DOCUMENT TYPE: 1988 PROCEEDINGS

TITLE: PARTITIONING LARGE PROCESSORS

AUTHOR: David J. Young
Richard L. Poliak
Russell W. Martindale

AFFILIATION: Amdahl Corporation

As the size, power, and cost of mainframe computers increases, so does the importance and complexity of efficiently using them. Using one mainframe to process competing workloads, such as IMS and TSO significantly increases the complexity of the tuning effort. This effort can be simplified by separating the workloads, either by placing them on different processors, or by partitioning the processor and dedicating a partition to each workload. IBM's PR/SM is one method which can be used to partition a processor. Another method is Amdahl's 580/Multiple Domain Feature. Both methods allow multiple partitions or domains to be executed concurrently in a single processing complex, by allocating appropriate amounts of processor(s), channels, and main storage to the various domains or partitions.

A series of experiments was conducted in an IMS/VS-TSO environment using Amdahl's 580/Multiple Domain Feature (580/MDF) to determine its effects on I/O processing. This paper will document 580/MDF's effects on I/O activity by describing the various RMF metrics used to measure I/O activity, relating the RMF metrics to 580/MDF scheduling concepts, and analyzing the experiments and their results.

PARTITIONING LARGE PROCESSORS

David J. Young
Richard L. Poliak
Russell W. Martindale

Amdahl Corporation

ABSTRACT

As the size, power, and cost of mainframe computers increases, so does the importance and complexity of efficiently using them. Using one mainframe to process competing workloads, such as IMS and TSO significantly increases the complexity of the tuning effort. This effort can be simplified by separating the workloads, either by placing them on different processors, or by partitioning the processor and dedicating a partition to each workload. IBM's PR/SM is one method which can be used to partition a processor. Another method is Amdahl's 580/Multiple Domain Feature. Both methods allow multiple partitions or domains to be executed concurrently in a single processing complex, by allocating appropriate amounts of processor(s), channels, and main storage to the various domains or partitions.

A series of experiments was conducted in an IMS/VS-TSO environment using Amdahl's 580/Multiple Domain Feature (580/MDF) to determine its effects on I/O processing. This paper will document 580/MDF's effects on I/O activity by describing the various RMF metrics used to measure I/O activity, relating the RMF metrics to 580/MDF scheduling concepts, and analyzing the experiments and their results.

1.0 580/MDF DESIGN CHARACTERISTICS

580/MDF allows domains to access hardware based on fixed time-slice allocations. We can make two observations about active domains in a 580/MDF environment.

First, 580/MDF will not preempt a domain entering a wait state during its dispatch (time slice). That is to say, 580/MDF will allow a domain to complete its time slice without the intermediate dispatch of another domain. The second observation involves the handling of I/O interrupts. I/O operations process asynchronous-to-CPU operations. Since the duration of a domain time slice is finite, some I/O requests complete while the domain owning the requests is not currently dispatched. The processing of the I/O interrupts that signal completion of the I/O requests remains pending until the domain is again dispatched for its time slice. Once again, 580/MDF does not preempt the currently dispatched domain to handle the completion of the I/O belonging to another domain.

The impact of these 580/MDF strategies is reduced domain dispatching, which limits the CPU overhead that accompanies domain switching. A consequence, however, is a possible prolonging of I/O response time caused by delays associated with I/O interrupt processing.

The scheduling parameter controls the frequency of 580/MDF domain dispatching. Simply stated, as the value of the scheduling parameter decreases, the frequency of domain dispatching decreases; the result is longer I/O service delays and less CPU overhead because of 580/MDF domain switching. As the value of the scheduling parameter increases, the frequency of domain dispatching increases; the result is shorter I/O service delays and more CPU overhead for domain switching.

