System/36

Internals

COMMON Session 45B

Dallas, Texas

October 22, 1986

Table of Contents

System/36

Internals

By: Al Brown

1.0 OPERATING SYSTEM IMPLEMENTATION

- MULTIPLE USERS (MULTIPROGRAMMING)
- VIRTUAL
  - STORAGE SWAPPING

WHY SWAPPING?

A SIMPLE, EXCELLENT APPROACH TO MANAGE "OVER-COMMINTMENT" OF MAIN STORAGE FOR AN INTERACTIVE ENVIRONMENT (WITH A FINITE NUMBER OF USERS).

2.0 SYSTEM/36 OPERATING SYSTEM

THE SYSTEM/36 OPERATING SYSTEM IS THE "SYSTEM SUPPORT PROGRAM" (SSP):

. CONTROLS THE EXECUTION OF ALL JOBS ON THE SYSTEM AND MUST BE IN MAIN STORAGE BEFORE JOBS ARE RUN.

. RESIDES ON DISK OR DISKETTE AND IS PLACED INTO MAIN STORAGE BY THE IPL (INITIAL PROGRAM LOAD) PROCESS WHICH THE SYSTEM OPERATOR INITIATES.

. USER INTERFACE:
  - OCL (OPERATION CONTROL LANGUAGE) STATEMENTS
  - SSP PROCEDURES AND COMMANDS
  - SSP UTILITIES
  - USER APPLICATIONS

2.1 SYSTEM SOFTWARE STRUCTURE

FOR EXAMPLE:

EXTERNAL	UTILITIES (NON-SSP) QUERY/36 . DEF DW/36 PS/36 ETC.	PROGRAM PRODUCTS ASSM. LANGUAGE BASIC LANGUAGE COBOL LANGUAGE RPG II LANGUAGE ETC.	USER APPLICATIONS	MAIN
FACILITIES	OPERATION CONTROL LANGUAGE (OCL)	UTILITIES IDDU ETC.	SERVICES AIDS	STORAGE
	ENHANCED HELP			
	SYSTEM SUPPORT PROGRAM (SSP)			
INTERNAL FACILITIES	DATA HANDLING	SYSTEM SERVICE	SUPERVISOR	CONTROL
CSP SERVICES	INPUT/OUTPUT	SYSTEM SUPPORT	SUPERVISOR	STORAGE
DEVICE SERVICES	DISK, DISKETTE, AND TAPE	WORK STATION	COMMUNICATIONS	OTHER	INPUT/ OUTPUT

3.0 MAIN STORAGE CONCEPTS

MAIN STORAGE CONCEPTS INCLUDES:

. MAIN STORAGE STRUCTURE

  - DATA BUFFERING
  - PROGRAM ATTRIBUTES

. MAIN STORAGE ADDRESSING

. MAIN STORAGE MANAGEMENT

3.1 MAIN STORAGE STRUCTURE

MAIN STORAGE STRUCTURE INCLUDES:

SYSTEM AREA

. NUCLEUS

  - FIXED

  - VARIABLE

USER AREA

. PROGRAMS

  - USER

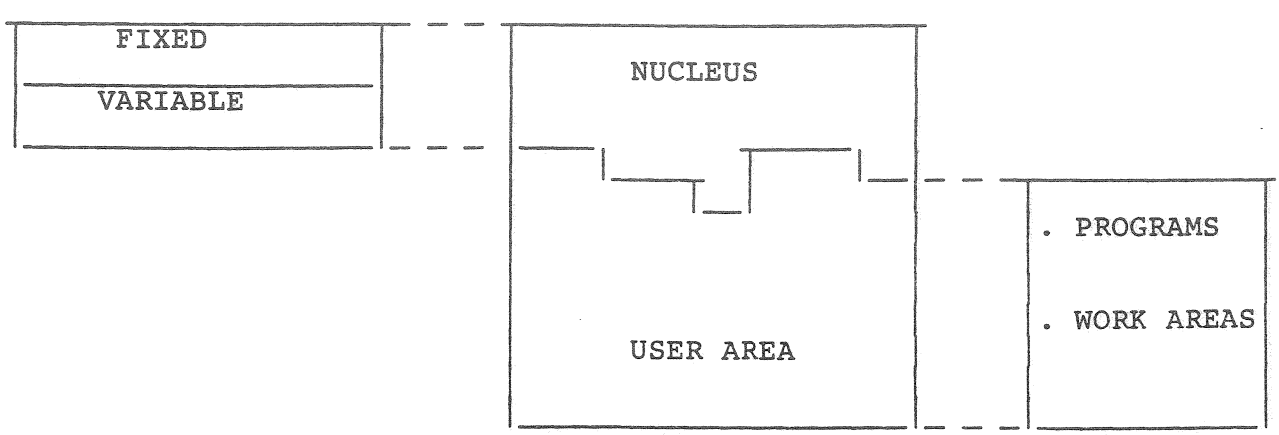
  - SYSTEM

. WORK AREAS

  - SYSTEM WORK SPACE (SWS)

  - TASK WORK SPACE (TWS)

FOR EXAMPLE:



3.1.1 SYSTEM AREA

THE SYSTEM AREA INCLUDES:

. A FIXED NUCLEUS

. VARIABLE NUCLEUS

3.1.1.1 FIXED NUCLEUS:

THE SYSTEM AREA ALWAYS INCLUDES A MINIMUM FIXED PORTION DEPENDING ON:

. MAIN STORAGE SIZES, AND

. THE SYSTEM/36 MODEL

MAIN STORAGE SIZE	MINIMUM FIXED NUCLEUS SIZE		
	5360 (1)	5362 (1)	5364 (1 AND 2)
128K - 256K	32K	32K	40K
512K	34K	34K	42K
1024K	36K	36K	44K
1792K	38K	N/A	N/A
2048K	40K	40K	N/A
3 MB - 7 MB	64K	N/A	N/A

NOTES:
1. IF DISPLAYWRITE/36 IS INSTALLED ON THE SYSTEM, ADD AN ADDITIONAL 2.25K.
2. INCLUDES AN ADDITIONAL 8K FOR PC TO CSP INTERFACE.

3.1.1.2 VARIABLE NUCLEUS

. THE VARIABLE NUCLEUS CONTENT DEPENDS ON THE OPTIONS SELECTED DURING CONFIGURATION AND/OR IPL:

VARIABLE NUCLEUS (NONSWAPPABLE SYSTEM ROUTINES AND SPACES)
. CONTROL BLOCKS AND POINTERS
. RESIDENT SYSTEM ROUTINES
. TRANSIENT SYSTEM ROUTINES
. SYSTEM QUEUE SPACE (SQS) (AKA: ASSIGN/FREE AREA)
- MEMORY RESIDENT FORMATS
- CACHE MANAGEMENT ROUTINES
. CACHE MEMORY

. THE SYSTEM AREA ALSO INCLUDES A VARIABLE PORTION DEPENDING ON THE ACTIVITY ON THE SYSTEM:

VARIABLE NUCLEUS (NONSWAPPABLE SYSTEM ROUTINES)	SIZES
- BATCH BSC INTERRUPT HANDLER	4 KB
- SDLC INTERRUPT HANDLER	8 KB
- SSP-ICF BSC INTERRUPT HANDLER	12 KB
- BSC 3270 INTERRUPT HANDLER	8 KB
- MSRJE BSC INTERRUPT HANDLER	10 KB
. X.25 INTERRUPT HANDLER	42 KB
. PRINTER DM AND SPOOL INTERCEPT	1 KB
. FOLDER MGR AND I/O ROUTER	2.25 KB

3.1.1.2.1 SYSTEM QUEUE SPACE ACCESS (Assign free)

NON - SWAPPABLE

. ASSIGN IN MULTIPLES OF 16 BYTES TO 2 KB

. QUICK FIT ALGORITHM

. AUTOMATIC RECOVERY WHEN HIGHLY UTILIZED:

  - TAKES 2 KB FROM USER STORAGE

  - DYNAMICALLY MAINTAINED

. CONTAINS SHARED CONTROL BLOCK DATA

. ONLY ADDRESSABLE BY PRIVILEGED PROGRAMS

THE FOLLOWING TABLE SHOWS THE APPROXIMATE AMOUNT OF ASSIGN/FREE AREA THAT IS USED FOR SPECIFIC ITEMS:

ITEMS	AMOUNT OF ASSIGN/FREE AREA NEEDED
ACTIVE PROGRAMS	512 BYTES
ACTIVE DISPLAY STATIONS	512 BYTES
ACTIVE PRINTERS	512 BYTES
ACTIVE FILES BEING USED	512 BYTES
DISK CACHE RESIDENT CODE	512 BYTES
DATA COMMUNICATIONS	VARIABLE AMOUNT BASED ON THE TYPE COMMUNICATIONS ACTIVE

NOTE: WHEN A FILE WITH MULTIPLE INDEXES IS USED, EACH INDEX IS AN ACTIVE FILE (EVEN WHEN THAT FILE IS NOT USED BY THE PROGRAM).

3.1.1.2.2 HOW MANY PROGRAMS CAN EXECUTE CONCURRENTLY

FORMULA: NUMBER OF JOBS = A - (B + C + D + E + F + G + H + I + J)

WHERE:

A = INSTALLED MEMORY SIZE

B = FIXED NUCLEUS SIZE

C = VARIABLE NUCLEUS SIZE, EXCLUDING CACHE MEMORY SIZE

D = (512 BYTES x THE NUMBER OF ACTIVE PROGRAMS)

E = (512 BYTES x THE NUMBER OF ACTIVE DISPLAY STATIONS PER PROGRAM)

F = (512 BYTES x THE NUMBER OF ACTIVE PRINTERS PER PROGRAM)

G = (512 BYTES x THE NUMBER OF ACTIVE FILES PER PROGRAM)

H = DATA COMMUNICATION SUPPORT PER ACTIVE PROGRAM (VARIABLE AMOUNT BASED ON THE TYPE COMMUNICATION ACTIVE)

I = CACHE MEMORY SIZE, IF ACTIVE

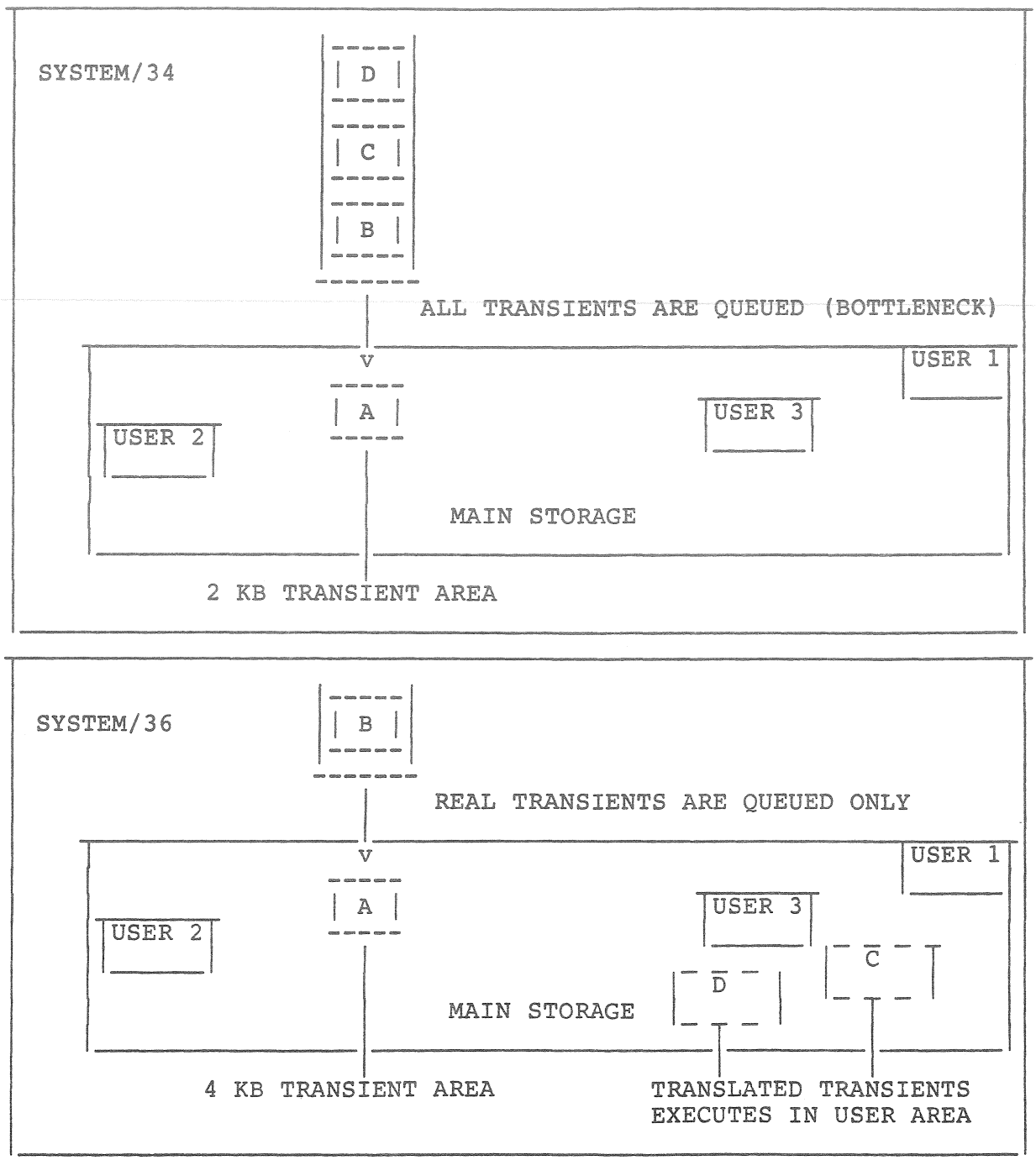
J = 512 BYTES FOR DISK CACHE RESIDENT CODE

NOTE: THE SYSTEM WILL GIVE YOU A WARNING:

- DURING IPL, IF LESS THAN 48K OF USER MEMORY IS REMAINING.
- DURING EXECUTION, WHEN USER MEMORY GOES BELOW 48K.

3.1.1.2.3 TRANSIENT AREA AND TRANSLATED TRANSIENTS

FOR EXAMPLE:



3.1.2 USER AREA

. MAIN STORAGE THAT IS NOT CURRENTLY PART OF THE NUCLEUS

. DO NOT HAVE TO BE CONTIGUOUS IN MAIN STORAGE

. DIVIDED INTO 2 KB "PAGES":

  - THE SYSTEM USES PAGES TO KEEP TRACK OF HOW PROGRAMS:

    -- ARE LOADED INTO MAIN STORAGE

    -- RUN IN MAIN STORAGE

    -- ARE SWAPPED IN AND OUT OF MAIN STORAGE

    -- USE DATA BUFFERS

3.1.2.1 USER PROGRAMS

THE SYSTEM VIEWS ALL PROGRAMS AS "TRANSLATED TRANSIENT":

. TRANSIENT ROUTINES:

  - ALLOCATE
  - COMMAND PROCESSOR (NOT MAINLINE)
  - INITIATOR
  - SSP-ICF MANAGEMENT
  - SYSIN
  - SYSLOG
  - TERMINATOR
  - WORK STATION MANAGEMENT (GET OPERATION)
  - WORK STATION MANAGEMENT (PUT AND PUT OVERRIDE OPERATION)

. LANGUAGE COMPILERS

. UTILITIES

. USER APPLICATIONS

  - MEMORY RESIDENT OVERLAYS

WORK SPACE INCLUDES:

SYSTEM WORK SPACE (SWS):

. TRACE BUFFER

. INSERT BUFFER (DATA MANAGEMENT)

. FORMAT INDEX AREA

. ACTIVE PROCEDURE TABLE

. WORK STATION BUFFER SPACE

TASK WORK SPACE (TWS):

. PROCEDURE PARAMETER SAVE AREA

. INITIATOR WORK AREA

. SYSIN WORK AREA

. DISK FILE SPACE:

  - DATA BUFFER

. COMMAND PROCESSOR WORK AREA

3.1.2.2 WORK SPACE CONSIDERATIONS

- VARY IN SIZE FROM 2 KB TO 1.75 MB
- TASK-RELATED OR SHARED AMONG MULTIPLE TASKS
- SPACE MAY BE ASSIGNED IN 64-BYTE MULTIPLES
- IDENTIFIED BY UNIQUE IDs
- NOT PHYSICALLY ATTACHED TO ANY PROGRAM:
  - MUST USE MAPPING FACILITY

3.2 DATA BUFFERING

3.2.1 USER PROGRAMS

. OBJECT CODE INCLUDES:

- EXECUTABLE INSTRUCTIONS
- WORK STATION BUFFER
- TABLES/ARRAYS

. OBJECT CODE DOES NOT INCLUDE:

- PRINT AND DISK BUFFERS
- FORMAT INDEX AREA
- STORAGE INDEX

USER PROGRAMS:

. DATA BUFFERS ARE NOT INCLUDED IN THE PROGRAM; THEY ARE IN A "DISK FILE SPACE" WHICH CAN BE EITHER:

  - APPENDED TO THE PROGRAM

  - STORED IN A TASK WORK SPACE

. MAXIMUM SIZE IS 44 KB PER FILE

. THE DISK FILE SPACE CONTAINS:

  - RECORD BUFFER

  - INDEX BUFFER

  - INDEX BLOCK

  - CONTROL BLOCK (200 BYTES)

3.2.1.1 "DISK FILE SPACE" - APPENDED

. IN ORDER TO GET AN APPENDED BUFFER:

- PROGRAM AND BUFFER SPACE SIZE MUST BE 64 KB OR LESS
- SYSTEM MUST HAVE SPACE IN USER AREA

. WHEN BUFFERS ARE APPENDED TO A PROGRAM, THEY ARE SWAPPED WITH THE PROGRAM

NOTE:

THIS IS THE PREFERRED METHOD. IT IS FASTER THAN NONAPPENDED BUFFERS.

3.2.1.2 "DISK FILE SPACE" - NONAPPENDED

. THE TASK WORK SPACE (TWS) IS A SEPARATE ADDRESS SPACE

. THE TWS REQUIRES MAPPING SERVICES FOR PROGRAMS TO ADDRESS THEM

. BUFFERS ARE SWAPPED SEPARATELY FROM PROGRAMS

NOTE:

THIS METHOD IS MUCH SLOWER THAN APPENDED BUFFERS.

