real primary numbers raised to real primary numbers.
7K words	Jobs 16,17, and 18 - Test real numbers being operated on by complex numbers.
8K words	Jobs 19,20, and 21 - Complex primary raised to an integer primary power.
7K words	Job 22 - Tests the evaluation of subscript expressions.
7K words	Job 23 - Calculates the square root input data using a successive approximation approach.
7K words	Jobs 24,25, and 26 - Tests the accuracy of real numbers being operated on by double precision numbers using the multiplication, division, addition, and subtraction operators.

Benchmark B

Throughput improvement at 96K was 1.88:1



Example 3 - University of Pittsburgh

This test compares the time required to run one copy of the program on a 1050 system versus the time required to run two copies of the same program on a 1055 system (1050-single processor, 1055-dual processor). The program is a block diagonal least squares analysis requiring 27K of core. During execution it makes approximately 300 disk accesses per minute. The elapsed execution time of a single copy of the program running on the 1050 system was 28:36.72 minutes. The elapsed execution time for two copies of the program running on the 1055 system was 30:01.57 minutes. This represents an increase in throughput of 189%.

Example 4 - Department of Defense

This test compares the time required to run one copy of a program on a 1050 system with the time required to run two copies of the program on a 1055 system. Three different tests were performed.

Core Required per Program	
7K words	Test No. 1 - an I/O bound FORTRAN program. Throughput improvement ratio 1.73:1
9K words	Test No. 2 - computes all magic squares. Throughput improvement ratio 1.90:1
24K words	Test No. 3 - Matrix manipulation using magtape. Throughput improvement ratio 1.89:1