2.0 RMF DATA COLLECTION

The RMF Monitor I routine collects data used by the RMF postprocessor (ERBRMFPP) for reporting. Data is collected by sampling system data areas at regular intervals and then passing this information to SMF for permanent recording. Both the sampling and recording interval are user specified in the ERBRMFxx member of SYS1.PARMLIB. This member is used during Monitor I initialization. The calculated times for the cycle (sample) "timer pop" and the write to SMF are derived from time-of-day (TOD) clock values. TOD clock values reflect elapsed time. A TOD clock facility is provided for each domain in a 580/MDF environment. A domain TOD clock measures elapsed time regardless of the domain's status, that is, dispatched or waiting. MDF can affect RMF sampling during data collection. If the "timer pop" based on cycle time occurs while a 580/MDF domain is not in its time slice (not dispatched), the best case is a delay in taking the sample. The worst case is a missed sample. In addition, note that once a domain is dispatched, RMF must compete with other work within the domain for access to resources. If a relatively low dispatching priority is assigned to RMF on a heavily loaded system, the same sampling error or delay occurs. Obviously this problem could occur in an MVS environment without MDF.

We recommend the following:

RECOMMENDATIONS:

* Assign a high dispatching priority for RMF in MVS.

* Do not set cycle time to less than 250 ms.

* Watch for CPU allocations of 75/25, 80/20, and 90/10. Long domain time slices cause domains with short time

slices to possibly incur sampling delays or errors.

* Set the recording interval to less than 60 minutes. This is a general rule of thumb involving postprocessor report generation. You cannot generate a postprocessor interval report for a time period shorter than the RMF recording interval.

* Validate the number of samples in an RMF report interval. Convert the report interval to seconds (1 hour is 3600 seconds) and multiply by the number of cycles (samples) per second. If the RMF number of samples varies by more than 10% of the calculated samples, the measured data . may be inaccurate.

3.0 580/MDF AND I/O ISSUES

The issues involving I/O in a 580/MDF environment are as follows:

* Which I/O measurements does 580/MDF affect?

* Which factors or characteristics that contribute to I/O response time components does 580/MDF affect?

* Can we control the effects of 580/MDF on I/O by adjusting 580/MDF or MVS/SP version 2 parameters?

We can answer the first two issues by defining how I/O response time components are measured and by observing the nature of the factors that can contribute to those values. We will address the third issue by analyzing RMF data from a set of 580/MDF experiments.

4.0 580/MDF AND MVS/SP VERSION 2 I/O MEASUREMENTS

To determine the impact of 580/MDF on I/O performance, one must understand how I/O is measured and reported. This study uses RMF to analyze I/O performance; therefore, the reader should understand the information that RMF reports on the DIRECT ACCESS DEVICE ACTIVITY report. Figure 1 should clarify the relationships among MDF, RMF, and I/O operation events.

AVG IOSQ TIME is the calculated amount of time an I/O request remains queued in the software because IOS detects that the requested device is busy or unavailable. IOS considers a device busy from the time the channel subsystem accepts the I/O request (SSCH RC=0) until the IOS SLIH clears the interrupt signalling I/O completion. MVS/SP version 2 RMF derives this value for us by dividing the average queue length (average number of requests queued at the UCB representing the device) by the device activity rate. The calculation of IOSQ time is based on the sampling of the number of requests queued to a device at each RMF cycle. If sampling is accurate, the value calculated is accurate. As mentioned in Section 2, 580/MDF can affect RMF's ability to take samples. RMF in MVS/370 reports only an average queue length, which includes queuing incurred because a channel path to the device is unavailable (busy) as well as queuing because the device is busy.

AVG PEND TIME is a penalty that an I/O request incurs within the channel subsystem before being processed. Channel subsystem facilities measure this time independent of CPU operations. It begins when the channel subsystem accepts the I/O request and ends when the device handles the first CCW in the channel program. Because pend time occurs in the channel subsystem and not in the MVS software, 580/MDF does not affect the measurement of the value.

AVG DISC TIME is the amount of time a device spends processing an I/O request without requiring service from any other I/O path component (channel, storage director, or head of string). This time usually refers to the mechanical portions of an I/O operation (seek, rotational delay) associated with RPS DASD. Disconnect time includes delays incurred because the device is unable to reconnect to the channel path for data transfer (RPS miss). The channel subsystem facilities measure this time. Because DISC time occurs in the channel subsystem (more specifically, at the device) and not in the MVS software, 580/MDF does not affect the measurement of the value.

AVG CONN TIME for the most part is the amount of time spent in data transfer. The channel subsystem facilities measure this value also. Because data transfer occurs independently of CPU operations (i.e., MVS software), 580/MDF will not affect the measurement of this value.