3.3 PROGRAM ATTRIBUTES

THERE ARE TWO TYPES OF PROGRAM ATTRIBUTES:

. EXECUTION ATTRIBUTES:

  - RELOADABLE
  - REUSABLE
  - REENTRANT

. STORAGE ATTRIBUTES:

  - NON-SWAPPABLE
  - SWAPPABLE
  - REFRESHABLE

3.3.1 EXECUTION ATTRIBUTES

. RELOADABLE:

- ONE COPY SERVICES ONE USER
- A NEW COPY OF THE CODE IS LOADED FOR EVERY USER
- CODE CAN BE CHANGED WHILE IN USE
- USER PROGRAMS, EXCLUDING BASIC PROGRAMS

. REUSABLE:

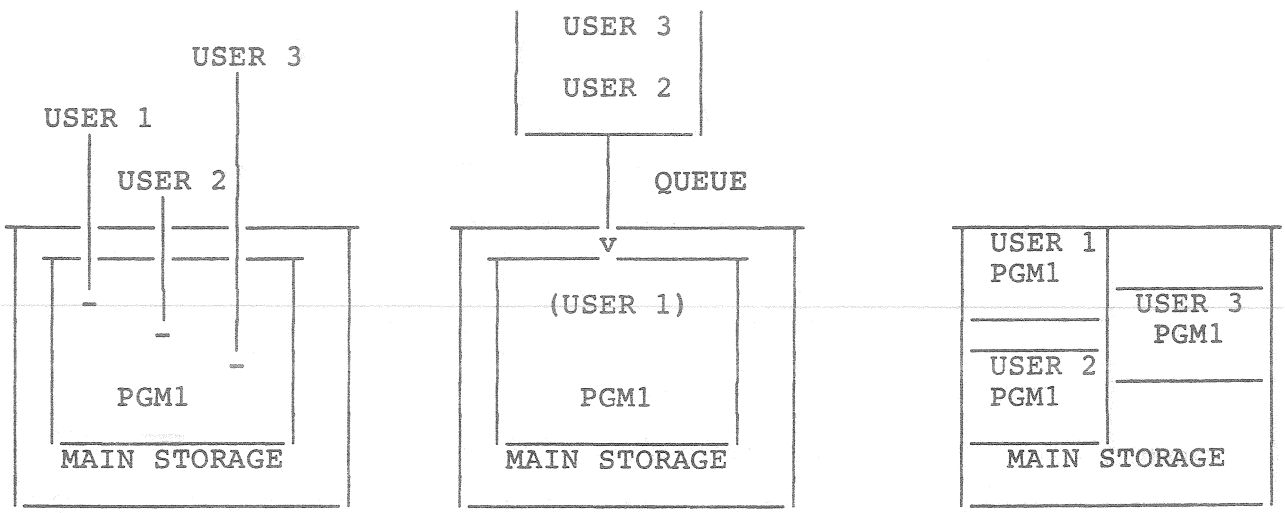
- ONE COPY SERVICES SEVERAL USERS, BUT:
  - -- ONE USER AT A TIME
  - -- OTHER USERS ARE QUEUED
- INTERRUPTIBLE AT A "SOFT STOP"
- CODE CAN BE CHANGED, BUT:
  - -- REINITIALIZED EACH TIME IT IS USED
- SYSTEM PROGRAMS ONLY

. REENTRANT:

- ONE COPY SERVICES SEVERAL USERS
- INTERRUPTIBLE - ANYWHERE
- CODE CANNOT BE CHANGED WHILE IN USE
  - SYSTEM PROGRAMS ONLY

3.3.1.1 EXECUTION ATTRIBUTES - EXAMPLE

EXAMPLE: THREE (3) USERS REQUESTING THE SAME PROGRAM (PGM1).



REENTRANT

- 1 COPY OF PGM1
- CAN BE SHARED BY ALL USERS CONCURRENTLY

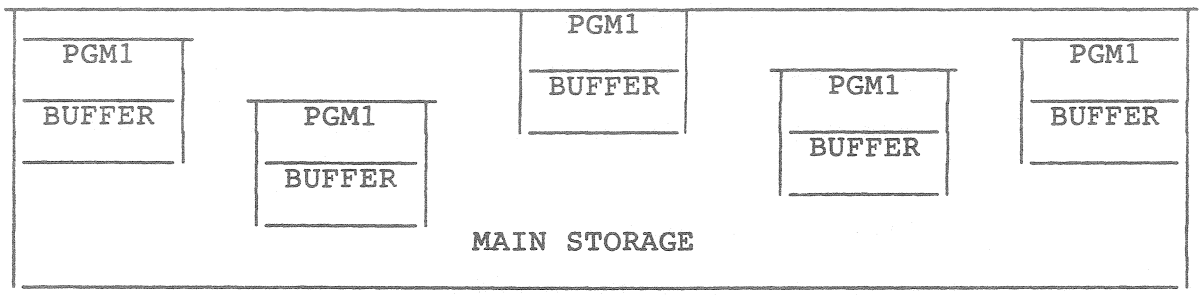
REUSABLE

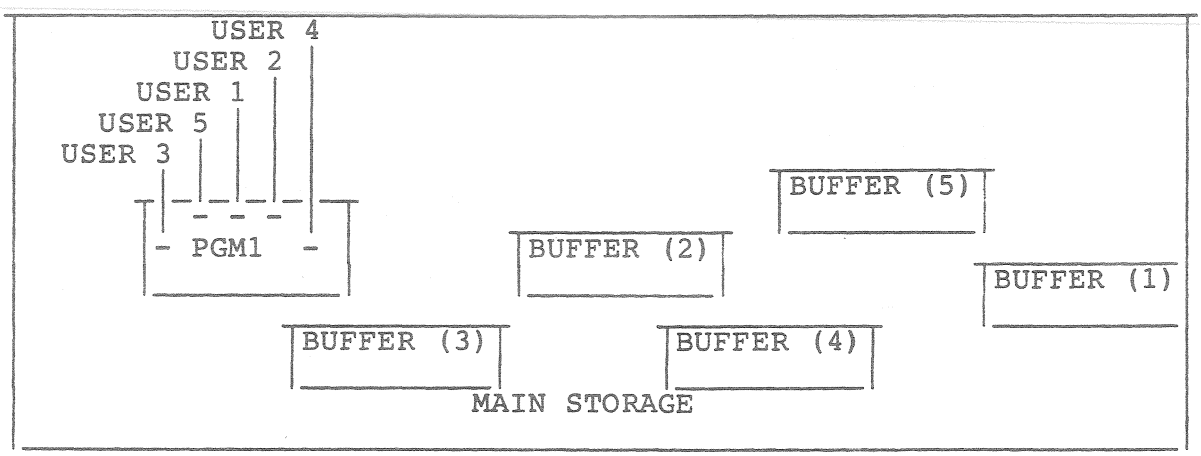
- 1 COPY OF PGM1
- CAN BE SHARED BY ALL USERS, BUT SERIALLY (PGM1 SERIALLY-REUSABLE)
- USERS ARE QUEUED

RELOADABLE

- 3 COPIES OF PGM1
- EACH COPY OWNED BY A DIFFERENT USER (PGM1 NONSHAREABLE)

3.3.1.2 EXAMPLE - REENTRANT CODE MAJOR BENEFIT





. LESS STORAGE REQUIRED TO GET THE SAME NUMBER OF JOBS DONE:

- PGM1 = 24 KB
  BUFFER = 6 KB

- FIVE (5) COPIES OF PGM1 WITHOUT REENTRANCY REQUIRES 150 KB

- A SINGLE COPY OF PGM1 WITH REENTRANCY REQUIRES 54 KB

3.3.2 STORAGE ATTRIBUTES

. SWAPPABLE (NONPINNED)

  - PROGRAM SWAPPED TO (TEMPORARILY STORED) DISK
  - SYSTEM DEFAULT FOR ALL USER APPLICATION PROGRAMS

. NONSWAPPABLE (PINNED)

  - PROGRAM IN MAIN STORAGE ALL THE TIME
  - SYSTEM PROGRAMS ONLY

. REFRESHABLE (READ ONLY)

  - NEVER SWAPPED TO DISK
  - A NEW COPY OF THE PROGRAM IS LOADED WHEN NEEDED

3.3.3 PROGRAM ATTRIBUTES SUMMARY

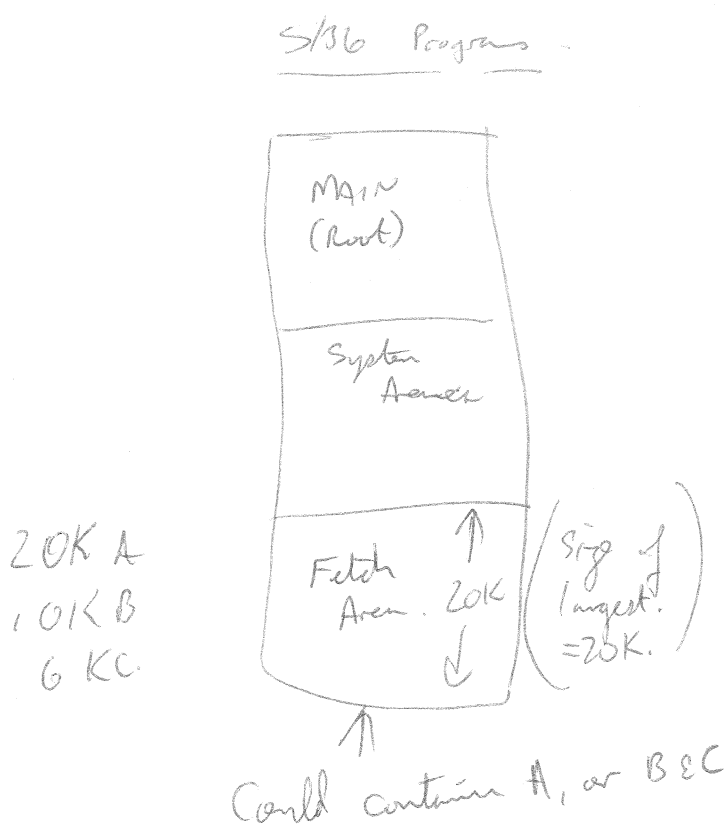
STORAGE ATTRIBUTES	EXECUTION ATTRIBUTES		
	REENTRANT	REUSABLE	RELOADABLE
SWAPPABLE (USER AREA)			. UTILITIES . USER PROGRAMS . PROGRAM PRODUCTS . DATA ENTRY FACILITY (DEF)
NONSWAPPABLE (SYSTEM AREA)	. FIXED NUCLEUS . VARIABLE NUCLEUS		
REFRESHABLE	. TRANSLATED SYSTEM PROGRAMS . BASIC PROGRAMS . QUERY/36	. REAL SYSTEM TRANSIENT PROGRAMS	

3.4 PROGRAM OVERLAY STRUCTURE

CONCEPTS

SWAPPING AND OVERLAYING ARE TWO METHODS THE SYSTEM/36 USES TO CONTROL THE USE OF MAIN STORAGE (OR MEMORY). THE OVERLAY STRUCTURE ALLOWS A PROGRAM TO EXECUTE USING LESS STORAGE THAN IT ACTUALLY NEEDS. THAT IS, OVERLAYING:

- REDUCES THE AMOUNT OF STORAGE COMMITTED BY A PROGRAM.
- LOADS FOR THE LIBRARY ONLY THOSE PORTIONS OF THE PROGRAM WHICH ARE NEEDED AT ANY GIVEN TIME.



AFTER A MAINLINE PROGRAM OR SUBROUTINE HAS BEEN COMPILED, THE OVERLAY LINKAGE EDITOR:

1. STORES THE OBJECT MODULE IN A LIBRARY AND CATALOGS IT IN THE LIBRARY DIRECTORY AS A SUBROUTINE MEMBER.

2. ATTEMPTS AFTER CREATING A LOAD MODULE, TO FIT A LINK-EDIT LOAD MODULE INTO THE AMOUNT OF MAIN STORAGE CURRENTLY AVAILABLE TO A PROGRAM.

3. IF ITEM 2 ABOVE CANNOT BE DONE, DIVIDES THE LOAD MODULE INTO SEGMENTS AND SUES THE SAME BLOCKS (OR AREA) OF MAIN STORAGE REPEATEDLY BY OVERLAYING OR REPLACING EACH SEGMENT AS IT IS NO LONGER NEEDED BY THE PROGRAM.

THE OVERLAY LINKAGE EDITOR:

. CAN BE CALLED DIRECTLY BY THE COMPILER, OR

. IT CAN BE EXECUTED BY YOU

3.4.1 OVERLAYS

YOU CAN OFTEN SOLVE COMPLEX DATA PROCESSING PROBLEMS BY USING SEPARATELY COMPILED BUT LOGICALLY INTERDEPENDENT PROGRAMS THAT, AT EXECUTION TIME, FORM LOGICAL AND PHYSICAL SUBDIVISIONS OF A SINGLE RUN UNIT. A RUN UNIT IS THE TOTAL MACHINE-LANGUAGE PROGRAM NECESSARY TO SOLVE A PROCESSING PROBLEM; IT INCLUDE ONE OR MORE OBJECT PROGRAMS, AND CAN INCLUDE OBJECT FROM SOURCE PROGRAMS WRITTEN IN ANY OF THE FOLLOWING LANGUAGES:

- ASSEMBLER
- COBOL
- FORTRAN IV

WHEN YOU SUBDIVIDE THE SOLUTION OF A PROBLEM INTO MORE THAN ONE PROGRAM, THE CONSTITUENT PROGRAMS MUST BE ABLE TO COMMUNICATE WITH EACH OTHER EITHER:

- THROUGH TRANSFER OF CONTROL
- THROUGH REFERENCE TO COMMON DATA

TRANSFER OF CONTROL

A CALLING PROGRAM CAN TRANSFER CONTROL TO A CALLED PROGRAM, AND THE CALLED PROGRAM CAN ITSELF TRANSFER CONTROL TO YET ANOTHER CALLED PROGRAM; HOWEVER, A CALLED PROGRAM MUST NOT DIRECTLY CALL THE PROGRAM THAT CALLED IT.

WHEN CONTROL PASSES TO THE CALLED PROGRAM, THE PROGRAM RUNS NORMALLY. WHEN A CALLED PROGRAM COMPLETES PROCESSING, THE PROGRAM CAN DO ANY OF THE FOLLOWING:

- TRANSFER CONTROL BACK TO THE CALLING PROGRAM
- CALL ANOTHER PROGRAM
- END THE RUN UNIT

COMMON DATA

PROGRAM INTERACTION MIGHT REQUIRE THAT BOTH PROGRAMS HAVE ACCESS TO THAT SAME DATA.

IN A CALLING PROGRAM, DESCRIBE COMMON DATA ITEMS IN THE SAME MANNER AS ANY OTHER DATA ITEMS. ALLOCATE STORAGE FOR THESE ITEMS IN THE CALLING PROGRAM.

3.4.2 SEGMENTATION

THE SEGMENTATION FEATURE LETS YOU EXECUTE A LARGE PROGRAM IN A SMALLER AREA OF MAIN STORAGE. YOU USE THE SEGMENTATION FEATURE TO BREAK A PROGRAM INTO A NUMBER OF OVERLAYS (THAT IS, A NUMBER OF SECTIONS THAT ARE SWAPPED INTO AND OUT OF THE SAME AREA OF STORAGE AS REQUIRED FOR PROGRAM EXECUTION).

- FOR COBOL PROGRAMS, BY WRITING SEGMENT-NUMBER ON EACH SECTION HEADER, YOU CAN CAUSE EACH SECTION TO BE STORED AS A PHYSICALLY INDEPENDENT SEGMENT.

- FOR RPG II PROGRAMS, THE OVERLAY LINKAGE EDITOR CONTROLS HOW PROGRAMS ARE SEGMENTED, NOT THE PROGRAMMER.

3.4.2.1 TYPES OF SEGMENTS

THERE ARE TWO TYPES OF PROGRAM SEGMENTS:

- FIXED SEGMENTS
- INDEPENDENT SEGMENTS

FIXED SEGMENTS ARE DIVIDED INTO TWO TYPES:

- SEGMENTS THAT ARE NEVER OVERLAID (FIXED PERMANENT SEGMENTS)
- SEGMENTS THAT ARE OVERLAID (FIXED OVERLAYABLE SEGMENTS)

INDEPENDENT SEGMENTS:

- CAN OVERLAY AND BE OVERLAID BY OTHER SEGMENTS DURING PROGRAM EXECUTION.
- IS MADE AVAILABLE IN ITS INITIAL STATE THE FIRST TIME YOU PASS CONTROL TO IT.

3.4.3 MEMORY RESIDENT OVERLAYS

MEMORY RESIDENT OVERLAYS ALLOWS THE USER:

. TO IMPROVE THE PERFORMANCE FOR PROGRAMS WHICH MAKE USE OF OVERLAYS BUILT AND LINKED BY THE OVERLAY LINKAGE EDITOR.

. TO KEEP A LARGE USER PROGRAM OVERLAY SEGMENTS IN MAIN STORAGE

. TO REDUCE THE THE NUMBER OF DISK READ OPERATIONS ISSUED FOR A PROGRAM ON EACH SUBSEQUENT LOAD OF AN OVERLAY

. TO REDUCE THE THE NUMBER OF DISK WRITE OPERATIONS THAT MAY OCCUR AS A RESULT OF SOMETHING BEING SWAPPED OUT

THE NUMBER OF OVERLAYS THAT ACTUALLY REMAIN IN MAIN STORAGE DEPENDS ON:

. THE SIZE OF EACH OVERLAY.

. THE SIZE OF THE AVAILABLE MAIN MEMORY.

. THE NUMBER AND SIZE OF OTHER PROGRAMS BEING EXECUTED.

