

IOS LAB ASSIGNMENT

- 1) Deadstart the IOS with your binaries.
- 2) Produce LISTP and LISTO printouts.
- 3) Test and debug the TNG1 and TNG2 overlays.
- 4) Using the debugger, locate the following in IOP0:
 - a) The MATX table. Address 6607
 - b) The contents of operand registers 0-63.
 - c) The overlay table entries for TNG1 and TNG2.
 - d) The buffer memory addresses of these overlays.
 - e) Their local memory addresses.
 - f) The executable code.
- 5) Using the debugger, set a breakpoint at the end of TNG2.
Then locate the activity descriptor and SMOD for the overlay. Remember this means you will have to enter the debugger in an IOP other than the one in which you will call TNG1.
- 6) Debug a "bad" TNG1 provided by the instructor.
Use PATCH to correct the problem and test the corrected TNG1.

*Find OUPD, Do Formula
70, 71*

Please bring yourself, compiled versions of your overlays, this page, and a "script" of what you will do for the exercise.

Arrange to bring -- SG-0051
SM-0046
SM-0007
IOS Workbook

IMPORTANT OPERAND REGISTERS

<u>Symbol</u>	<u>Value</u>	<u>Description</u>
AA	Ø	Base address of kernel (always Ø)
%B	3	Current overlay base address (Ø if none)
%ACTIVE	6	Current Activity Descriptor (177777 if none)
%SMOD	7	Current SMOD (177777 if none)
%FUNC	1Ø	Kernel service function code number
%EX	11	Kernel service entry point (SERVICE)
%INIT	17	Initialization complete flag
TMP	26	Target memory processor
%OVLNUM	63	Current overlay number (177777 if none)
IFLG	72	Interrupt flag, checked in idle loop
%ICHAN	1Ø6	Interrupting channel number
%MYID	1Ø7	IOP number
%FEI	11Ø	Front end interrupt table
%OVPT	121	Pointer to base of overlay table
%MEMORY	141	Pointer to start of overlay memory
%OVEN	142	Current overlay table entry address
%OVCNT	143	Current number of overlays in memory
%HISP	165	High speed channel number
%HISPBUF	166	Dedicated HISP buffer address
%SSD	167	SSD high speed channel number
%SSDBUF	17Ø	Dedicated SSD buffer address
%DD49	245	Base address of resident code (data overlay)

address of ovl tbl entry = (OVLTABLE) + (4 * overlay number)
(OVLTABLE) = R1%OVPT

OVERLAY TABLE ENTRY

overlay length in words (12 bits)	MOS upper (4 bits)
MOS lower (16 bits)	
parameter register descriptor (e.g., 007430)	
local memory address, if resident (0 if not resident)	

OVERLAY MACRO HEADER

ASCII Overlay Name (4 parcels)	
"	
"	
"	
overlay type (1 bit)	overlay number (15 bits)
parameter register descriptor (e.g., 007430)	