AVG RESP TIME is the amount of time required to complete an I/O request. It is computed by adding the calculated IOSQ time to the measured values for PEND, DISC, and CONN time. The reported average response time equals the sum of the reported values for IOSQ, PEND, DISC, and CONN time.

5.0 580/MDF AND I/O CHARACTERISTICS

Having described how I/O response time components are measured or calculated, we can determine the factors that contribute to these components and any influence 580/MDF might have on them.

IOSQ time reflects the amount of time a device is not available for servicing new requests. Factors that contribute to this value include the time it takes to service the request at the device, data transfer time, and any time required to process the interrupt signaling completion of the request at the device. An increase in the amount of time spent on any of these activities causes a device to remain busy longer, thus increasing IOSQ time. I/O arrival rates may also influence this value.

PEND time reflects path contention. Contributing factors are high utilization at the channel, storage director, or head of string, as well as cross-system contention represented by reserves at the device.

DISC time depends on seek time and the ability of a device to reconnect to a channel path to complete data transfer. Keeping frequently accessed data on a volume within a relatively small range of cylinders controls seek time. Path utilization and device type influence RPS reconnect (or miss). Latency, the other component of DISC time, statistically is equal to one half a rotation of the requested device.

CONN time is mostly data transfer, although protocol (communication between channel and storage director) and CCW execution time are included. Contributing factors are device speed, blocksize, and access-method considerations such as number and size of buffers.

Components of I/O operations that contribute to PEND, DISC, and CONN time occur independently of CPU processing. That is , they are associated with the channel subsystem or I/O device, not MVS software. Thus, if the same workload were run in a native MVS, single-domain 580/MDF environment, the PEND, DISC, and CONN time values for utilized devices would be the same.

Channel subsystem facilities do not measure and RMF does not report the time required to process an interrupt that signals completion of an I/O request. This time (in native MVS or a single-domain environment) is usually quite small. Factors that contribute to I/O interrupt processing time are total system I/O rate, number of CPUs enabled for I/O interrupt handling (when processing on MP systems), and the amount of dispatchable higher priority work preceding the I/O interrupt requests. MDF's influence on I/O interrupt processing appears in two areas.

First, the delay on a completed but pending I/O is a function of the duration of time slices assigned to each active domain in the 580/MDF environment. Once again refer to Figure 1. Assume a two-domain 580/MDF environment (i.e., this study), with each domain running in MP mode. MP mode indicates the two domains are sharing the processors, versus UP mode or logically partitioned mode, where each domain has a dedicated processor. The processor allocation is controlled through the Configuration Attributes screen, as illustrated in Figure 2. The processing of IMS domain I/O requests that complete during a TSO domain time slice will be delayed by the portion of the TSO time slice not overlapped by the channel subsystem and device processing of the IMS requests (t3-t2), plus the time required to switch domains when the TSO time slice expires. The duration of fixed time slices is a derived value controlled by the scheduler parameter option of 580/MDF and the CPU allocation for each active domain. The scheduler parameter is set through the System Scheduling screen, as illustrated in figure 3.

Second, when the IMS domain is again dispatched, the IOS interrupt handler routines spend some initial portion of the time slice (t5-t4) processing IMS I/O requests that completed during the TSO time slice. Unlike the non- overlap and domain switch delay that is unique to 580/MDF environments, I/O interrupt processing delays occur regardless of the presence of 580/MDF. A native MVS environment would handle the I/O interrupts as they occur with little delay being imposed on any given I/O. 580/MDF, however, may effect the arrival rate of requests to the IOS interrupt handler; thus, the average amount of time to process any given I/O may increase.

Once again assume the two-domain MP mode environment and refer to Figure 1. One of the CPUs in the domain begins processing the interrupts; however, the other CPU is quite capable of issuing new I/O requests. A new request could be for a device that is busy because IOS has not yet cleared a prior I/O request. IOS queues the new request at the device (UCB). This queuing eventually appears in the IOSQ time calculation. The interrupts handled by the other CPU do not arrive in a random fashion (because of the delay before the domain was dispatched) but now appear to be a burst or steady stream of completed I/Os. The System Resource Manager (SRM) controls the processing of "bursts" or heavy rates of I/O interrupts on MP systems. SRM attempts to keep the fewest number of CPUs enabled for I/O interrupt handling to reduce CPU overhead. When the percentage of interrupts processed by the I/O SLIH via TPI increases beyond a specified high threshold, the SRM enables another processor for I/O interrupt processing. When this rate drops below a specified low threshold, the SRM again disables the additional CPU from processing I/O interrupts. The thresholds are implemented via the CPENABLE(LO,HI) parameter in the IEAOPT member of SYS1.PARMLIB. Lowering the HI CPENABLE threshold can reduce the I/O interrupt processing time (t5-t4) in heavy I/O environments but at the cost of some increase in CPU overhead.

