Problem Determination under VSE/ESA
Session E56

VSE/ESA
Technical Conference
Miami, Florida
Oct 7th – 10th, 2002

Charles E. Olsen
colsen@us.ibm.com

RETURN TO INDEX

VSE Runs on Hardware

Machines eat hexadecimal for lunch:

It's all "ones" and "zeros":

```
0000 = 0      1000 = 8
0001 = 1      1001 = 9
0010 = 2      1010 = 10 (A)
0011 = 3      1011 = 11 (B)
0100 = 4      1100 = 12 (C)
0101 = 5      1101 = 13 (D)
0110 = 6      1110 = 14 (E)
0111 = 7      1111 = 15 (F)
```

Of course, nothing is quite that simple. Hex is always shown as a byte (8 bits):

Hex	Dec
00	00
40	64
80	128
C0	196
FF	255

Extended Binary Coded Decimal Interchange Code (EBCDIC)

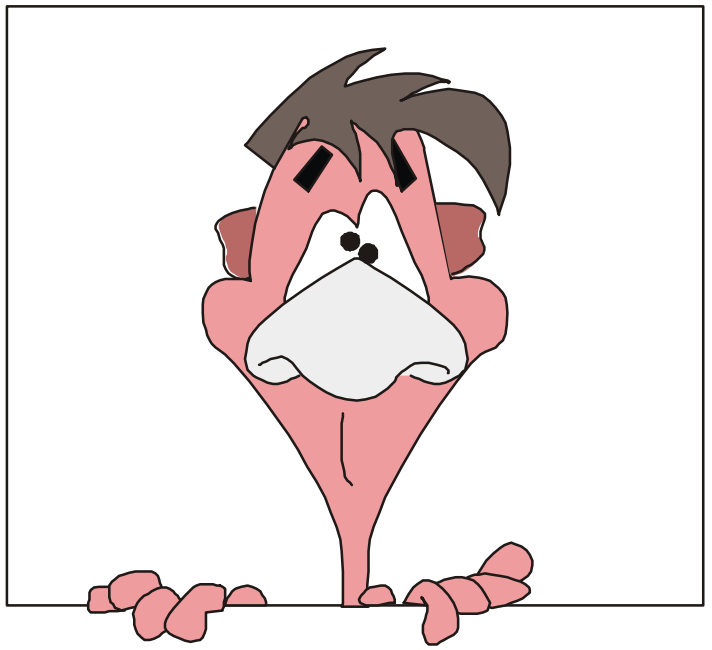
4A	$	81	a	C1	A	F0	0
4B	.	82	b	C2	B	F1	1
4C	<	83	c	C3	C	F2	2
4D	(	84	d	C4	D	F3	3
4E	+	85	e	C5	E	F4	4
4F	|	86	f	C6	F	F5	5
50	&	87	g	C7	G	F6	6
5A	!	88	h	C8	H	F7	7
5B	$	89	i	C9	I	F8	8
5C	*	91	j	D1	J	F9	9
5D	)	92	k	D2	K		
5E	;	93	l	D3	L		
5F	(	94	m	D4	M		
60	-	95	n	D5	N		
61	/	96	o	D6	O		
6B	,	97	p	D7	P		
6C	%	98	q	D8	Q		
6D	_	99	r	D9	R		
6E	>	A2	s	E2	S		
6F	?	A3	t	E3	T		
7A	:	A4	u	E4	U		
7B	#	A5	v	E5	V		
7C	@	A6	w	E6	W		
7D	'	A7	x	E7	X		
7E	=	A8	y	E8	Y		
7F	"	A9	z	E9	Z		

Exercise One: Program Check

Exercise 1:

1. Recognize VSE/ESA Dump Format.
2. Locate a hexadecimal address in a dump.
3. Identify the module containing a given address, using:
   - Base Registers
   - Branch and Link registers
   - Scanning for an eyecatcher.

Caution: Here comes the boring stuff.