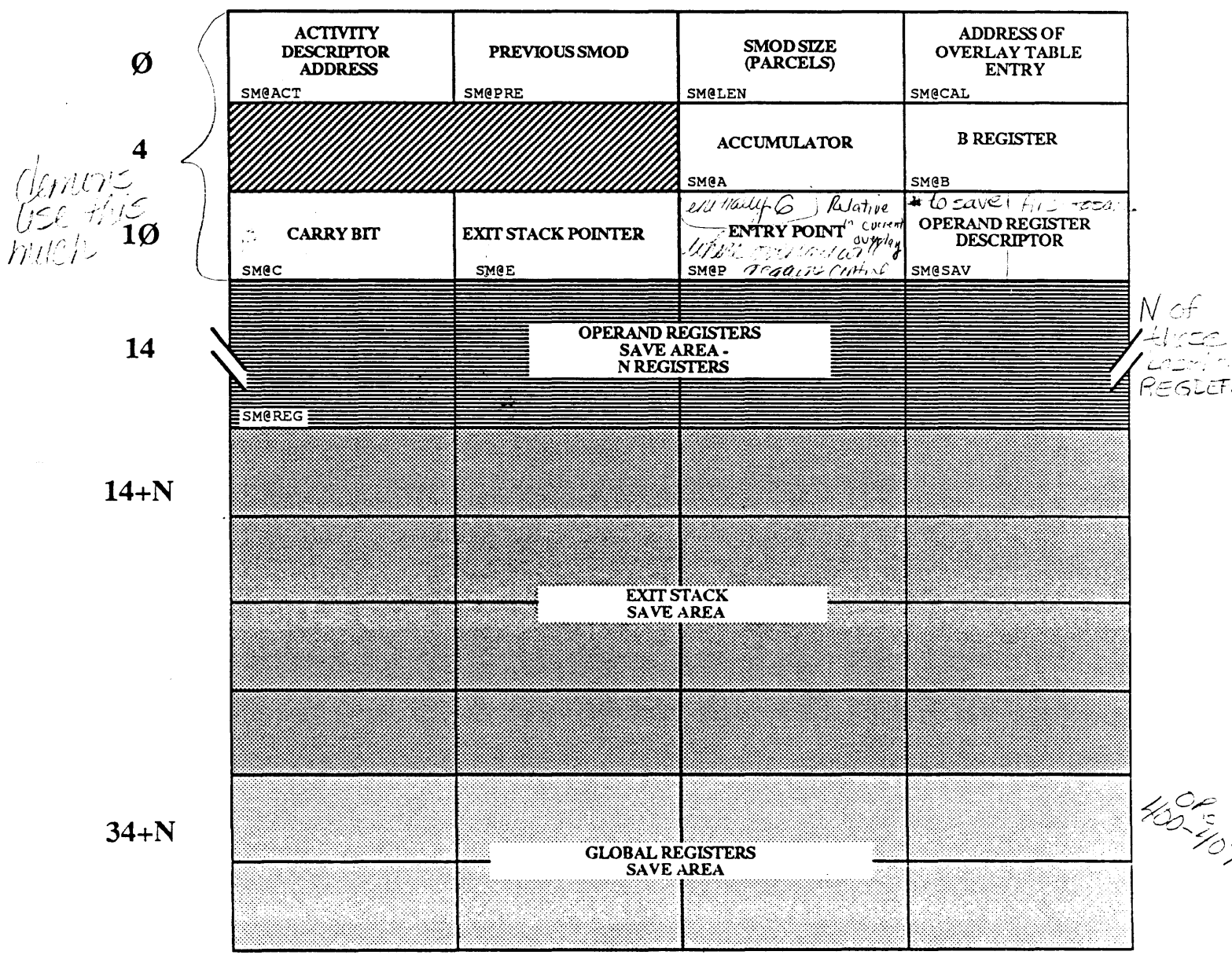
OVERLAY LOCAL MEMORY BLOCK HEADER (MD@)

MD
adjacent block list succeeding block pointer
adjacent block list previous block pointer
type (0 means header/trailer) (1 means free) (2 means overlay)
memory search list forward pointer
memory search list backward pointer
overlay table entry address for type = 2
0

1 per overlay

0 - 013 minimum to SMOD

STORAGE MODULE (SMOD)



$$R \% SMOD = ORN$$

ACTIVITY DESCRIPTOR

0	<p>LINK FOR QUEUES AD@QL</p> <p><i>points to next AD on ECPG</i></p>	<p><i>EACTH chain</i> ACTIVITY DESCRIPTOR CHAIN OF AD@AL</p> <p><i>those in LM</i></p>	<p>ACTIVITY PRIORITY (BITS 12-15) AD@PRI</p> <p><i>0-17 priority</i></p>	<p>MOS SW STACK ADDR (UPPER) AD@MSU</p> <p><i>BMR addr</i></p>
4	<p>MOS SW STACK ADDR (LOWER) AD@MSL</p> <p><i>BMR SS for this AD</i></p>	<p>MOS SW STACK SIZE AD@SIZ</p>		
10	<p>CURRENT SMOD ADDRESS AD@SMD</p> <p><i>the entry in the SS</i></p>	<p>ACTIVE FLAG DEMON FLAG AD@FLG</p> <p><i>for demo only</i></p>	<p>FUNCTION CODE AD@FU</p> <p><i>last PCB issued</i></p>	<p>RESPONSE CODE AD@RC</p> <p><i>KSR return status</i></p>
14	<p>RETURNED PARAMETER 1 AD@P1</p> <p><i>used for TPUSHs, for example</i></p>	<p>RETURNED PARAMETER 2 AD@P2</p>	<p>RETURNED PARAMETER 3 AD@P3</p>	<p>RETURNED PARAMETER 4 AD@P4</p>
20	<p>ASCII ACTIVITY NAME AD@NM</p> <p><i>root overlay. name of this activity</i></p>			

8-7
DH

October 2, 1986

(ESMD) → SS - collection of SMDs