4.0 ADDRESSING

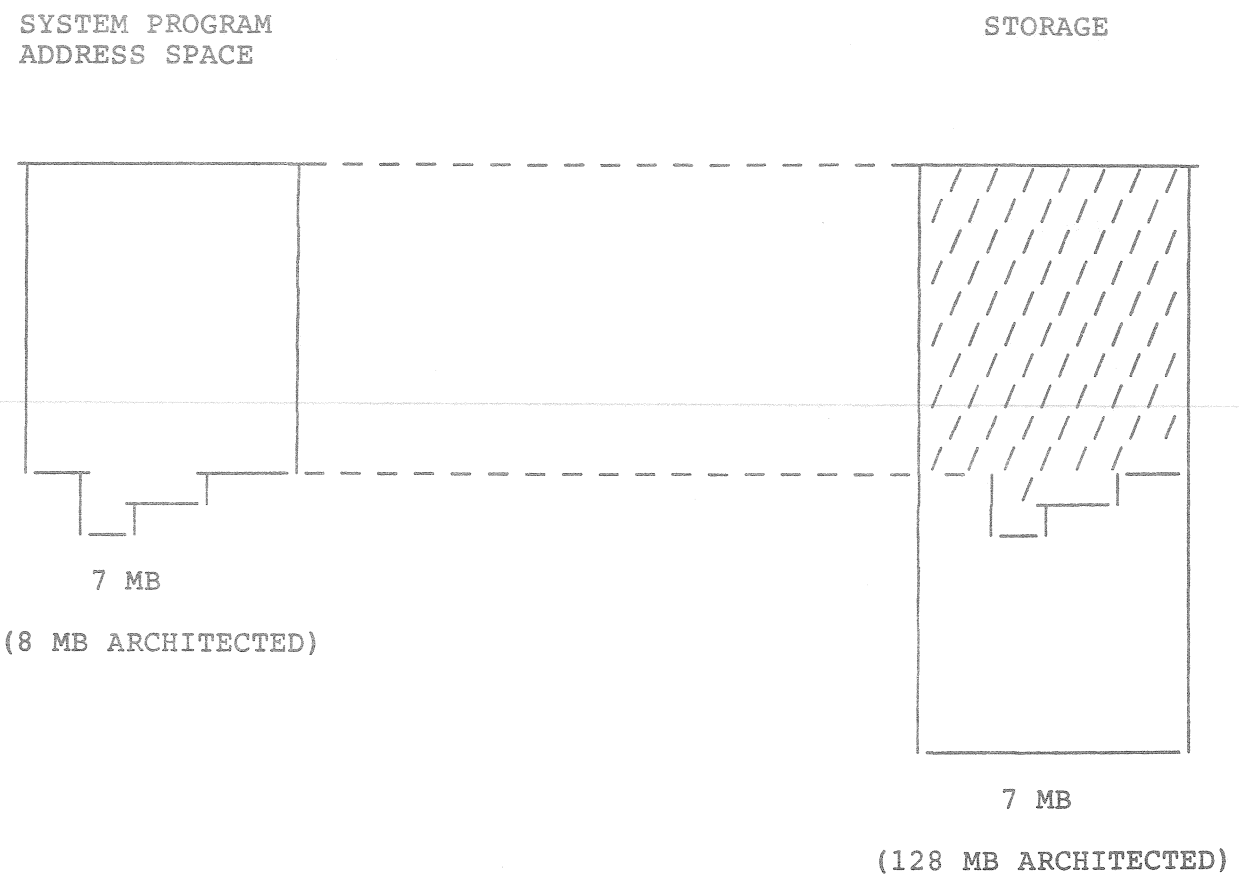
THE SYSTEM HAS TWO MODES OF ADDRESSING:

. REAL ADDRESSING

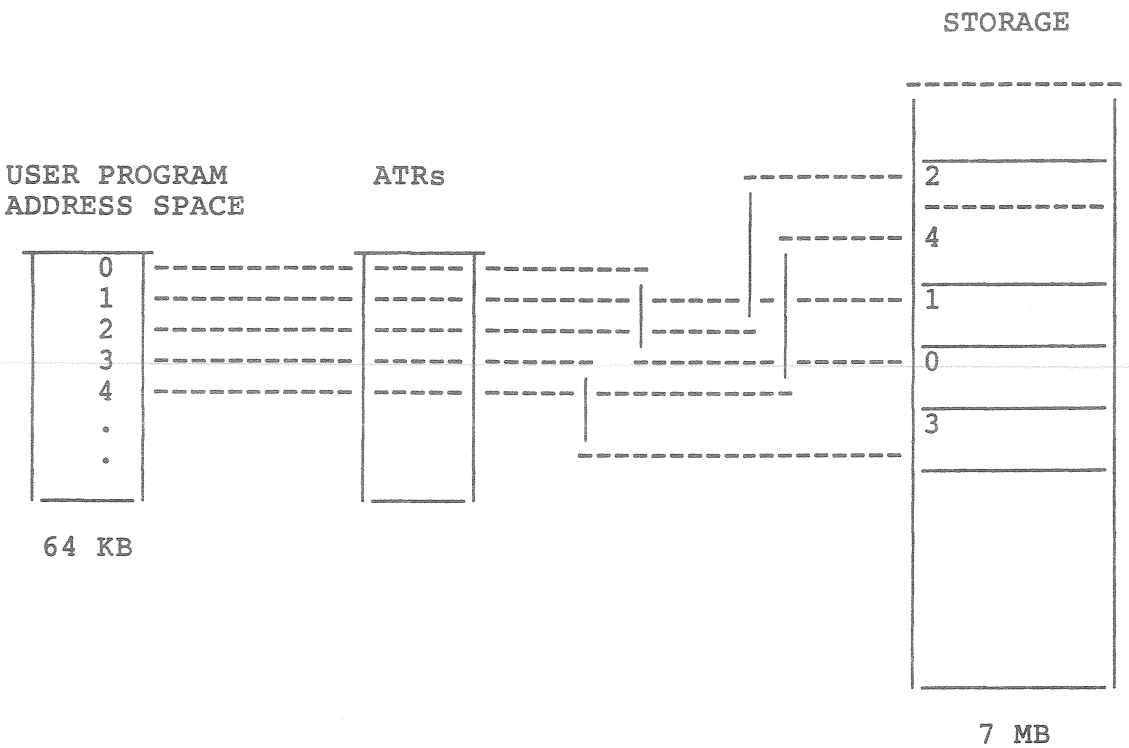
. TRANSLATED ADDRESSING

SYSTEM PROGRAMS CAN USE EITHER OR BOTH MODES WHILE USER PROGRAMS CAN ONLY USE THE TRANSLATED ADDRESSING MODE.

4.1 REAL ADDRESSING OVERVIEW



4.2 TRANSLATED ADDRESSING OVERVIEW



4.3 PREFIX ADDRESS CONTAINED TRANSLATION

MAIN STORAGE ADDRESSING (REAL) BEYOND 64 KB CAN BE ACCOMPLISHED WITH 2-BYTE ADDRESSES BY THE USE OF PREFIX ADDRESS CONTAINED TRANSLATION (PACT) REGISTER. THIS ALLOWS A "SIMULATION OF 3-BYTE ADDRESSING".

. A BIT IN THE ADDRESS DESCRIBES WHETHER THE ADDRESS IS TRANSLATED:

- PROVIDES FOR EXPANSION OF USER REGION BEYOND 64 KB
- REMOVES NUCLEUS LIMITATIONS (NUMBER OF SIMULTANEOUS EVENTS)
- KNOWLEDGE OF PROCESSOR OPERATION IS NOT REQUIRED OF THE SYSTEM PROGRAMMER

4.4 REGION MAPPING

. ALTER TASK ADDRESSABILITY AT 2 KB BOUNDARIES

. MAP TO:

  - REQUESTOR'S REGION
  - ANY REGION CURRENTLY ACTIVE
  - WORK SPACE (SWS OR TWS)

. DATA IS NOT MOVED

. LOGICAL ADDRESS IS COMPUTED

4.5 ADDRESS TRANSLATION REGISTERS (ATRS)

. CAN ADDRESS MAIN STORAGE UP TO A MAXIMUM STORAGE SIZE OF 2 KB

. THERE ARE 128 ATRs:

  - 64 FOR PROGRAM ADDRESSING:

    -- TWO SET OF 32

  - 8 FOR PACT REGISTER ADDRESSING:

    -- SINGLE BYTE REGISTERS

  - 56 FOR INPUT/OUTPUT USE

5.0 PROGRAM EXECUTION CHARACTERISTICS

1. PROGRAM A IS LOADED (SWAPPED-IN) FROM SECONDARY STORAGE
2. PROGRAM A MUST EXECUTE IN THE MAIN STORAGE PROCESSOR (MSP)
3. PROGRAM A IS DISPATCHED TO THE MSP BY PRIORITY; PRIORITY DEPENDS ON:
   A. USER DESIGNATION OR DEFAULT
   B. FREQUENCY OF A WORK STATION READ OPERATION
   C. THE NUMBER OF USERS OF THE PROGRAM
4. PROGRAM A EXECUTES IN THE MSP UNTIL ONE OF FIVE EVENTS OCCURS:
   A. THE PROGRAM COMES TO A "LONG WAIT" CONDITION
   B. THE PROGRAM ISSUES I/O REQUEST
   C. THE PROGRAM'S UTILIZATION TIME INTERVAL EXPIRES
   D. THE PROGRAM IS PREEMPTED BY A SIGNIFICANTLY HIGHER PRIORITY PROGRAM
   E. THE PROGRAM TERMINATES
5. PROGRAM B IS LOADED INTO MAIN STORAGE (SWAPPED-IN) FOR EXECUTION. IF MAIN STORAGE IS UNAVAILABLE, THE MAIN STORAGE OF THE MOST ELIGIBLE ACTIVE TASK IS SELECTED AND TRANSFERRED TO SECONDARY STORAGE (SWAPPED-OUT) TO MAKE ROOM FOR THE INCOMING PROGRAM
6. STEP 4 IS REPEATED FOR PROGRAM B
7. PROGRAM C IS SWAPPED-IN. THE PROCESS CONTINUES AS FOR PROGRAM B.

6.0 TASK MANAGEMENT

FUNCTION

- CONTROLS THE TASK ACTIVITY FOR THE MAIN STORAGE PROCESSOR.
- ASSIGNING EACH TASK A STATE (OR CONDITION):
  - A WAIT STATE,
  - A READY STATE, OR
  - AN ACTIVE STATE.

  (BOTH THE TASK DISPATCHER AND MSP CONTROL HANDLE ACTIVE TASKS.)

6.1 WAIT STATE

WAIT STATE OF A TASK MEANS THE TASK IS WAITING ON THE ACTIVE PROGRAM LIST QUEUE FOR SOMETHING TO HAPPEN OR TO FINISH BEFORE THE TASK IF CONSIDERED READY. THE TASK COULD BE WAITING FOR SOME I/O TO BE COMPLETED. FOR EXAMPLE:

A "LONG WAIT" STATUS IS GIVEN TO A USER TASK IMMEDIATELY AFTER AN ACCEPT OR GET OPERATION TO A WORK STATION HAS BEEN ISSUED. THS USER TASK REMAINS IN "LONG WAIT" STATUS UNTIL THE WORK STATION OPERATOR PRESSES THE ENTER KEY, ANY COMMAND KEY, OR ANY FUNCTION KEY.

6.2 READY STATE

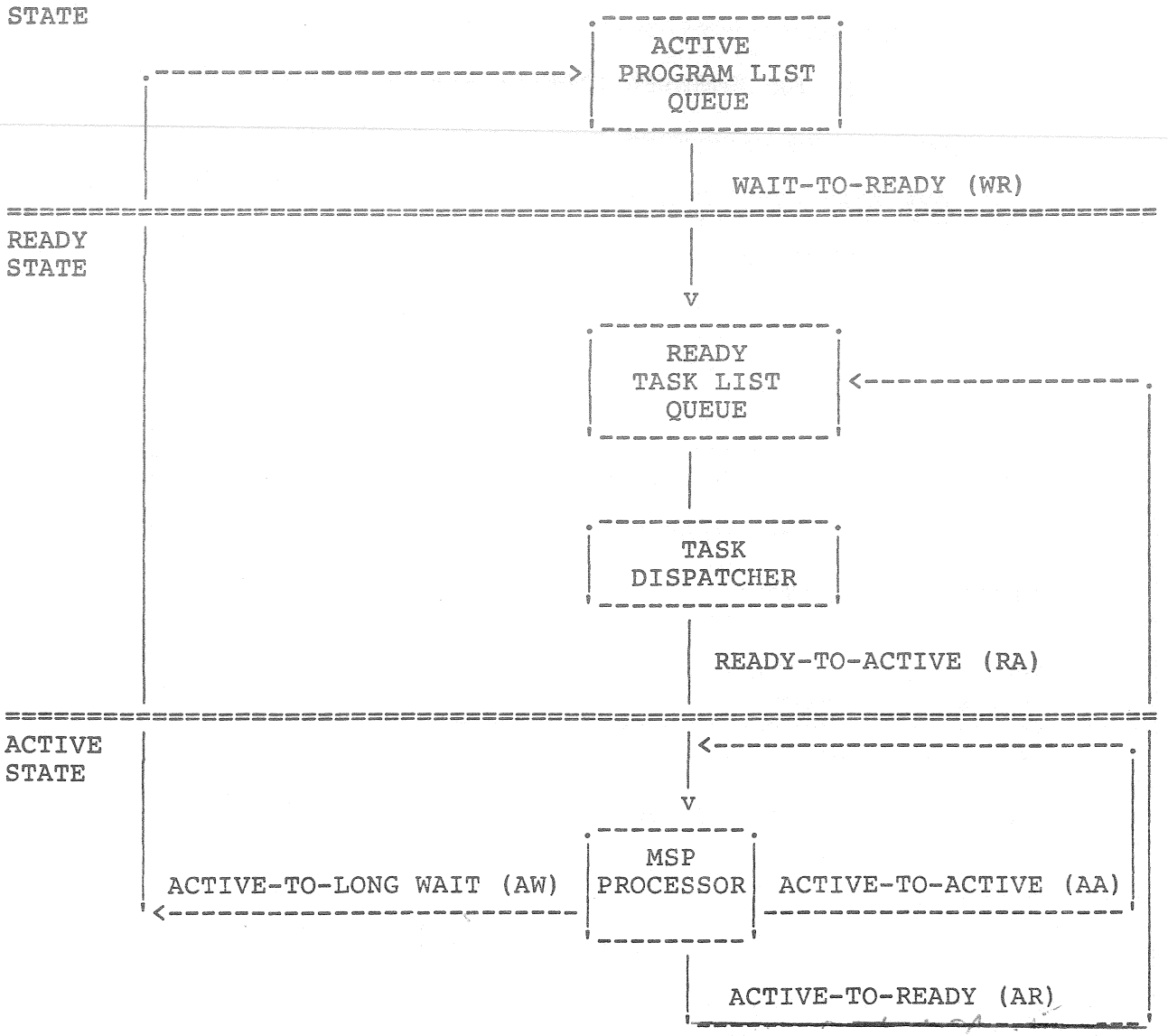
THE READY STATE OF A TASK MEANS THE TASK IS NOT CURRENTLY EXECUTING IT IS ON THE READY TASK LIST QUEUE), BUT HAS ALL THE REQUIREMENTS IT NEEDS TO EXECUTE.

6.3 ACTIVE STATE

THE ACTIVE STATE OF A TASK MEANS THE TASK IS NOW RUNNING IN THE MSP.

6.4 TASK EXECUTION TRANSITIONS

IT IS IMPORTANT TO UNDERSTAND HOW TASKS ARE QUEUED AND COMPETE FOR SYSTEM RESOURCES. THE DIAGRAM BELOW SHOWS THE VARIOUS TRANSITIONS TASKS CAN MAKE BETWEEN THE WAIT, READY, AND ACTIVE STATES.



A BRIEF DESCRIPTION OF EACH OF THE TRANSITIONS FOLLOWS:

(AA) ACTIVE-TO-ACTIVE. THE TASK WILL NOT GIVE UP THE MSP WHEN R-TIME OR MSP-TIME IS EXCEEDED, BECAUSE THERE ARE NO OTHER TASKS OF EQUAL OR HIGHER PRIORITY WAITING IN THE READY TASK LIST QUEUE.

(AR) ACTIVE-TO-READY. THE TASK GIVES UP THE MSP BECAUSE IT EXCEEDS ITS MSP-TIME AND THERE ARE OTHER TASKS OF EQUAL OR HIGHER PRIORITY WAITING IN THE READY TASK LIST QUEUE. THE TASK IS PUT INTO THE READY TASK LIST QUEUE IN FIFO ORDER WITHIN ITS PRIORITY GROUP.

(AW) ACTIVE-TO-LONG WAIT. THE TASK GIVES UP THE MSP AS A RESULT OF A LONG WAIT. (A PROGRAM DOES A READ OPERATION TO A WORK STATION). THE TASK IS PUT INTO THE ACTIVE PROGRAM LIST QUEUE IN FIFO ORDER WITHIN ITS PRIORITY GROUP.

(RA) READY-TO-ACTIVE. THE TASK IS SELECTED TO EXECUTE FROM THE READY TASK LIST IN FIFO ORDER WITHIN ITS PRIORITY GROUP.

(WR) WAIT-TO-READY. THE OPERATION THAT PUT THE TASK IN A LONG WAIT STATUS COMPLETES. THE TASK IS PUT INTO THE READY TASK LIST QUEUE IN FIFO ORDER WITHIN ITS PRIORITY GROUP.

6.4.1 ACTIVE PROGRAM LIST QUEUE

THE SYSTEM MAINTAINS A LIST (OR QUEUE) TO KEEP TRACK OF THE STATUS OF ALL ACTIVE PROGRAMS

AN ACTIVE PROGRAM CAN BE EITHER:

- IN MAIN STORAGE
- IN THE SWAP AREA ON DISK (THAT IS, THE TASK WORK AREA -- TWA)

THE SYSTEM USES THE POSITION OF THE PROGRAM ALONG WITH THE EXECUTION PRIORITY TO DECIDE WHICH PROGRAMS TO SWAP IN AND OUT OF MAIN STORAGE.