The delay in processing I/O interrupts is not measured in MVS/SP version 2; thus, the influence of 580/MDF on this value is not directly measured. However, a relationship exists among the interrupt delay, the frequency of domain dispatching, and the RMF-reported IOSQ times. We conducted an experiment to see if indeed these relationships exist and if we could detect their impact in RMF reports.

6.0 WORKLOAD DESCRIPTIONS

We measured two workloads in this study: IMS and TSO, both running under MVS/SP 2.1.7. For the purposes of this study, we considered IMS the production workload and TSO the test/development workload.

IMS WORKLOAD

Figure 4 depicts the IMS workload.

We executed the IMS workload using IMS/VS version 1, release 3.0, and simulated 1,000 terminal users with the cross-domain simulation feature of TPNS and VTAM/MSNF. One hundred application programs processed transactions submitted to IMS. We defined 100 data bases distributed as shown in Figure 4. Sixty message-processing regions (MPR) were available for transaction processing, and no BMPs were active. SURF/IMS assisted in generating the workload. (SURF/IMS is an Amdahl application prototyping tool).

We simulated an update workload of moderate resource requirements as illustrated in Figure 5.

TSO WORKLOAD

The TSO workload consisted of 450 TSO users, simulated using the cross-domain simulation feature of TPNS and VTAM/MSNF. Figure 6 depicts the application profile.

7.0 EXPERIMENTS and RESULTS

We ran the workloads in two MP domains, each with a fixed 50% of the processor (target allocation = 50%, maximum allocation = 50%). We ran three tests, using scheduling parameter values of 1, 3, and 5. Consequently the 580/MDF time slices for each domain were 30, 20, and 10 milliseconds, respectively. Table 1 shows the AVG IOSQ and device RESPonse times. Figure 7 presents a graph of the data.

The longest IOSQ times are associated with the longest domain time slices, as indicated by a scheduler parameter value of 1. As the length of time slices decreases for both the IMS and TSO domains, IOSQ times for devices in those domains decrease. We decreased the duration of domain time slices by raising the SCHED parameter value. Raising the SCHED parameter value from 1 to 5 reduced the average IOSQ time in the IMS domain from 7.00 to 3.45 milliseconds, and in the TSO domain from 4.75 to 2.66 milliseconds. This effect is magnified as device activity increases. Table 2 lists some examples.

Figure 8 plots I/O per second as the value of the scheduler parameter changes. The number of I/Os per second for the IMS domain rises from 705.3 to 835.5 when the value of the parameter changes from 1 to 5. The major reason for the difference in I/Os is the removal of a bottleneck that was slowing the processing of message traffic.

Figures 9 and 10 are graphic representations of the external transaction rates and response times from the three runs. As can be seen, changing the value of the scheduling parameter has a much greater effect on IMS than it does on TSO, partly because IMS performs significantly more I/Os than TSO. Also, a bottleneck develops at the front-end processor on the IMS domain when lower scheduling values are used. Note that the worst IMS performance is associated with a scheduling parameter value of 1, the slowest domain-dispatching rate. Slower domain-dispatching rates cause longer average I/O service delays, which in turn cause longer response times. IMS response time and ETR improve significantly as the domain- dispatching rate increases. This is illustrated by table 3, which contains some results for the IMS domain including TPNS queue time, IMS output queue time, and the activity rate for the channel to which the Front End Processor (4705) is attached.

The TPNS queue time is the average time that a terminal must wait to transmit a message after it is ready to transmit (wait for poll). IMS output queue time is the amount of time a transaction response sits in the output queue awaiting VTAM processing. Both of these queue times are part of the overall response time. Both are also highly dependent on the availability of the 4705. If the 4705 is constrained, queue times and consequently response times will rise.