ACTIVE PROGRAMS	PROCESSING PRIORITY	POSITION ON THE ACTIVE PROGRAM LIST QUEUE
PROGRAM A	HIGH = 196	1
PROGRAM B	MEDIUM = 132	3
PROGRAM C	LOW = 68	5
PROGRAM D	HIGH = 196	2
PROGRAM E	MEDIUM = 132	4
PROGRAM F	LOW = 68	6

6.4.2 READY TASK LIST QUEUE

WHEN A TASK IS ON THIS LIST (OR QUEUE), IT:

- IS NOT CURRENTLY EXECUTING IN THE MSP, BUT IS READY
- HAS ALL THE REQUIREMENTS IT NEEDS IN ORDER TO EXECUTE
- HAS NOT BEEN SELECTED TO EXECUTE BY THE TASK DISPATCHER

ACTIVE PROGRAMS QUEUE	PROCESSING PRIORITY	DISPATCHING STATUS	POSITION ON THE ACTIVE PROGRAM LIST QUEUE	POSITION ON THE READY TASK QUEUE
PROGRAM 1	HIGH = 196	WAITING	1	-
PROGRAM 2	LOW = 68	READY	6	2
PROGRAM 3	NORMAL = 121	WAITING	5	-
PROGRAM 4	HIGH = 196	WAITING	2	-
PROGRAM 5	Normal = 129	READY	4	1
PROGRAM 6	Normal = 133	WAITING	3	-

7.0 MAIN STORAGE CONTENTS MANAGEMENT

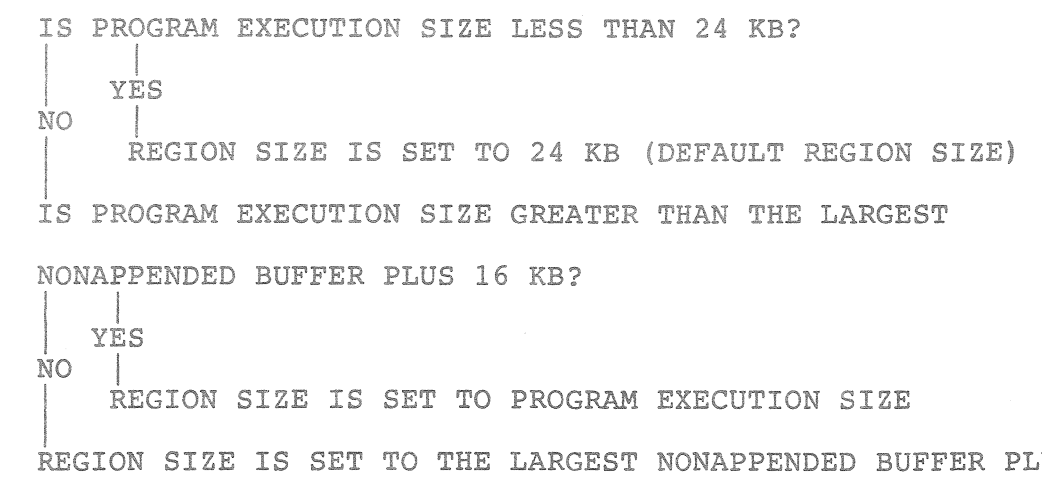
7.1 PROGRAM LOADING

BEFORE LOADING A PROGRAM, THE SYSTEM:

A. RESERVES A 64 KB AREA IN THE TASK WORK AREA (TWA) ON DISK (I.E., THE TASK SWAP AREA)

B. REQUIRES A SPECIFIC AMOUNT OF MAIN STORAGE SPACE (I.E., A REGION) FOR PROGRAM EXECUTION. A REGION SIZE FOR A PROGRAM IS DETERMINED BY EITHER:

. DEFAULT LOGIC



NOTE: AN ADDITIONAL 16 KB IS ADDED TO THE TASK WORK SPACE BY DISK DATA MANAGEMENT WHEN A BUFFER IS NONAPPENDED.

. USER SPECIFIED EITHER:

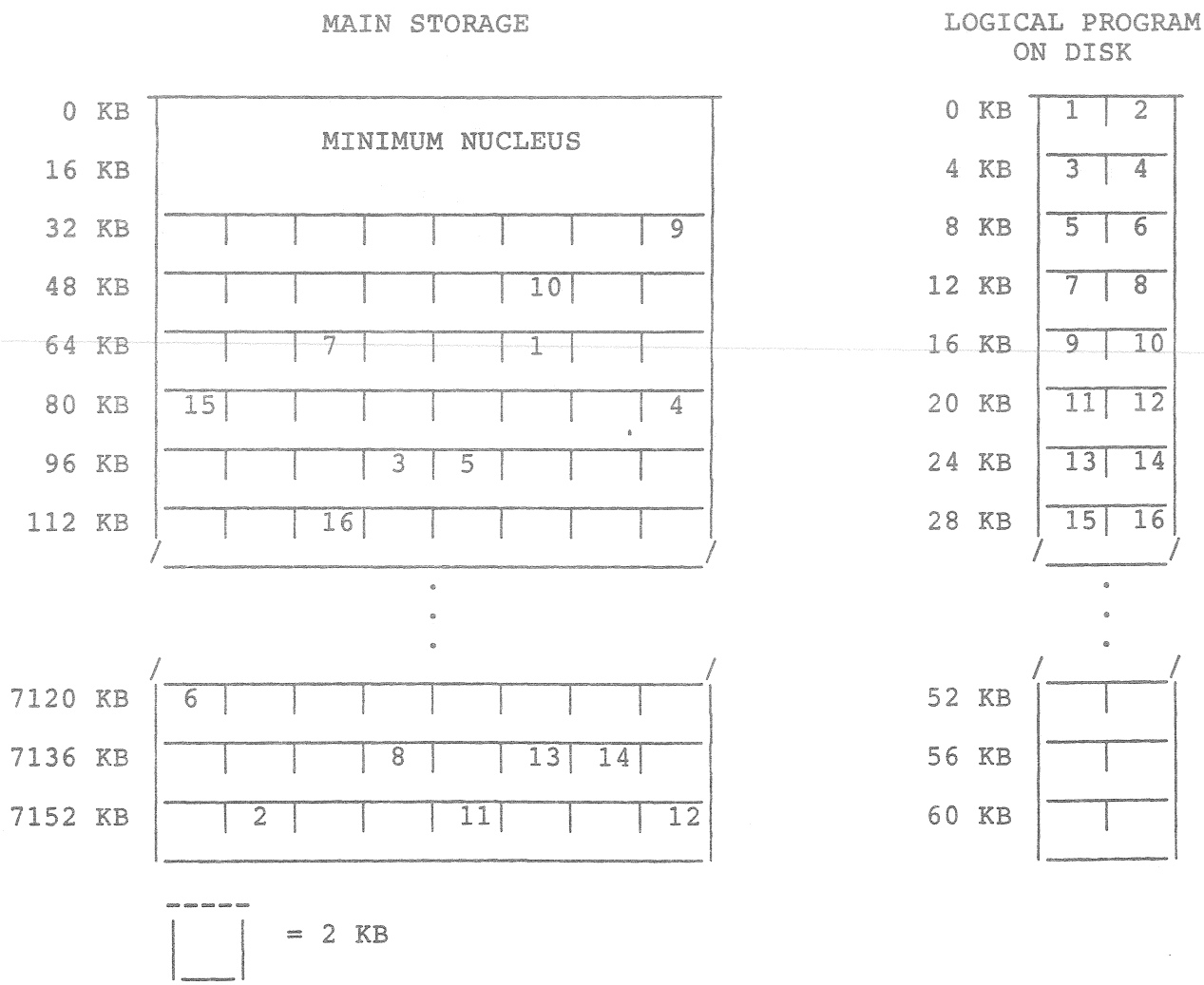
- // REGION OCL STATEMENT
- SET PROCEDURE

ERROR CONDITIONS:

THE PROGRAM TERMINATES DUE TO LACK OF SUFFICIENT USER MAIN STORAGE WHEN PROGRAM EXECUTION SIZE IS:

- GREATER THAN AVAILABLE USER MAIN STORAGE
- LESS THAN AVAILABLE USER MAIN STORAGE, BUT A NONAPPENDED BUFFER PLUS 16 KB IS LARGER THAN AVAILABLE USER MAIN STORAGE

7.1.1 PROGRAM LAYOUT IN MAIN STORAGE - EXAMPLE



. PROGRAMS ARE SCATTERED IN MAIN STORAGE IN 2 KB PAGES

. HARDWARE ADDRESS TRANSLATION PROVIDES LOGICAL/REAL MAPPING:

  - ELIMINATES STORAGE FRAGMENTATION

7.2 PROCESSING CONSIDERATIONS

SYSTEM/36 LETS YOU ASSUME AN IMPORTANT ROLE IN THE MANAGEMENT AND SCHEDULING OF YOUR JOBS. YOU CAN AFFECT:

- THE ORDER IN WHICH YOUR JOBS ARE PRESENTED TO BE EXECUTED BY THE USE OF DIFFERENT JOB QUEUE PRIORITIES IN THE INPUT JOB QUEUE.
- THE SWAPPING AND MAIN STORAGE PROCESSOR UTILIZATION OF YOUR PROGRAMS BY THE USE OF DIFFERENT EXECUTION PRIORITIES.

THREE OTHER FACTORS TO CONSIDER IN OBSERVING THE BEHAVIOR OF YOUR JOBS ARE:

- TIME SLICE
- EXECUTION PRIORITIES
- SWAPPING

7.3 SYSTEM TIME SLICE

SYSTEM/36 TIME SLICE INCLUDES TWO ELEMENTS:

- A RESOURCE UTILIZATION SCHEME:
  - R-TIME (400 MS OF TIME FOR OVERALL SYSTEM RESOURCE USAGE)
  - MSP-TIME (100 MS OF TIME FOR MAIN STORAGE PROCESSOR USAGE)
- A SYSTEM TIMER
  - MANAGES RESOURCE UTILIZATION
  - 200 MS

THE ACTUAL EXECUTION TIME FOR A TASK DEPENDS ON WHEN THE TASK:

- GOES TO A "LONG WAIT" STATUS
- EXCEEDS ITS R-TIME

7.3.1 SYSTEM TIMER

THE SYSTEM HAS A 200 MS INTERNAL TIMER INTERVAL CALLED THE "SYSTEM TIME-OUT COUNT". WHEN IT EXPIRES, IT:

- RELEASES MAIN STORAGE OF A SHARED SYSTEM PROGRAM THAT'S IN A "LONG WAIT" STATUS (I.E., WHEN THE CURRENT DEMAND COUNT REACHES ZERO AND THERE ARE NO ACTIVE USERS OF THE PROGRAM)
- DOWNGRADES THE PROGRAM'S PRIORITY WHEN R-TIME IS EXCEEDED, EXCLUDING THE FIRST TIME R-TIME IS EXCEEDED
- DOWNGRADES THE DEMAND OF SYSTEM PROGRAMS AND SYSTEM WORK SPACE (SWS)
- AUTOMATICALLY RESETS TO 200 MS AND, UPON EXPIRING, REPEATS THE ABOVE STEPS
- MAKES MAIN STORAGE AVAILABLE

7.3.2 RESOURCE UTILIZATION SCHEME

7.3.2.1 RESOURCE TIMER (R-TIME)

THIS IS A SCHEME EMPLOYED BY THE SYSTEM TO CHARGE A USER TASK FOR USING A SYSTEM RESOURCE. THE NUMBER OF MILLISECONDS (MS) DEDUCTED FROM THE 400 MS ALLOTTED, DEPENDS ON THE RESOURCE USED.

. AN APPROXIMATE TIME FOR MSP AND CSP UTILIZATION.

. A THEORETICAL TIME FOR PHYSICAL I/O OPERATIONS.

FORMULA: R-TIME = S - (U x F) - (MSP-TIME) - (CSP-TIME)

S = 400 MS (BASE VALUE)
U = 8.192 MS (TIMER UNIT)
F = I/O TIMER UNIT FACTOR

DEVICES	I/O TIME UNIT FACTOR VALUE
1255 (MICR)	= 2 PER PHYSICAL OPERATION
DISK	= 3 PER PHYSICAL OPERATION
PRINTER (SYSTEM NON-SPOOL)	= 11 PER PHYSICAL OPERATION
DISKETTE	= 49 PER PHYSICAL OPERATION
TAPE	= 50 PER PHYSICAL OPERATION
DISKETTE DATA COMPRESSION	= 51 PER MULTIPLE PHYSICAL OPERATIONS

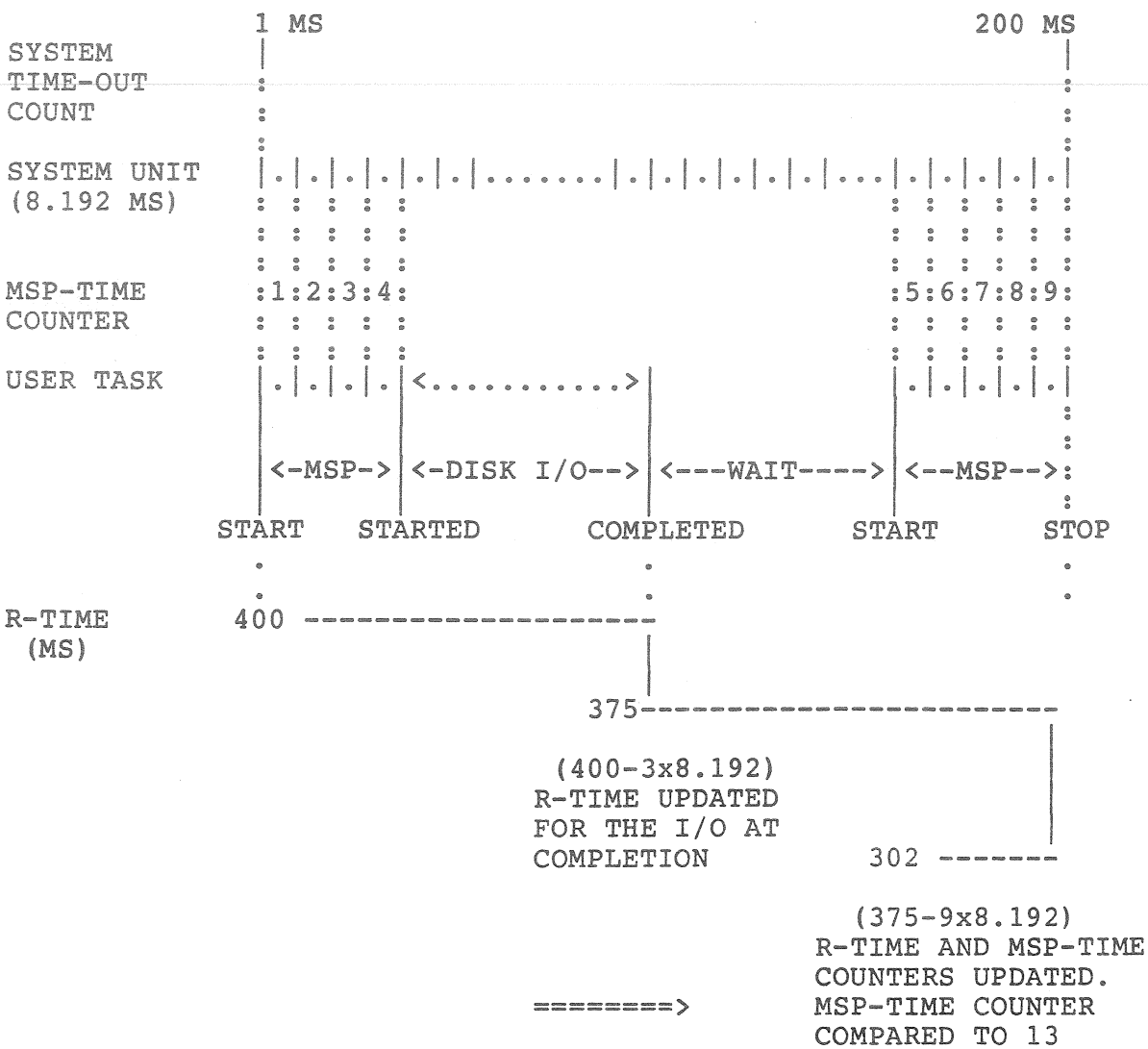
MSP-TIME = MAIN STORAGE PROCESSOR TIME USED SINCE THE LAST TIME OUT

CSP-TIME = CONTROL STORAGE PROCESSOR TIME (FOR BASIC AND FORTRAN EMULATION)

WHEN R-TIME IS EXCEEDED AND THERE IS A TASK OF EQUAL OR HIGHER PRIORITY READY TO EXECUTE, THE TASK COULD BE FORCED OUT (SWAPPED-OUT) OF MAIN STORAGE.

7.3.2.1.1 RESOURCE UTILIZATION (DISK I/O AND MSP) - EXAMPLE

A PROGRAM IS EXECUTED BY THE MSP FOR A PERIOD OF 4 SYSTEM UNITS BEFORE STARTING AN I/O DISK OPERATION. WHEN THIS OPERATION IS COMPLETED, IT HAS TO WAIT A LITTLE (BECAUSE THE MSP IS CURRENTLY USED BY ANOTHER TASK) BEFORE BEING EXECUTED AGAIN BY THE MSP FOR 5 SYSTEM UNITS.



7.3.2.2 MSP TIME-OUT TIMER (MSP-TIME)

THIS IS A SCHEME EMPLOYED BY THE SYSTEM TO LIMIT THE INFLUENCE OF A MSP PROCESSOR-BOUND PROGRAM, IF ONE PROGRAM HAS BEEN EXECUTING FOR MORE THAN 100 MS.

. APPROXIMATE TIME FOR MSP UTILIZATION:

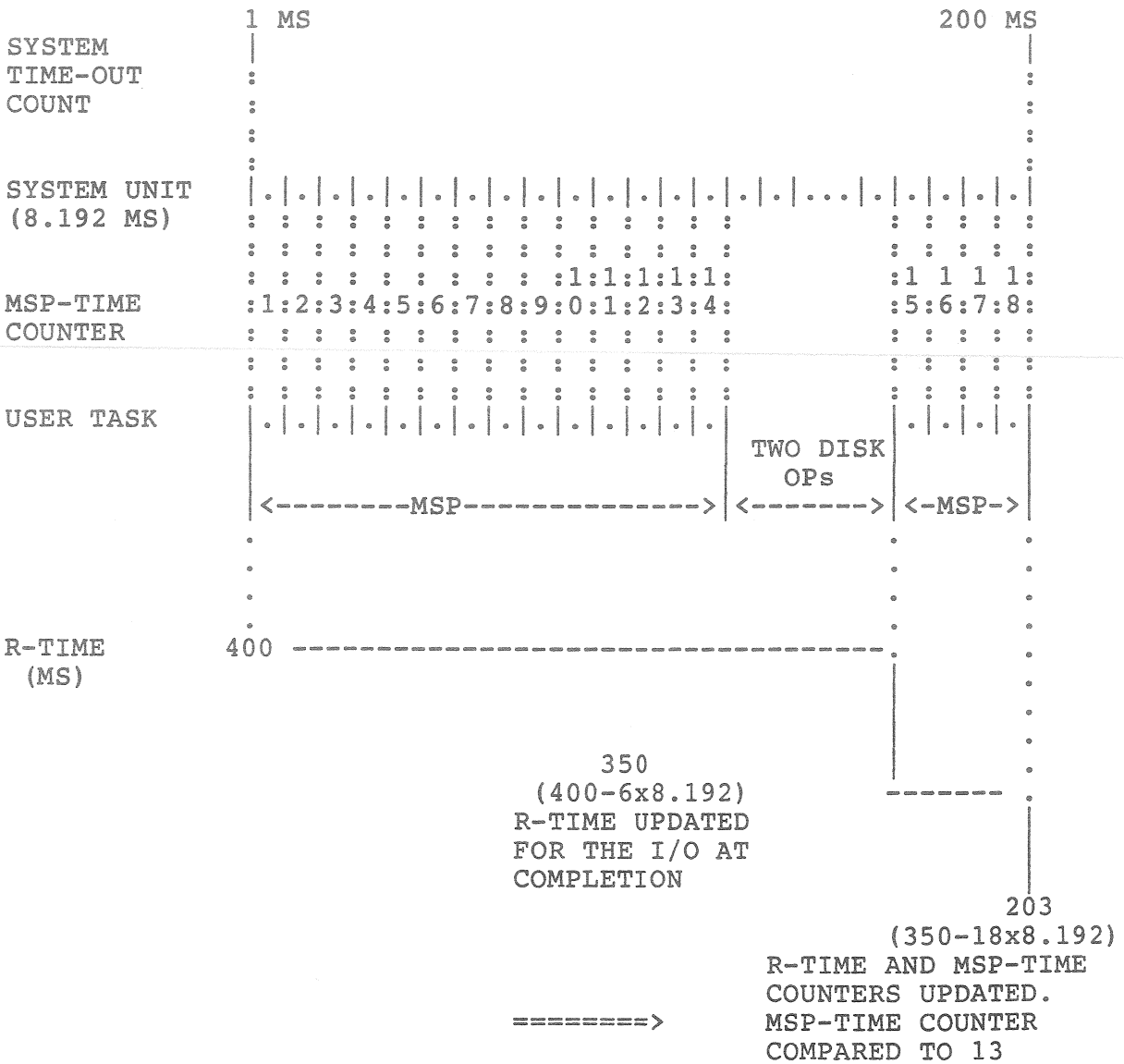
FORMULA: MSP-TIME = C x U

C = MSP-TIME COUNTER

U = 8.192 MS (TIMER UNIT)

. CHECKED EVERY 200 MS (I.E., WHEN THE SYSTEM TIMER EXPIRES)
. MSP-TIME EXPIRES WHEN THE "MSP-TIME COUNTER" REACHES A VALUE OF 13 (I.E., 13 x 8.192 = 106 MS) OR HIGHER.
. WHEN MSP-TIME IS EXCEEDED AND THERE IS A TASK OF EQUAL OR HIGHER PRIORITY READY TO EXECUTE, THE TASK IS KICKED OUT OF THE MSP.

7.3.2.2.1 RESOURCE UTILIZATION (MSP-TIME) - EXAMPLE



7.4 TASK EXECUTION PRIORITY

TWO PRIORITY TYPES:

. FIXED

  - SYSTEM (NOT AVAILABLE TO USERS)

  - USER SPECIFIED:

    -- HIGH

    -- MEDIUM

    -- LOW

. VARIABLE

  - NORMAL (DEFAULT PRIORITY)

AUTOMATIC PRIORITY ADJUSTMENT IS PROVIDED.

7.4.1 EXECUTION PRIORITY VALUES

THE FOLLOWING TABLES SHOW CATEGORIES AND VALUES (OR LEVELS) ASSIGNED TO:

. SYSTEM PROGRAMS:

- SEVEN FIXED CATEGORIES
- RANGING FROM A LOW OF 240 TO A HIGH OF 255.

USAGE	FIXED		
	DEFINED		VALUE
SYSTEM	SYSTEM	1	255
		2	
		3	
		4	
		5	
		6	
		7	240

. USER PROGRAMS:

- THREE FIXED CATEGORIES
- ONE VARIABLE CATEGORY

USAGE	FIXED			VARIABLE		
	DEFINED	VALUE	SWITCH	DEFAULT	VALUE	SWITCH
USER	HIGH	239 192	YES or ON		239	
	MEDIUM	191 128		NORMAL		NO or OFF
	LOW	127 64			69	

7.4.1.1 USER EXECUTION PRIORITIES

PRIORITY IS THE RELATIVE RANKING OF ITEMS. FOR EXAMPLE, A JOB WITH HIGH PROCESSING PRIORITY SHOULD RUN FASTER THAN A JOB WITH MEDIUM OR LOW PRIODITY.

. YOU CAN SPECIFY FOUR DIFFERENT EXECUTION PRIORITIES FOR YOUR JOB OR JOB STEPS.

. THESE EXECUTION PRIORITIES AFFECT THE SWAPPING AND THE WAY YOUR PROGRAM GAINS CONTROL OF THE MAIN STORAGE PROCESSOR FROM THE DISPATCHER.

IF YOU DO NOT SPECIFY AN EXECUTION PRIORITY FOR YOUR JOB, THE SYSTEM ASSIGNS YOUR JOB A NORMAL PRIORITY.

TO SPECIFY THE EXECUTION PRIORITY OF YOUR JOB(S), YOU CAN USE THE FOLLOWING:

. PRTY COMMAND:

  - HIGH OR ON
  - MEDIUM
  - NORMAL OR OFF
  - LOW

. // ATTR OCL STATEMENT:

  - HIGH OR YES
  - MEDIUM
  - NO (EQUIVALENT TO NORMAL)
  - LOW

7.4.1.2 USER EXECUTION PRIORITY FORMULA

FORMULA: PRIORITY = (4 x N) + V

4 = INCREMENT/DECREMENT FACTOR

N = NUMBER OF ATTACHED WORK STATIONS OR SSP-ICF SESSIONS

V = BASE PRIORITY VALUE

FIXED

- HIGH OR SWITCH ON = 192
- MEDIUM = 128
- LOW = 64

VARIABLE

- NORMAL OR SWITCH OFF = 129

AUTOMATIC PRIORITY ADJUSTMENT:

. THE PRIORITY DECREASES BY 4 (DECREMENT FACTOR):
  - WHEN R-TIME IS EXCEEDED
  - BY A "LONG WAIT" STATUS (TO INCREASE SWAP ELIGIBILITY)
  - WHEN A WORK STATION OR SSP-ICF SESSION IS RELEASED
. THE PRIORITY INCREASES BY 4 (INCREMENT FACTOR) WHEN A WORK STATION OR SSP-ICF SESSION IS ATTACHED
. THE PRIORITY IS RESET TO ORIGINAL PRIORITY WHEN THE LAST SUBTRACTION BRINGS THE PRIORITY EQUAL TO OR BELOW THE BASE VALUE OF ITS GROUP ON A "LONG WAIT" STATUS

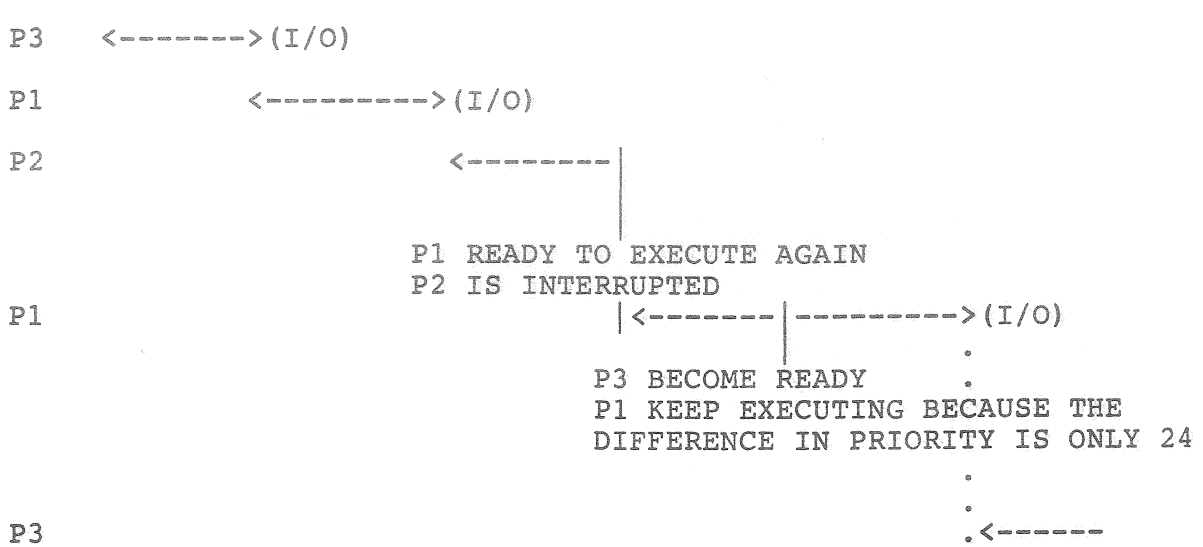
- Fixed Pty Only never goes below base value.

7.4.1.2.1 USER PRIORITY (VARIABLE) AND DISK I/O - EXAMPLE

THREE TASKS READY TO EXECUTE:

ACTIVE PROGRAMS QUEUE	PROCESSING PRIORITY	DISPATCHING STATUS	POSITION ON THE ACTIVE PROGRAM LIST QUEUE	POSITION ON THE READY TASK QUEUE
PROGRAM 1	NORMAL = 109	READY	2	2
PROGRAM 2	NORMAL = 69	READY	3	3
PROGRAM 3	NORMAL = 133	READY	1	1

EXECUTION SEQUENCE:

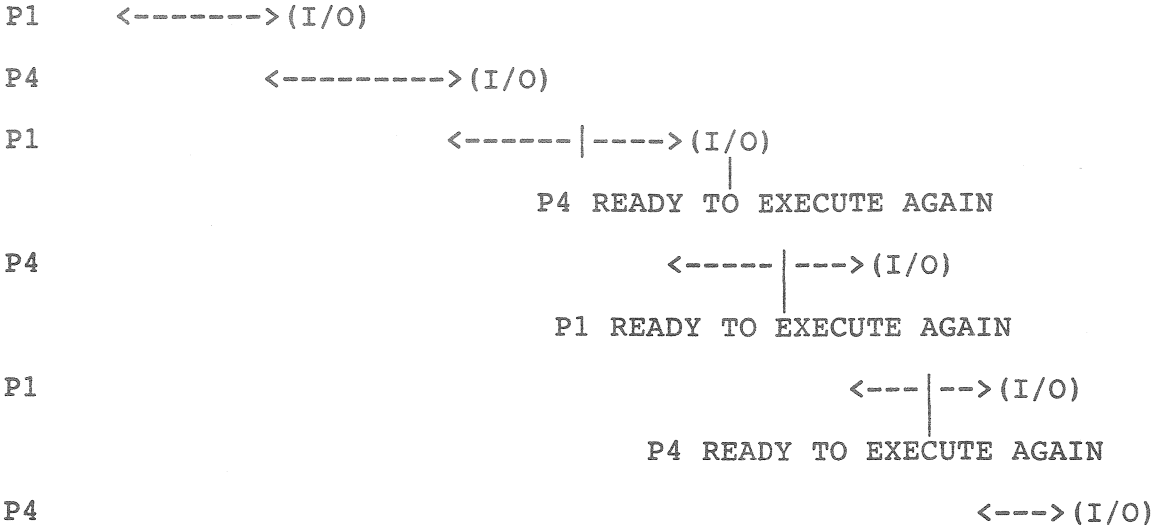


7.4.1.2.2 USER PRIORITY (FIXED AND VARIABLE) AND DISK I/O - EXAMPLE

FIVE TASKS READY TO EXECUTE:

ACTIVE PROGRAMS QUEUE	PROCESSING PRIORITY	DISPATCHING STATUS	POSITION ON THE ACTIVE PROGRAM LIST QUEUE	POSITION ON THE READY TASK QUEUE
PROGRAM 1	HIGH = 196	READY	1	1
PROGRAM 2	NORMAL = 133	WAITING	3	5
PROGRAM 3	NORMAL = 69	READY	5	3
PROGRAM 4	HIGH = 196	READY	2	2
PROGRAM 5	NORMAL = 109	WAITING	4	4

EXECUTION SEQUENCE:



NOTES:

TASKS P1 AND P4 BEING 32 PRIORITY VALUES HIGHER CAUSES:

. TASKS P2, P5, AND P3 TO SELDOM EXECUTE
. TASKS P2, P5, AND P3 TO BE INTERRUPTIBLE DURING EXECUTION:
  - TASKS P1, P4, P2, AND P5 WILL INTERRUPT TASK P3 DURING EXECUTION
  - TASKS P1, P4, AND P2 WILL INTERRUPT TASK P5 DURING EXECUTION
  - TASKS P1 AND P4 WILL INTERRUPT TASK P2 DURING EXECUTION

7.4.2 TASK PRIORITY FREQUENCIES - EXAMPLE

EXAMPLE - PRIORITY FREQUENCIES DURING EXECUTION

PRIORITY	FREQUENCY	PERCENT	
64	27	0.034	
69	7557	9.515	BATCH JOBS
73	32	0.040	
77	37	0.047	
81	114	0.144	
85	245	0.308	
89	483	0.608	
93	789	0.993	
97	1062	1.337	
101	1073	1.351	
105	1155	1.454	
109	1372	1.727	
113	1583	1.993	
117	1932	2.433	
121	2524	3.178	
125	4498	5.663	
128	128	0.161	
129	18670	23.597	THE LARGEST
133	16877	21.249	PERCENTAGES
137	3453	4.348	
141	1388	1.748	
145	922	1.161	
149	177	0.223	
192	6	0.008	
196	38	0.048	
240	1	0.001	
250	199	0.251	
252	198	0.249	
254	12875	16.210	DISK LOCKS

7.5 SWAPPING FUNCTION

. PROVIDES OPERATOR/PROGRAMMER INDEPENDENCE FROM:

  - STORAGE
  - DISK TYPE/SIZE
  - OTHER CONCURRENT OPERATIONS

. UTILIZES ALL STORAGE AVAILABLE IN ALL CONFIGURATIONS

. SWAPS

  - A MINIMUM OF 24 KB OR THE ENTIRE PROGRAM OR WORK SPACE
  - PROGRAMS (SYSTEM OR USER)
  - SYSTEM WORK SPACE
  - TASK WORK SPACE

. ALLOWS FINE TUNING THROUGH THE PRIORITY PARAMETER ON COMMANDS AND OCL STATEMENTS

7.5.1 SWAPPING CRITERIA

SWAP-IN:

. PRIORITY SELECTION

SWAP-OUT:

. PREEMPTIVE DISPATCHING BY PRIORITY GROUP
  - FIXED AND VARIABLE
    A) LONG WAIT
    B) TASK PRIORITY
    C) PROGRAM DEMAND COUNT
  - VARIABLE
    -- RESOURCE UTILIZATION TIME INTERVALS

THRASHING PREVENTION TECHNIQUES

. ATTEMPT TO GIVE ALL TASKS A GUARANTEED FIXED TIME INTERVAL AFTER SWAP-IN, EXCEPT WHEN:
  - THE TASK GOES TO "LONG WAIT" STATUS
  - THE TASK REQUIRES MORE MAIN STORAGE TO EXECUTE
. REDUCE SWAPPING OF EQUAL PRIORITY TASKS

7.5.1.1 SWAP IN

THE SWAP-IN PROCESS STARTS:

. WHENEVER STORAGE BECOMES AVAILABLE (END-OF-JOB, FOR EXAMPLE)

. AT EACH CYCLE OF THE SYSTEM TIMEOUT COUNT

. WHEN STORAGE IS NEEDED

AFTER THE PRIORITY OF EACH TASK HAS BEEN UPDATED, THE CONTROL STORAGE SUPERVISOR LOOKS FOR A CANDIDATE TO SWAP IN. THE CONTROL STORAGE SUPERVISOR SEARCHES:

STEP 1. THE ACTIVE TASKS LIST QUEUE, FROM TOP TO BOTTOM (THAT IS, FROM HIGH TO LOW PRIORITY)

STEP 2. FOR A TASK READY TO EXECUTE, BUT CURRENTLY NOT COMPLETELY IN STORAGE.

IF A TASK IS SELECTED TO HAVE SOME OF ITS COMPONENTS SWAPPED IN, IT IS GUARANTEED TO RECEIVE AT LEAST ONE FULL R-TIME (400 TO 599 MS, 500 MS BEING THE AVERAGE OF TIME), UNLESS:

. IT GOES INTO A LONG WAIT, OR

. MORE STORAGE IS REQUESTED BY THAT TASK.

7.5.1.2 SWAP OUT/RELEASE OF MEMORY

IN ORDER TO SELECT THE CANDIDATE TO BE EJECTED, THE SYSTEM USES THE STORAGE OWNER LIST, ALREADY MENTIONED. THIS LIST IS IN TWO PARTS:

PART I - LONG WAIT:

CONTAINS, FROM THE OLDEST TO THE MOST RECENT WAIT, AREAS IN A LONG WAIT (SYSTEM PROGRAMS AND AREAS WITH DEMAND COUNT = 0, OR USER PROGRAMS WITH GET OR ACCEPT FROM DISPLAY PENDING)

PART II - TASK PRIORITY:

IS FOLLOWED BY THE LIST OF ALL PROGRAMS OR WORK AREAS NOT INCLUDED IN THE FIRST PART OF THE LIST, WITH THE LEAST RECENTLY SWAPPED IN AT THE TOP AND THE MOST RECENTLY SWAPPED IN AT THE BOTTOM

THE SEARCH IS BASED ON THE PRIORITY OF THE TASK BEING SWAPPED IN AND CAN BE UP TO FOUR STEPS.