A value of 1 for the scheduling parameter corresponds with the lowest activity rate for the 4705, the worst ETR, and the longest response time. The majority of the response time (4.88 out of 5.76 seconds) was spent in a TPNS queue or the IMS output queue. As the value of the scheduling parameter increases, the I/O rate for the 4705 increases, ETRs increase, and queue times drop steadily to almost nothing. As a result, the system processes more transactions and CPU utilization now increases so much that the CPU is now the constraint to higher ETRs.

The longer time slice associated with lower scheduling parameter values restricts the even flow of messages across the 4705. The longer the time slice, the lower the frequency of domain dispatch, which causes the messages to be processed in surges. The high queue times and poor ETR and response times reflect these surges. As the duration of the time slice decreases, the frequency of domain dispatching increases and the flow of messages starts to even out as indicated by the reduction in queue times. This reduced queue time in turn causes response times to decrease and ETRs to increase.

The increase in IOSQ times associated with lower scheduling parameter values also contributes to the increased response times. The increase in IOSQ time had a very slight effect on overall response times when compared to our results with the 4705. However, these effects are magnified for slower devices (3350 versus 3380) and busier devices (greater than 10 I/Os per second).

The TSO workload had a much lower transaction arrival rate than the IMS workload (20 transactions/second versus 62.5 transactions/second) and was not affected as much by the changing value of the scheduling parameter, as illustrated in Table 4.

A lower transaction arrival rate translates into a less busy 4705. As noted earlier, the busier the device, the greater MDF's

effect on that device's throughput. The inverse is also true: the less busy the device, the less MDF's effect on its throughput. Consequently ETR and response time for TSO were not affected like they were for IMS.

TSO response time increases by 0.5 seconds whereas ETR shows a slight 2% decrease, when moving from parameter value 1 to 5. The TSO workload is more CPU bound and less I/O intensive than the IMS workload, so one would expect response times to increase as the scheduler parameter and, consequently, frequency of domain dispatching increases.

4.8 SUMMARY

Having a well-tuned I/O subsystem initially can minimize the effects of 580/MDF on I/O. However, the use of 580/MDF causes some I/O service delays that can prolong device response times for busy devices (approximately 10 I/Os per second and greater). In heavy I/O activity environments, increasing the scheduler parameter value may help reduce the impact of 580/MDF on overall I/O response time.

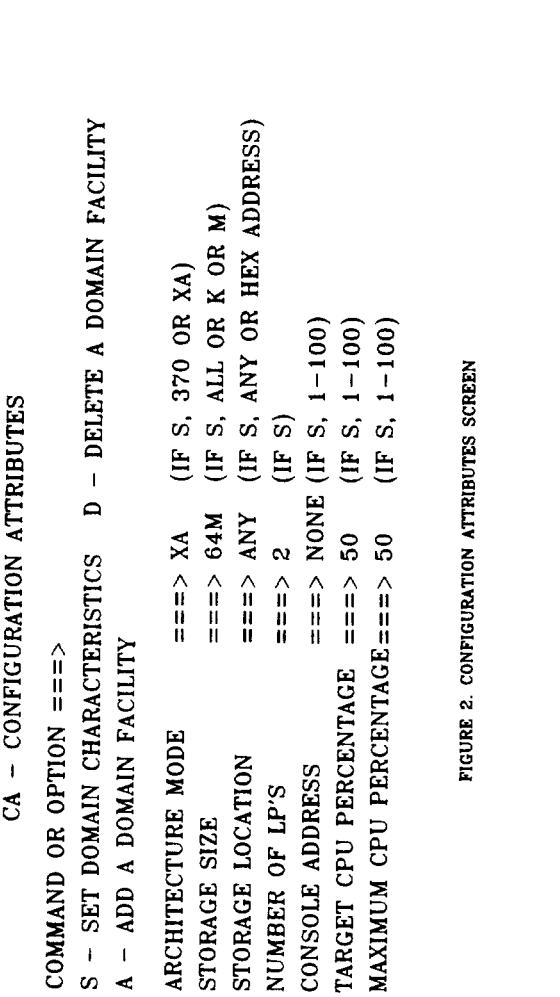

FIGURE 2. CONFIGURATION ATTRIBUTES SCREEN

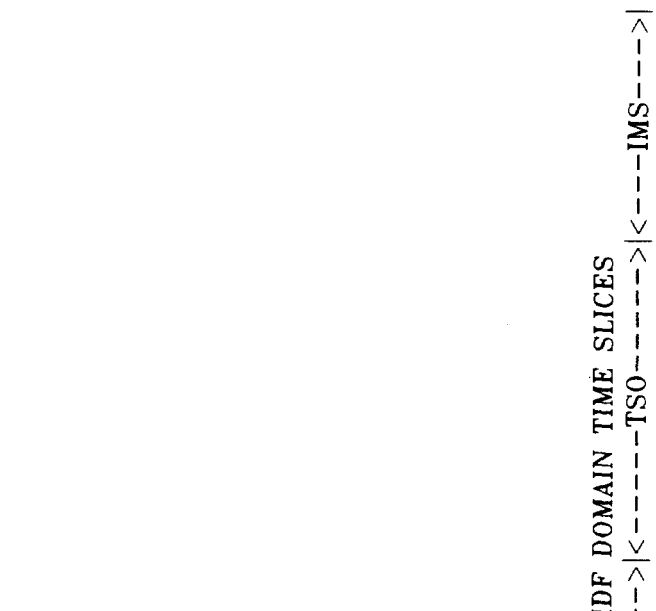

FIGURE 1. 580/MDF AND I/O CONCEPTS

```
        SS - SET SYSTEM SCHEDULING
COMMAND OR OPTION ===>
S - SET DOMAIN CPU PERCENTAGES
P - SET SCHEDULING PARAMETER

TARGET CPU PERCENTAGE  ===> 050 (IF S, 1-100)
MAXIMUM CPU PERCENTAGE===> 050 (IF S, 1-100)
SCHEDULING PARAMETER   ===> 1   (IF P, 1-5)

        DOMAIN       CPU UTILIZATION
        # NAME       TARG  MAX  NORM
        ----------------------------
        0 PEMDFI      50    50    50
        1 PEMDFT      50    50    50


MASTER DS1A @PEMDFI
```

FIGURE 3. SYSTEM SCHEDULING SCREEN

- 1,000 TERMINALS
  - VTAM/3270
  - TRANRESP
- 100 DATA BASES
  - 26 HDAM/OSAM
  - 26 HDAM/VSAM
  - 24 HIDAM/OSAM
  - 24 HIDAM/VSAM
- 100 APPLICATION PROGRAMS
  - RESPONSE MODE
  - PROCOPT = AP
- 60 MESSAGE PROCESSING REGIONS

FIGURE 4. IMS WORKLOAD

DB GU	41.01%
DB GNP	27.67%
DB GHU	5.03%
DB GHNP	3.72%
DB ISRT	0.90%
DB DLET	0.12%
DB REPLACE	3.45%
DC GU	11.43%
DC ISRT	6.66%
AVG DL/I CALLS/TRAN	15.02
AVG I/O PER TRAN	16.2
AVG CPU TIME/TRAN (ms)	10.0

FIGURE 5. IMS CALL PROFILE

DESCRIPTION	SPF OPTION	WORKLOAD %
GENERAL EDIT	2	5
GENERAL EDIT	2,1	5
EDIT,SUBMIT	2	5
SAS PRINT, PLOT	6	3
SAS CHART	6	2
SAS GLM	6	2
SAS ANALYZE SMF	6	3
BROWSE JOB	3.8	5
HELP, DELETE, RESET	3.5	5
HOUSEKEEPING	3.2,3.1	5
VTOC LIST	3.7	5
TSO COMMANDS	6	5
ASSEMBLER	4.1,3.2	10
COBOL COMPILE	4.2,3.2	4
PL/I COMPILE	4.5,3.2	4
FORTRAN COMPILE	6	4
EXEC COBOL	6	4
EXEC ASSEMBLER	6	1
EXEC PL/I	6	10
EXEC FORTRAN	6	2
FOCUS	6	10
LOGOFF	-	1