A TASK TEMPORARILY UNSWAPPABLE (BECAUSE OF AN OUTSTANDING I/O OPERATION OR BECAUSE IT IS CURRENTLY RUN BY THE MSP, FOR EXAMPLE) CANNOT BE SELECTED.

IF THE TASK IS JUST SWAPPED IN, IT IS NOT SWAP ELIGIBLE.

THE FOUR STEPS ARE:

STEP 1 - LEVEL 1 (LONG WAITS)

SEARCHES FOR PROGRAMS AT A LONG WAIT; FROM THE OLDEST TO THE MOST RECENT WAIT. THE FIRST APPROPRIATE AREA IS TAKEN. IF NONE IE FOUND, STEP 2 IS PERFORMED.

STEP 2 - LEVEL 2 (TASK PRIORITY)

SEARCHES FOR A PROGRAM TO SWAP-OUT THAT IS MORE THAN 32 PRIORITY POINTS LOWER THAN THE PROGRAM BEING SWAPPED-IN. BECAUSE OF THE WAY THE STORAGE OWNER LIST IS ORGANIZED, THE AREA SELECTED WILL ALSO BE THE ONE WHICH HAS BEEN SWAPPED IN FOR THE LONGEST. IF NONE IS FOUND, STEP 3 IS PERFORMED.

STEP 3 - LEVEL 3 (TASK PRIORITY)

SEARCHES FOR A PROGRAM TO SWAP-OUT THAT IS 32 PRIORITY POINTS LOWER THAN THE PROGRAM BEING SWAPPED-IN. IF NONE IS FOUND, STEP 4 IS PERFORMED.

STEP 4 - LEVEL 4 (TASK PRIORITY)

THE PROGRAM BEING SWAPPED-IN IS LOWER IN PRIORITY THAN THE PROGRAM BEING SWAPPED-OUT, THE SWAP WILL OCCUR. (IF THE SWAP DOES NOT OCCUR, A "DEADLOCK CONDITION" IS POSSIBLE.)

AFTER A STORAGE AREA HAS BEEN SELECTED FOR SWAP OUT, IT IS PUT INTO THE DISK TASK WORK AREA (TWA), EITHER:

- ENTIRELY, IF IT IS SMALLER OR EQUAL TO THE SPACE NEEDED OR IF THE SIZE LEFT IN MAIN STORE AFTER SWAPPING THE SPACE NEEDED IS LESS THAN 24 KB.

- PARTIALLY FOR ALL OTHER CASES. THE PORTION OF THE PROGRAM WRITTEN ON DISK IS EQUAL TO THE SPACE NEEDED WITH A MINIMUM OF 24 KB.

7.5.1.2.1 SWAP ALGORITHM

IS THE MEMORY REQUESTED EQUAL TO THE ELIGIBLE SWAP-OUT PROGRAM ?
|
| YES
NO |
| SWAP OUT THE ENTIRE PROGRAM
|
IS THE MEMORY REQUESTED GREATER THAN THE ELIGIBLE SWAP-OUT PROGRAM ?
|
| YES
NO |
| SWAP OUT THE ENTIRE PROGRAM PLUS SOMETHING ELSE
|
IS THE MEMORY REQUESTED LESS THAN THE ELIGIBLE SWAP-OUT PROGRAM AND

24 KB OF MEMORY WILL NOT BE LEFT AS A RESULT OF THE SWAP ?
|
| YES
NO |
| SWAP OUT THE ENTIRE PROGRAM
|
IS THE MEMORY REQUESTED LESS THAN THE ELIGIBLE SWAP-OUT PROGRAM AND

GREATER THAN 24 KB AND

WILL 24 KB OF MEMORY BE LEFT AS A RESULT OF THE SWAP ?
|
| YES
NO |
| SWAP OUT THE EXACT AMOUNT REQUESTED
|
SWAP OUT 24 KB -- DEFAULT SWAP/REGION SIZE

EXAMPLE:

MEMORY NEEDED	SIZE OF THE PROGRAM TO BE SWAPPED-OUT	PORTION SWAPPED OUT
12 KB	10 KB	10 KB (PLUS SOMETHING ELSE)
12 KB	30 KB	30 KB (ENTIRE PROGRAM)
18 KB	50 KB	24 KB (DEFAULT SWAP SIZE)
26 KB	60 KB	26 KB (EXACT AMOUNT)
24 KB	24 KB	24 KB (ENTIRE PROGRAM)

7.5.2 SWAPPING OCCURRENCES

SWAPPING OR RELEASING STORAGE CAN OCCUR AT TWO TIMES:

. IMMEDIATELY (UNCONDITIONALLY)
  - WHEN THE TASK IS PREEMPTED BY A TASK OF SIGNIFICANTLY HIGHER PRIORITY
  - WHEN THE PROGRAM ISSUES A WAIT SVC OR A READ OPERATION TO THE WORK STATION
  - WHEN THE PROGRAM ENDS

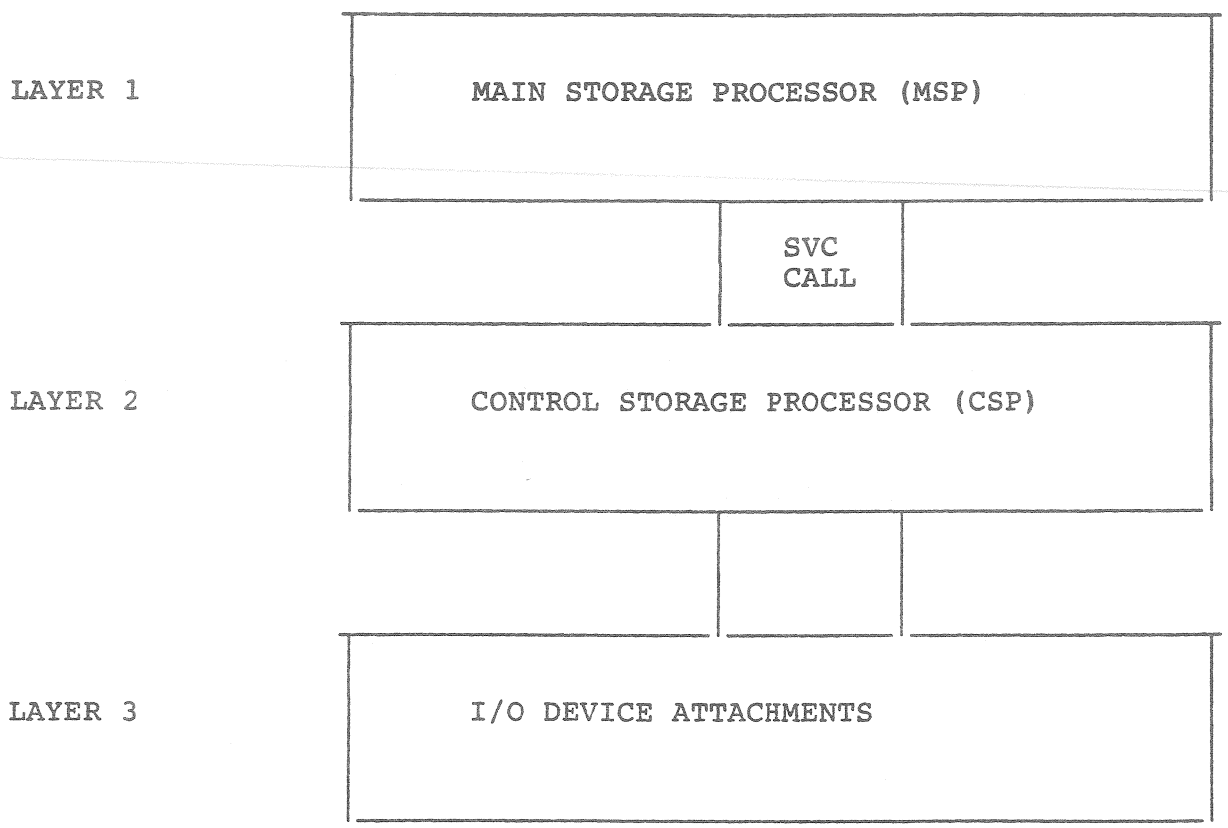
. DELAYED (CONDITIONALLY)

  AFTER THE SYSTEM TIME-OUT COUNT EXPIRES:
  - WHEN THE TASK EXCEEDS ITS R-TIME
  - WHEN THE TASK PRIORITY IS DOWNGRADED TO EQUAL TO OR BELOW THE INCOMING TASK
  - WHEN THE TASK PROGRAM DEMAND COUNT REACHES ZERO (i.e., CURRENTLY IN A LONG WAIT)

OTHER FACTORS THAT CAN INCREASE SWAPPING ARE:
  - A HEAVILY LOADED SYSTEM
  - NOT ENOUGH MAIN STORAGE SPACE
  - POOR PROGRAM DESIGN

8.0 DISTRIBUTION OF SOFTWARE FUNCTIONS AND EXECUTION

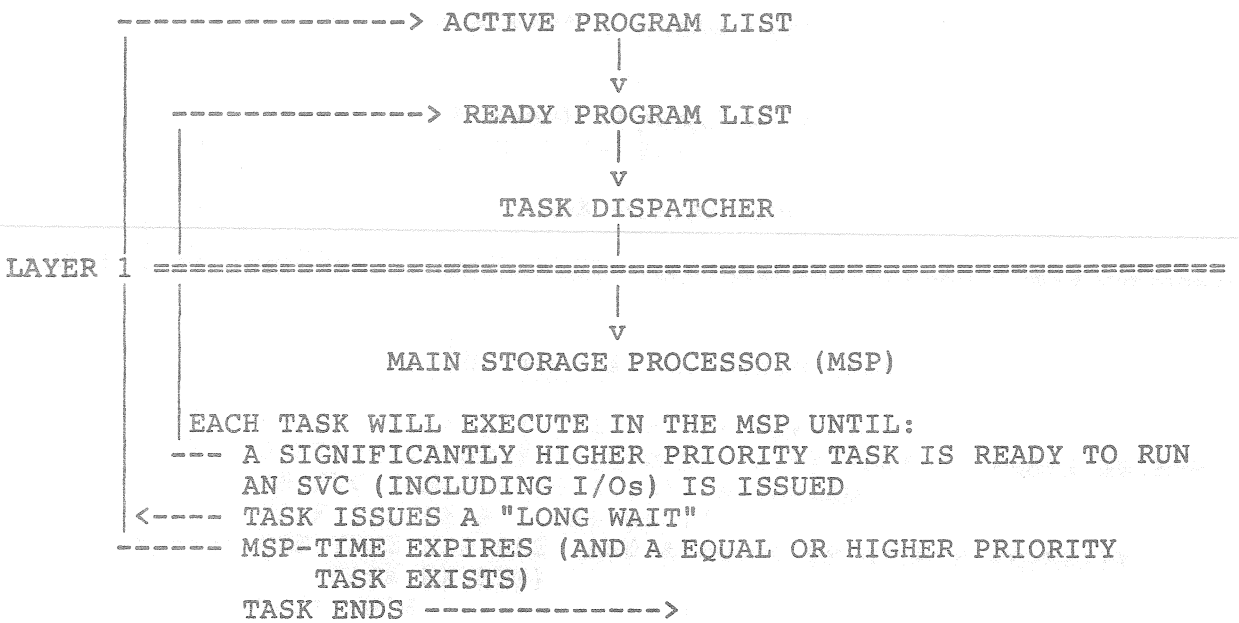
THE SYSTEM/36 SOFTWARE FUNCTIONS AND EXECUTION IS DISTRIBUTED INTO THREE LAYERS WITHIN THE CPU. THESE THREE LAYERS ARE:



FUNCTION ARE DISTRIBUTED IN THIS MANNER TO PROVIDE MORE OVERLAPPED EXECUTION AND THEREFORE MORE SYSTEM PERFORMANCE THAN WOULD AVAILABLE WITH A MULTIPROCESSOR SYSTEM. THESE MULTIPLE PROCESSORS OPERATE ALMOST INDEPENDENTLY WITH THE FUNCTION SYNCHRONIZATION OCCURRING WITHIN THE CSP.

8.1 MAIN STORAGE PROCESSOR (MSP) - LAYER 1

THE FOLLOWING SHOWS THE MAIN STORAGE PROCESSOR (LAYER 1) EXECUTION LOGIC FLOW:

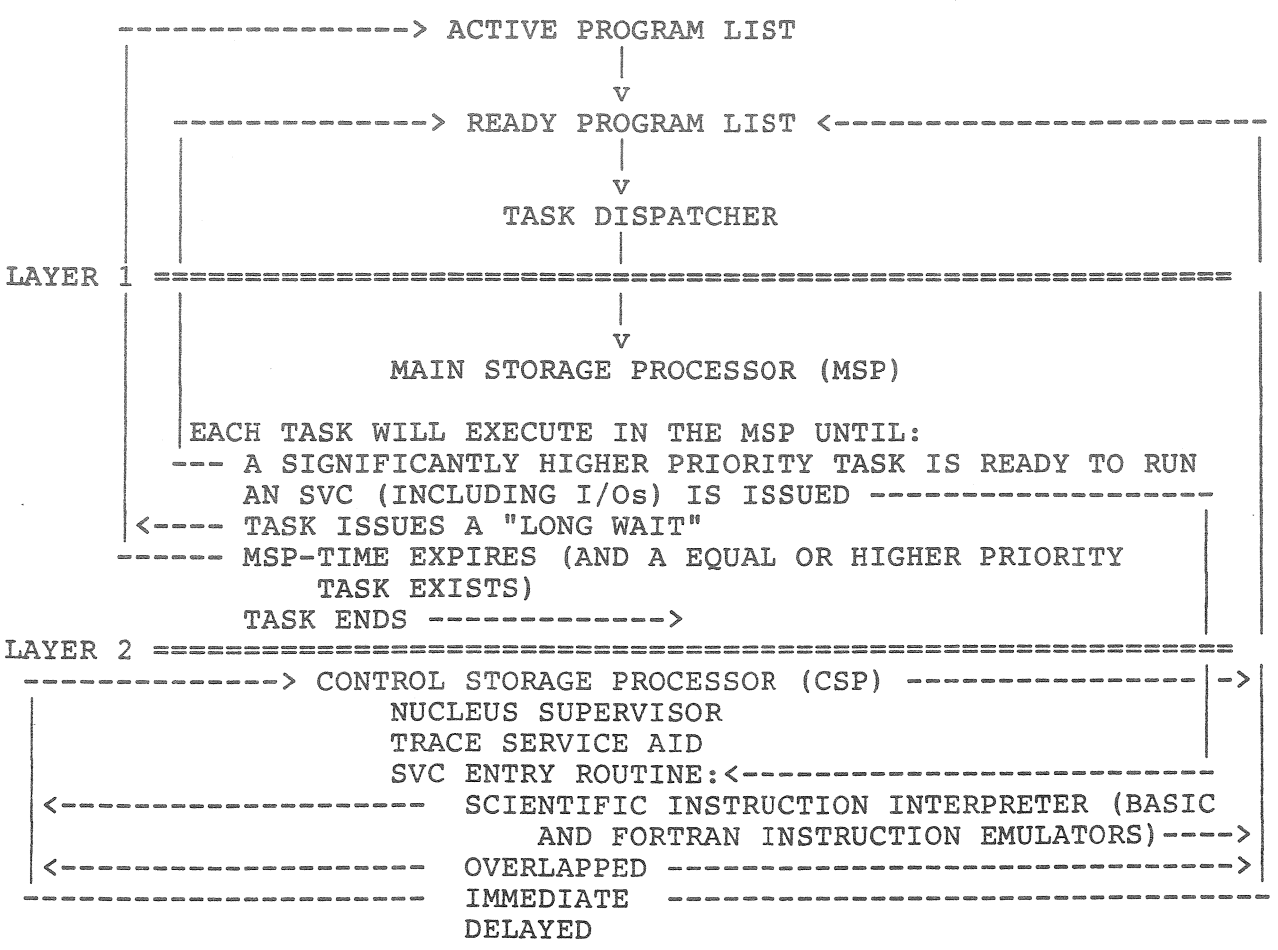


8.2 CONTROL STORAGE PROCESSOR (CSP) - LAYER 2

MSP/CSP INTERACTION:

- ONE OF THE MAJOR IMPROVEMENTS TO S/36, WHICH IMPACTS SYSTEM PERFORMANCE, IS THE WAY THE MSP AND THE CSP INTERACT.
- THE CONTROL STORAGE PROCESSOR AND THE MAIN STORAGE PROCESSOR INTERFACE THROUGH SUPERVISOR CALL (SVC) INSTRUCTIONS AND COMMON DATA AREAS.

THE FOLLOWING SHOWS THE EXECUTION INTERFACE BETWEEN THE CONTROL STORAGE PROCESSOR (LAYER 2) AND THE MAIN STORAGE PROCESSOR (LAYER 1):



8.2.1 SUPERVISORY CALLS (SVCS)

BASED ON THE SVC REQUESTED, THE CONTROL STORAGE PROCESSOR DETERMINES IF THE OPERATION IS VALID AND IMMEDIATE, OVERLAPPED, OR DELAYED:

. IMMEDIATE

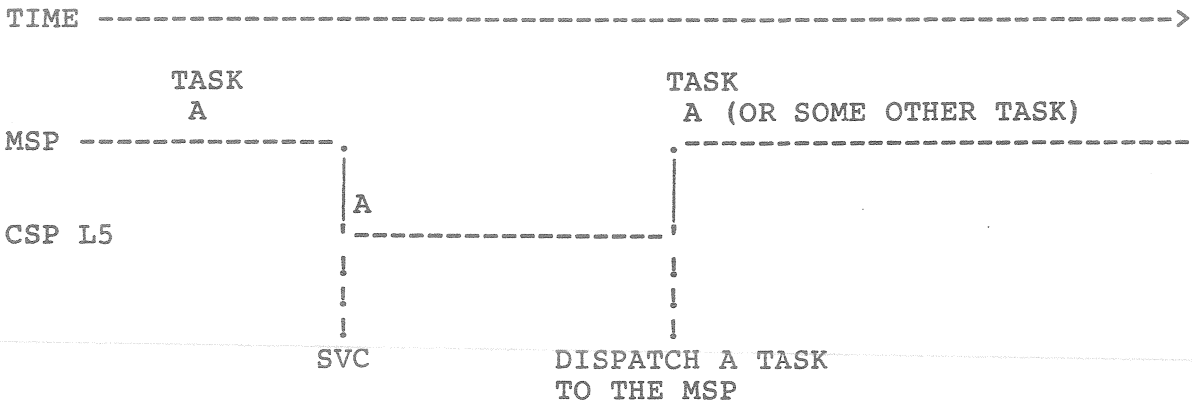
A - THE MAIN STORAGE PROCESSOR IS STOPPED TEMPORARILY.

B - AFTER THE FUNCTION IS PERFORMED, CONTROL RETURNS TO THE MAIN STORAGE PROCESSOR AND EXECUTION RESTARTS AT THE FIRST INSTRUCTION AFTER THE SVC, IF THERE IS NO OTHER TASK OF A SIGNIFICANTLY HIGHER PRIORITY READY TO EXECUTE.

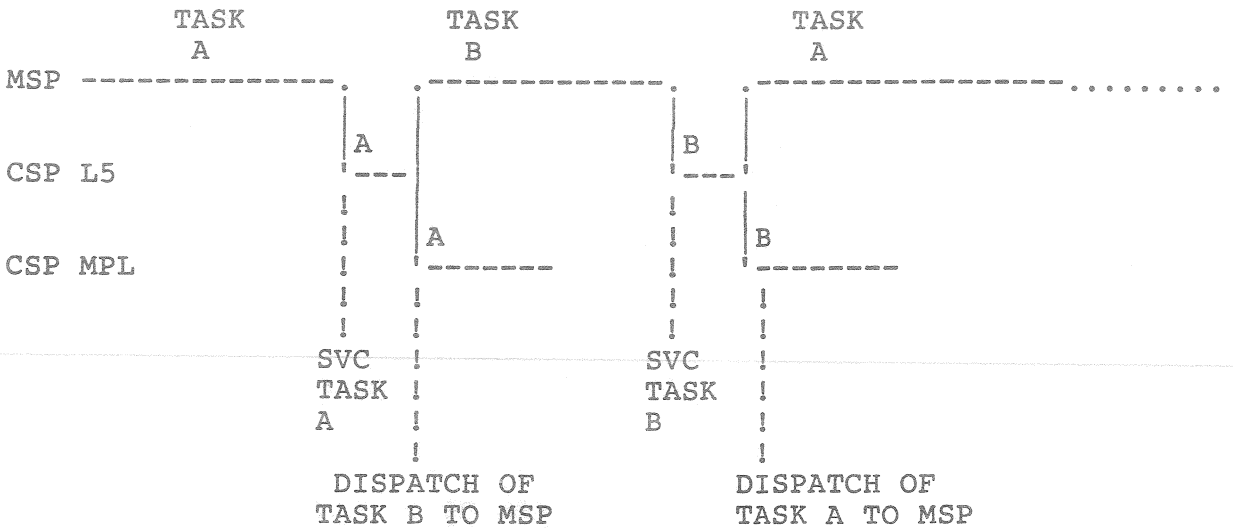
. OVERLAPPED OR DELAYED

IF THE SVC INSTRUCTION IS OVERLAPPED OR DELAYED, AN ENTRY ROUTINE IN THE CONTROL STORAGE PROCESSOR DETERMINES IF ANOTHER TASK CAN BE EXECUTED AND, IF SO, REQUESTS THE MSP HARDWARE TO LOAD THE REGISTERS OF THE MSP FOR A DIFFERENT TASK AND RESUME EXECUTION.

8.2.1.1 IMMEDIATE SVC - EXAMPLE

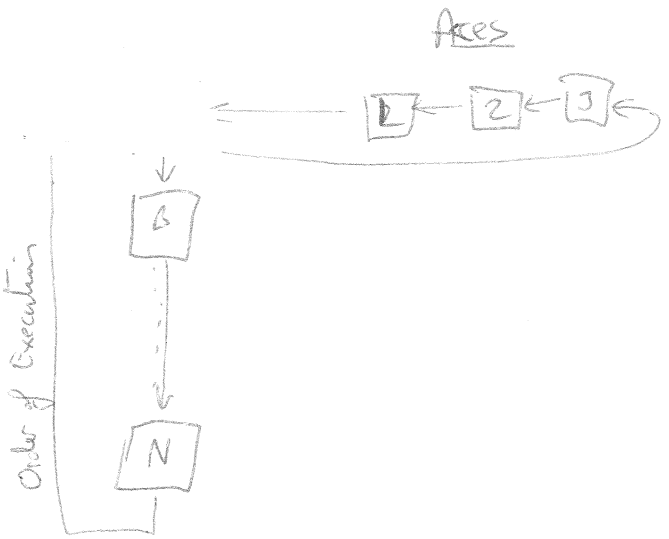


8.2.1.2 OVERLAPPED SVC - EXAMPLE



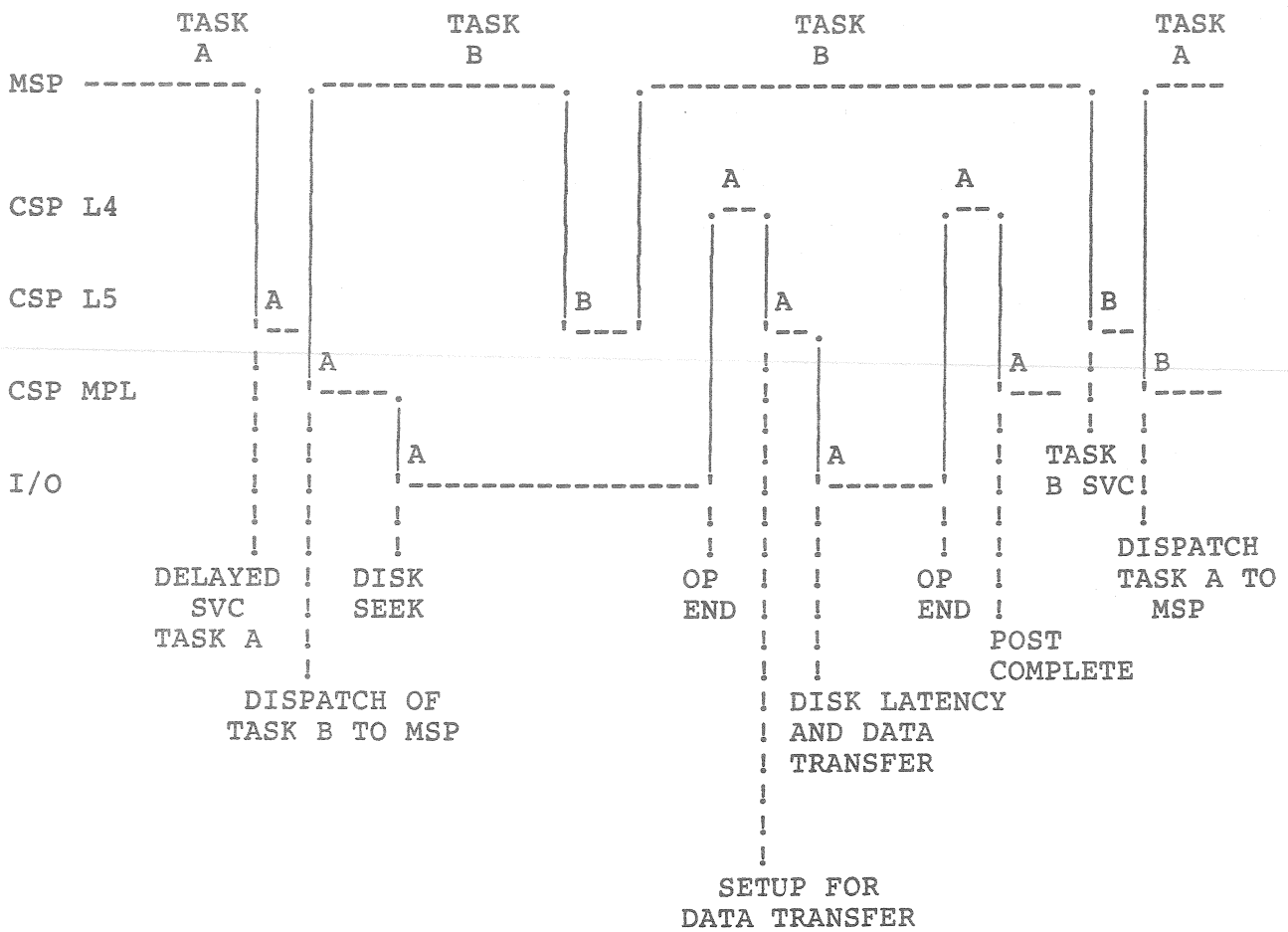
. MSP AND CSP RUNNING SIMULTANEOUSLY

. "DEAD" TIMES FOR BOTH PROCESSORS GREATLY REDUCED



Key Think vs. Service Time

8.2.1.3 DELAYED SVC (FOR DISK) - EXAMPLE



. PARALLEL PROCESSORS (MULTIPLE MICROPROCESSORS):

- MSP
- CSP
- CSP/I (I/O)

8.3 I/O DEVICE ATTACHMENTS - LAYER 3

8.3.1 I/O SUPPORT FUNCTION

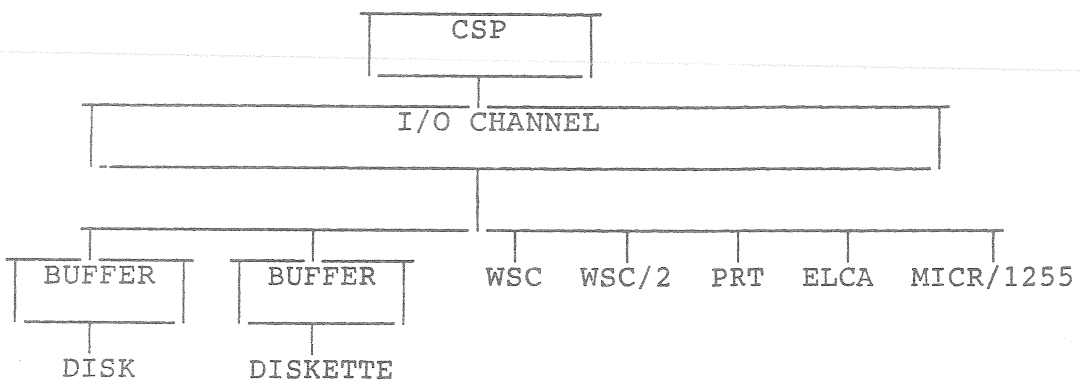
THE I/O SUPPORT FUNCTION SUPPORTS THE SYSTEM I/O DEVICES. EACH I/O DEVICE HAS ITS OWNS FUNCTIONAL MICROCODE THAT, IN CONJUNCTION WITH THE I/O DEVICE AND I/O ADAPTER, PRESENT AN I/O INTERFACE TO THE CONTROL PROCESSOR.

THE I/O SUPPORT FUNCTION PROVIDES SOME COMMON SUBROUTINES FOR ALL I/O DEVICES, BUT THE MAJORITY OF THE SUBFUNCTIONS WITHIN THE I/O SUPPORT FUNCTION ARE DEVICE-DEPENDENT.

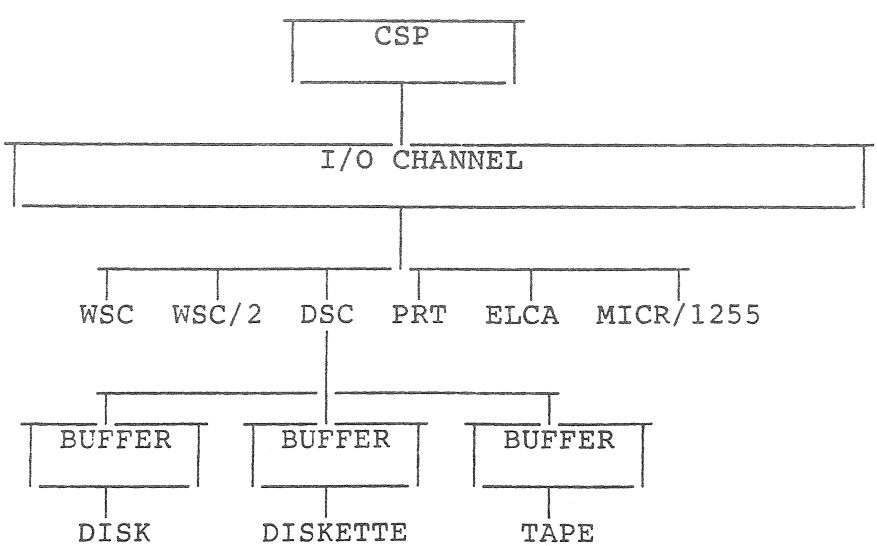
8.3.1.1 I/O CHANNEL

THE I/O PROCESSORS ARE LINKED TO THE CSP BY A "CHANNEL", WHICH IS REALLY A 2-BYTE BIDIRECTIONAL BUS, RUNNING AT 2.5 MEGABYTES BURST DATA RATE.

THE FOLLOWING ILLUSTRATES THE I/O CHANNEL WITHOUT THE DSC INSTALLED:



THE FOLLOWING ILLUSTRATES THE I/O CHANNEL WITH THE DSC INSTALLED:



8.3.2 I/O DEVICE ATTACHMENTS

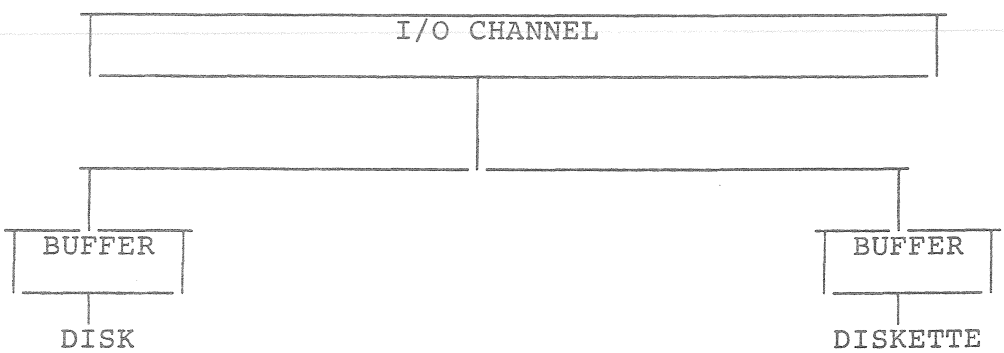
I/O DEVICES ARE ATTACHED TO THE SYSTEM I/O CHANNEL BY VARIOUS I/O DEVICE ATTACHMENTS, THESE ARE:

- DATA STORAGE ATTACHMENT (DSA)
- WORKSTATION ATTACHMENT
- MAGNETIC TAPE ATTACHMENT
- SYSTEM PRINTER ATTACHMENT
- COMMUNICATIONS ATTACHMENT
- 1225 MICR ATTACHMENT

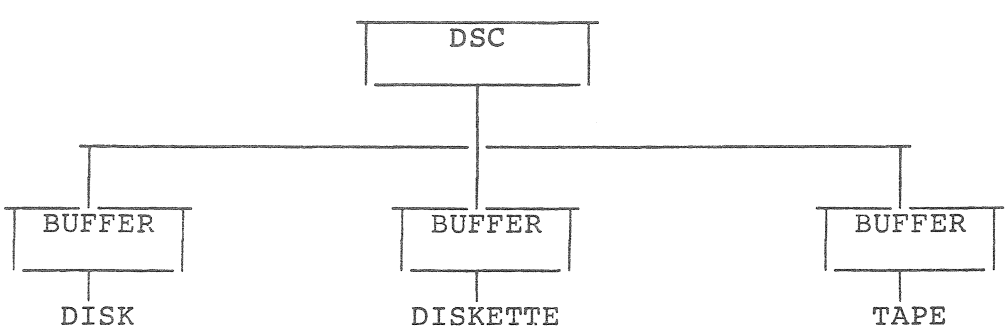
8.3.2.1 DATA STORAGE ATTACHMENT (DSA)

THE DATA STORAGE ATTACHMENT IS A PIECE OF HARDWARE TO WHICH SEVERAL DEVICES CAN BE ATTACHED.

THE DISK/DISKETTE DSA HAS EIGHT 256-BYTE BUFFERS, ORGANIZED IN TWO SETS OF FOUR. EACH SET CAN BE ALLOCATED TO DISK SPINDLES 1, 2, 3, 4, OR DISKETTE.



THE TAPE DSA HAS ONE SET OF FOUR 256-BYTE BUFFERS, AND CAN BE ALLOCATED TO EITHER ONE OF THE TAPE DRIVES.