FIGURE 6 TSO WORKLOAD

	IMS			
SCHEDULING PARM.	1	3	5	BASE
AVG RESP	24.72	24.90	21.54	24.36
AVG IOSQ	7.00	6.54	3.45	5.63

	TSO			
SCHEDULING PARM.	1	3	5	BASE
AVG RESP	25.41	23.41	22.00	21.91
AVG IOSQ	4.75	3.83	2.66	3.41

TABLE 1. I/O RESPONSE TIME VERSUS SCHEDULER PARAMETER VALUE

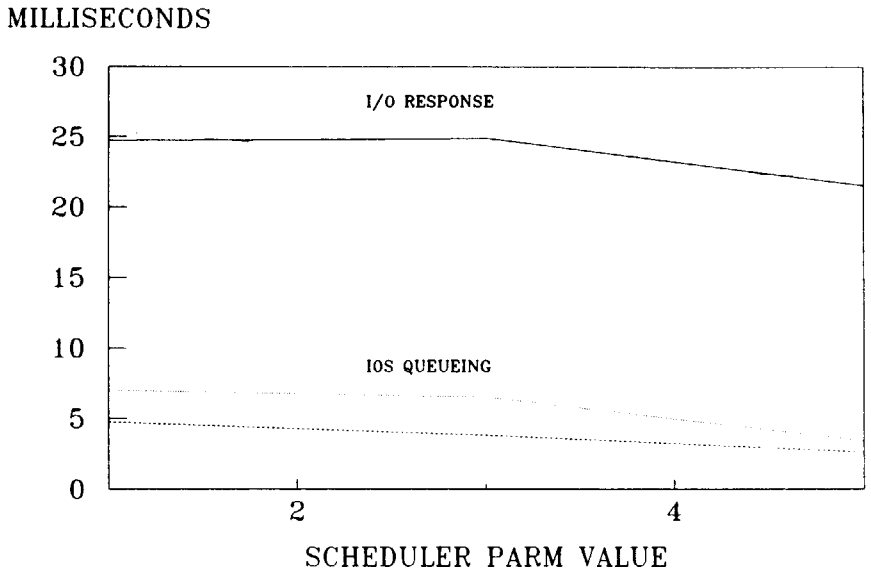

FIGURE 7. I/O RESPONSE TIME VERSUS SCHEDULER PARAMETER VALUE

SCHEDPARM =	1	3	5	BASE
IMS-C23 RATE	25.54	29.01	31.09	32.78
RESP	46	43	29	31
IOSQ	29	26	12	15
IMS-8C2 RATE	24.84	27.84	26.85	32.47
RESP	53	51	31	42
IOSQ	34	32	12	23
TSO-CC1 RATE	10.81	10.86	11.25	11.68
RESP	33	26	24	29
IOSQ	13	7	6	9
TSO-CC2 RATE	10.93	11.22	10.67	11.38
RESP	36	28	23	29
IOSQ	15	9	4	8