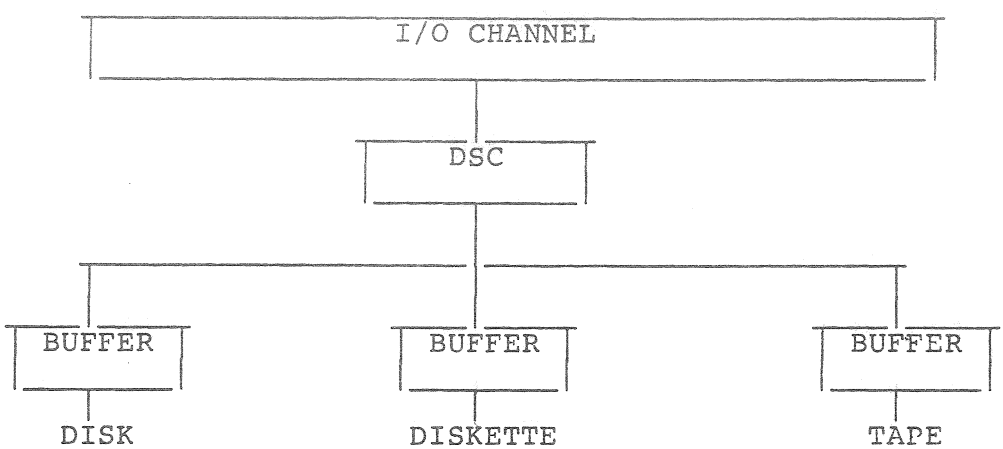


8.3.2.2 MAGNETIC TAPE ATTACHMENT

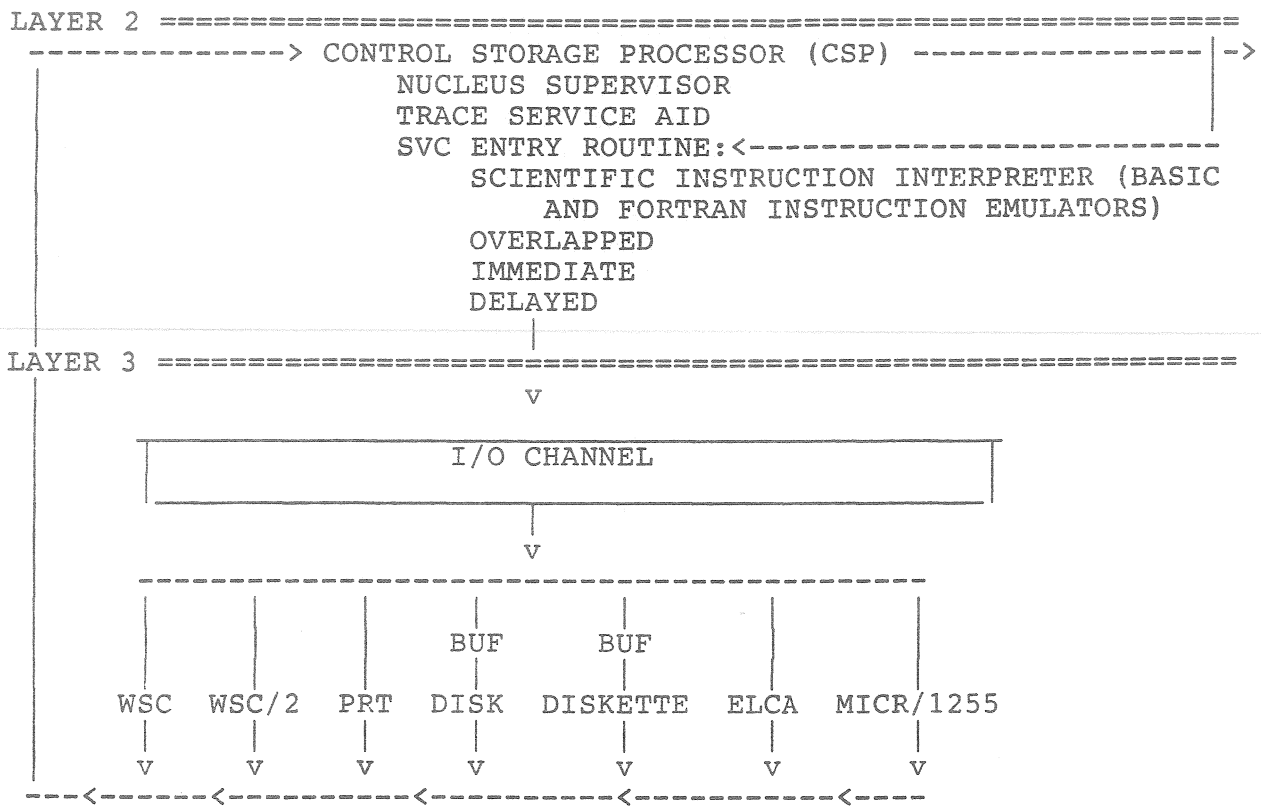
THE DATA STORAGE CONTROLLER (DSC) PROCESSES DISK, DISKETTE, AND TAPE I/O ON SYSTEMS THAT HAVE THE TAPE ATTACHMENT. IN ADDITION, DATA FILE TRANSFERS FROM DEVICE TO DEVICE ARE DONE IN THE DSC.

THE DEVICE-TO-DEVICE OPERATION APPLIES TO TRANSFER OF DATA:

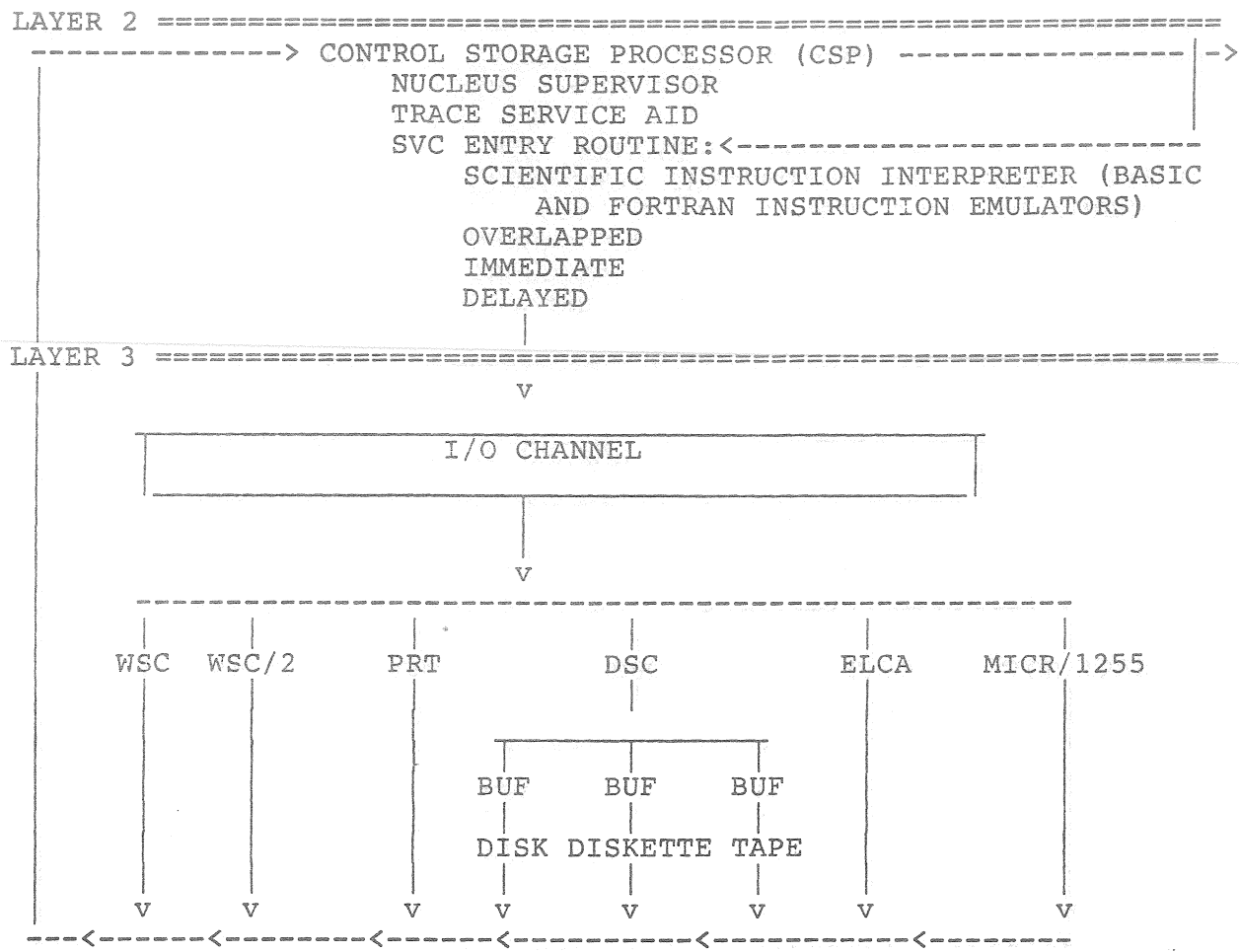
- FROM TAPE TO DISK
- FROM DISK TO TAPE
- FROM DISK TO DISKETTE
- FROM DISKETTE TO DISK
- FROM DISK TO DISK



THE FOLLOWING SHOWS THE RELATIONSHIP OF THE I/O PROCESSORS (LAYER 3 WITHOUT THE DATA STORAGE CONTROLLER - DSC) AND THE CSP (LAYER 2):

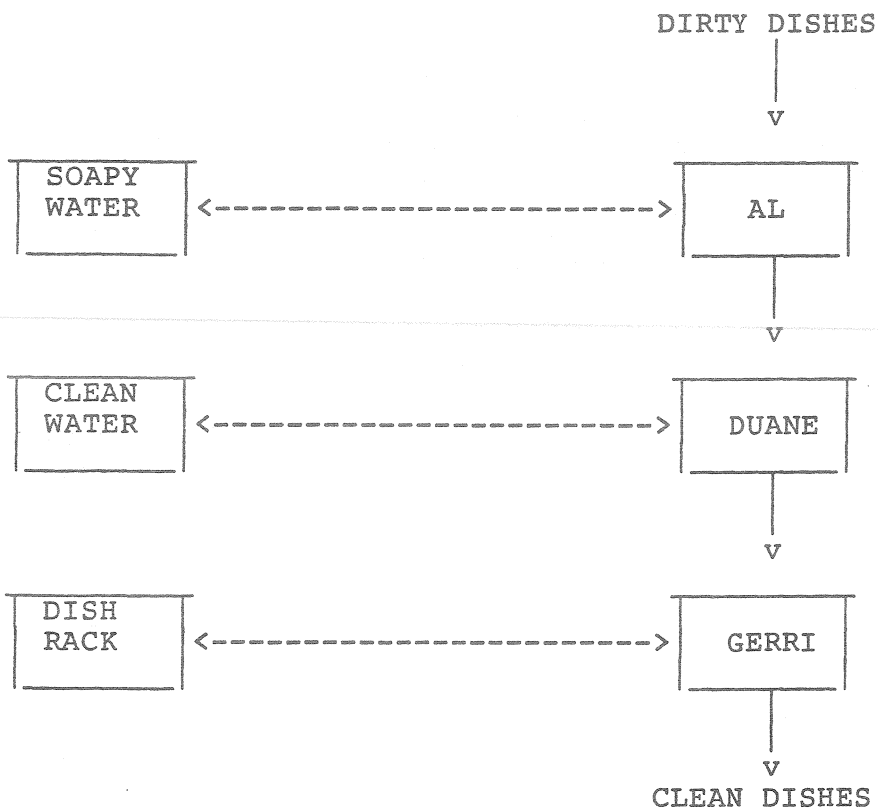


THE FOLLOWING SHOWS THE RELATIONSHIP OF THE I/O PROCESSORS (LAYER 3 WITH THE DATA STORAGE CONTROLLER - DSC) AND THE CSP (LAYER 2):



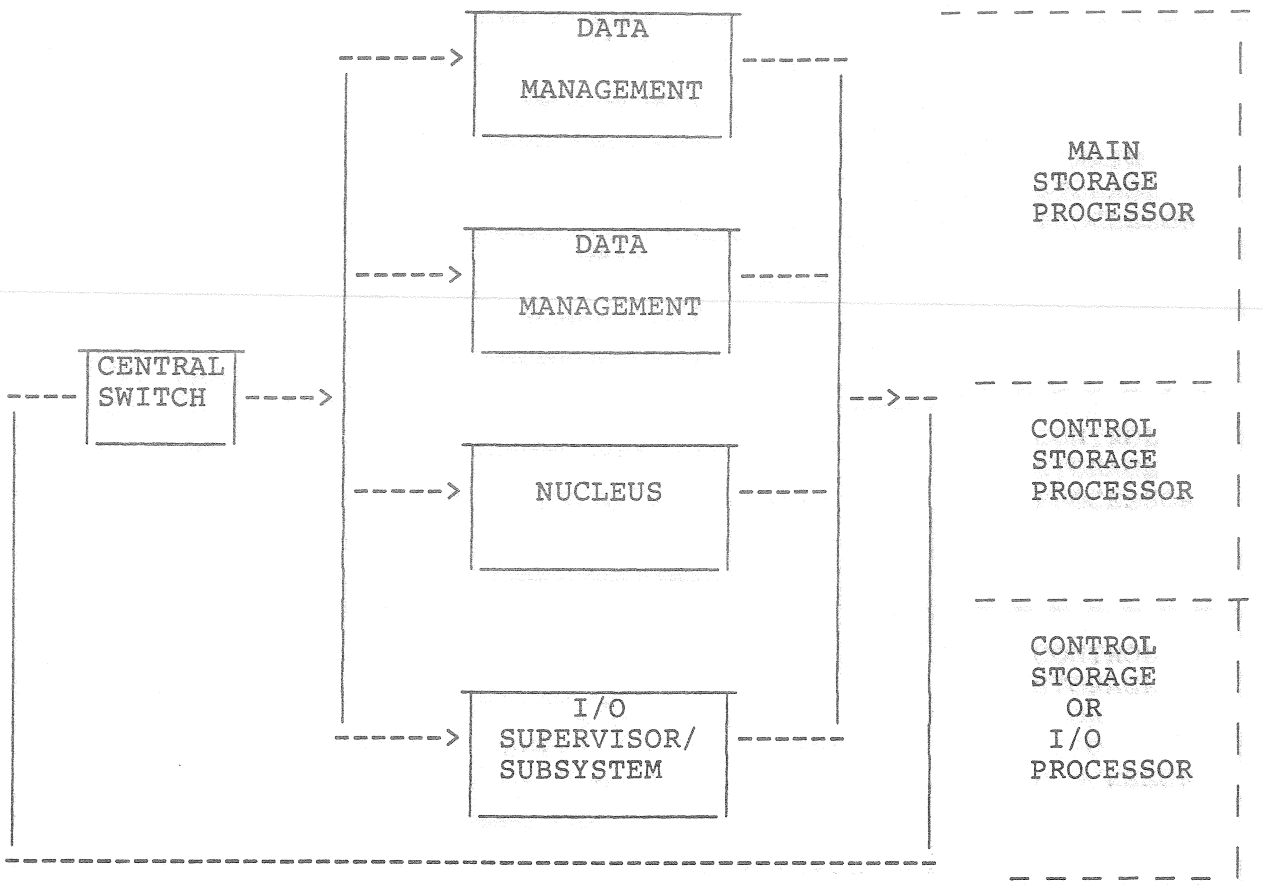
8.4 PIPLELINED OPERATING SYSTEM

WHAT IS PIPELINING?



. PIPELINING APPLIED TO A HIGHER LEVEL OF ABSTRACTION

. EXECUTION STATIONS ARE PROCESSORS



8.5 PROCESSORS SUMMARY

REQUIRED PROCESSORS ON SYSTEM/36:

- MAIN STORAGE PROCESSOR (MSP)
- CONTROL STORAGE PROCESSOR (CSP)
- SECOND WORK STATION CONTROLLER (WSC/2) -- PROVIDES CAPABILITY FOR THE 1ST THROUGH 36TH LOCAL WORKSTATIONS

ADDITIONAL PROCESSORS AVAILABLE ON SYSTEM/36:

- DATA STORAGE CONTROLLER (DSC), INSTALLED WHEN TAPE IS ORDERED
- EIGHT LINE COMMUNICATIONS ADAPTER (ELCA), INSTALLED WHEN MORE THAN ONE COMMUNICATION LINE IS USED ON A 5360
- PRINTER CONTROLLER (PRT) FOR THE 3262
- MAGNETIC INK CHARACTER RECOGNITION UNIT CONTROLLER (MICR) FOR THE 1255
- SECOND WORK STATION CONTROLLER (WSC/2) -- PROVIDES CAPABILITY FOR THE 37TH THROUGH 72ND LOCAL WORKSTATIONS

PROCESSOR SPECIFICATIONS

FOR 5360, 5362, AND 5364:

PROCESSOR NAME	PROCESSOR TYPE	ADDRESS WIDTH (BITS)	DATA WIDTH (BITS)	MIPS (NATIVE)	STORAGE SIZE (KB)	MEMORY CYCLE (NS)	INST SET
MSP	MSP	16 - 24	8 - 2048	.360	128 -2048	200	SS
CSP	CSP	16	16	1.3	64 - 128	200	RR
WSC	CSP/I	16	16	1.3	32 - 96	200	RR

FOR 5360:

PROCESSOR NAME	PROCESSOR TYPE	ADDRESS WIDTH (BITS)	DATA WIDTH (BITS)	MIPS (NATIVE)	STORAGE SIZE (KB)	MEMORY CYCLE (NS)	INST SET
PRT	ATOM	16	4 - 8	1.1	8	325	RR
DSC	CSP/I	16	16	1.3	128	200	RR
ELCA	CSP/I	16	16	1.3	32 - 64	200	RR
MICR	ATOM	16	4 - 8	1.1	6 - 34	325	RR
WSC/2	CSP/I	16	16	1.3	32 - 96	200	RR

FOR 5360 MODEL D (ONLY)

PROCESSOR NAME	PROCESSOR TYPE	ADDRESS WIDTH (BITS)	DATA WIDTH (BITS)	MIPS (NATIVE)	STORAGE SIZE (KB)	MEMORY CYCLE (NS)	INST SET
MSP	MSP	16 - 24	8 - 2048	.610	128 -7168	100	SS
CSP	CSP	16	16	1.9	64 - 128	50	RR

DEFINITIONS:

ATOM	A TINY OPTIMIZED MICROPROCESSOR
CSP/I	CONTROL STORAGE PROCESSOR FOR I/O
MIPS	MILLIONS OF INSTRUCTIONS PER SECOND
NS	NANOSECONDS (I.E., BILLIONTH OF A SECOND) PER BYTE
RR	REGISTER-TO-REGISTER INSTRUCTION SET
SS	STORAGE-TO-STORAGE INSTRUCTION SET

9.0 SUMMARY

THE SYSTEM/36 INCORPORATES INNOVATIVE TECHNOLOGY TO MAINTAIN A HIGH LEVEL OF SYSTEM:

. USABILITY

. AVAILABILITY

. DATA INTEGRITY

THE DISTRIBUTION OF FUNCTION WITHIN THE SYSTEM/36 PROVIDES:

. PARALLELISM WITHIN THE OPERATING SYSTEM IN ALL MAJOR AREAS

. SEVERAL PROCESSORS, EACH TUNED TO A SEPARATE TASK; THUS THE PERFORMANCE AND RELIABILITY OF THE ENTIRE SYSTEM IS SIGNIFICANTLY ENHANCED

. A COMBINATION OF:
  - DATA PROCESSING
  - OFFICE AND PROFESSIONAL SERVICES
  - DISTRIBUTED PROCESSING

INTO ONE MULTIPURPOSE SYSTEM.