TABLE 2. DEVICE ACTIVITY RATES AND RESPONSE TIMES

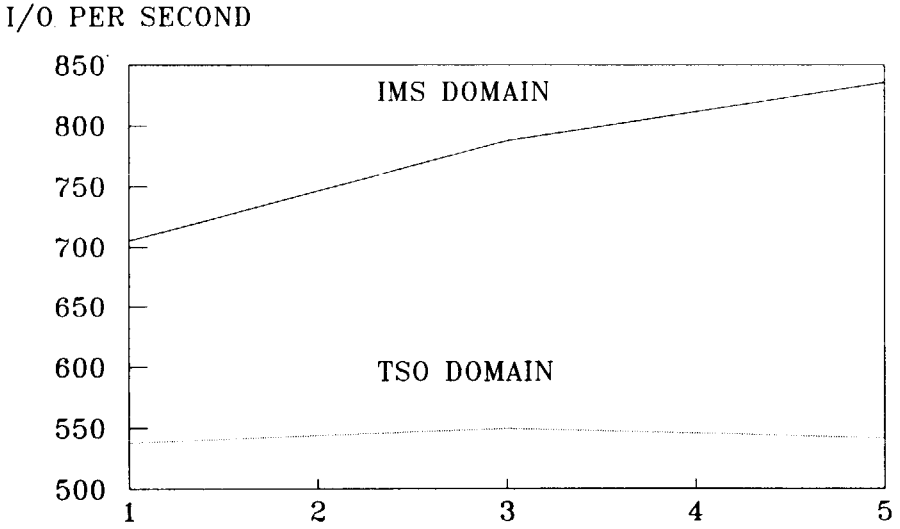

FIGURE 8. I/O PER SECOND VERSUS SCHEDULER PARM VALUE

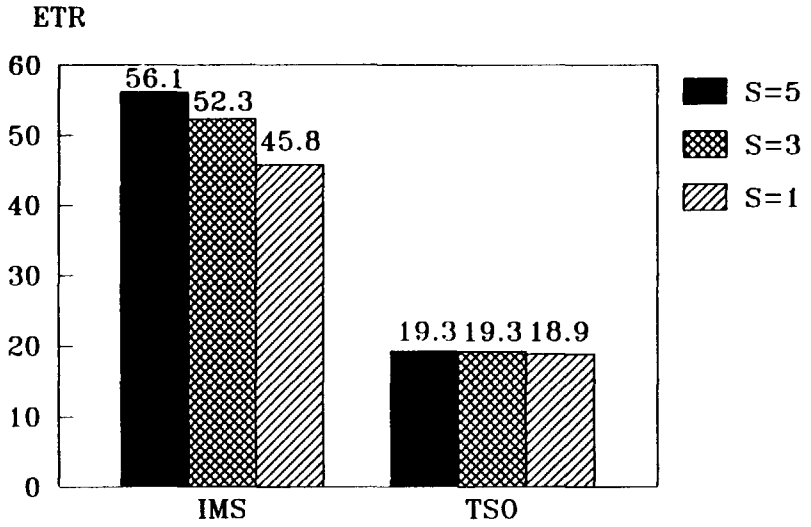

FIGURE 9. ETR VERSUS SCHEDULER PARM VALUE

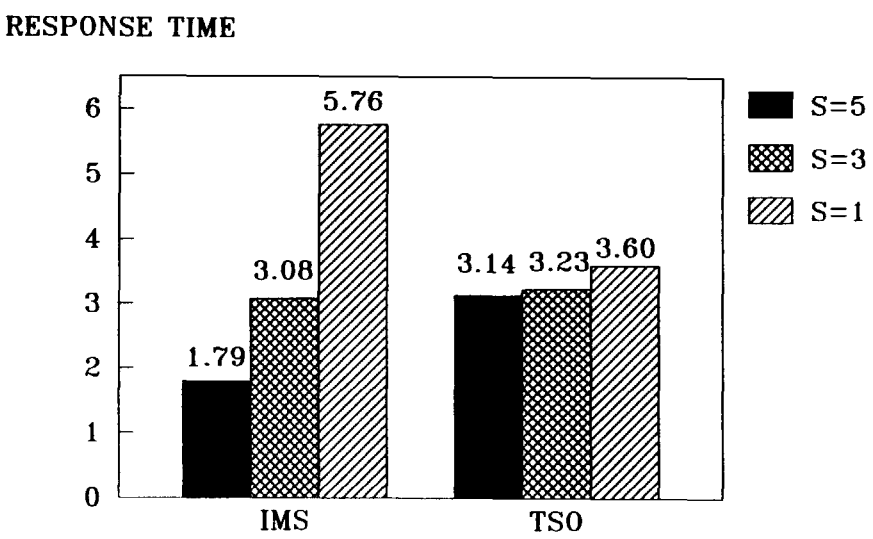

FIGURE 10. RESPONSE TIME VERSUS SCHEDULER PARM VALUE

SCHED. PARM.	4705 RATE	TPNS Q	IMS IN Q	IMS EXEC	IMS OUT Q	RESP TIME	ETR	CPU
1	58.88	2.78	0.2	0.6	2.1	5.76	45.89	82.28
3	64.28	1.43	0.2	0.6	0.7	3.08	52.31	92.58
5	75.41	0.07	0.6	0.9	0.0	1.79	56.13	99.68

TABLE 3. 4705 I/O RATES, TPNS AND IMS Q TIMES

SCHEDULING PARAMETER	RESPONSE	ETR	CPU
1	3.14	19.36	99.58
3	3.23	19.34	99.80
5	3.60	18.98	100.00

TABLE 4. SCHEDULER PARM EFFECTS ON TSO