Exercise One: Program Check

Understanding the following machine instructions will be helpful in this exercise:

47C0RDDD Conditional branch: Location derived from adding **DDD** to the contents of base register **R**. This instruction uses the condition code (**C**) set by a previous instruction (compare or test).

e.g. **4770415C** = Branch to Register 4 + x'**15C**', if previous check was "not equal".
4780415C = Branch to above address, if field is zero.

47F0RDDD Unconditional branch

e.g. **47F0415C** = Branch to Register 4 + x'**15C**'

45B0RDDD Branch and Link (BAL): To **DDD** plus base register **R**. Stores next instruction address in register **B**.

e.g. **45E0415C** = Used for sub-routine calls. Branch to Register 4 + x'**15C**'.
In addition, place the return address in register 14.

05BR Branch and Link Register (BALR): To address is in register **R**. Store return address in register **B**.

e.g. **05E5** = Branch to where register 5 is pointing, and, in addition, place the return address in register 14.
05C0 = Store the contents of the next sequential instruction in register 12, but do not branch. This sets register 12 up as a base register.

You can recognize "branch and link" registers by examining the high-order byte in the register. If the program is executing in 31-bit mode, the bal register will have the x'80' bit turned on in the high-order byte. If executing in 24-bit mode, the high-order two bits contain the instruction length. (b'10000000' or x'80' for 4 byte instruction; b'01000000' or x'40' for a two byte instruction).

Exercise One: Program Check

Problem Description:

Getting program check interruption in some COBOL programs. Message 0S03I with interruption code 05 using ISAM interface (IIP).

Console log excerpt:

```
0S03I PROGRAM CHECK INTERRUPTION - HEX LOCATION 001C3E7A -INTERRUPTION CODE 05-
      ADDRESSING EXCEPTION
0S00I JOB CSFLDUPD CANCELED
0S07I PROBLEM PROGRAM  PSW = 071D0000 801C3E7C
0S30I DUMP STARTED. MEMBER=DBG00018.DUMP IN SUBLIB=SYSDUMP.BG
1I49I DUMP LIBRARY FULL
```

Exercise One: Program Check

Analysis:

1. When VSE detects a program check in program mode (bit in PSW), it will take a dump if:

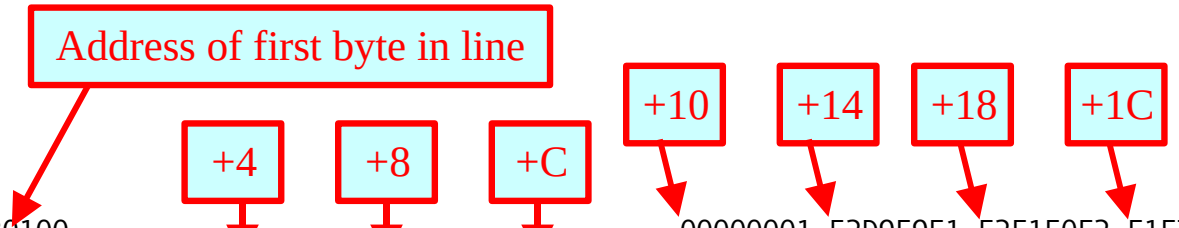
// **OPTION NODUMP**	No dump will be taken
// **OPTION PARTDUMP**	Failing partition and selected supv control blocks will be dumped
// **OPTION DUMP**	Failing partition and entire supervisor will be dumped
// **OPTION SYSDUMP**	Dump will be taken to VSE System Dump Library (SYSDUMP)
// **OPTION DSPDUMP**	Data Space will be dumped (see caveats)
// **OPTION SYSDUMPC**	Dump will not be spooled to SYSLST if SYSDUMP library is full.

2. If the abend occurs in an SVA phase, the failing phase is included in the dump
3. Locate point of foul, and establish module where the error occurred:
   - Check for bal registers.
   - Check for base registers.
   - Check for eyecatcher.

Exercise One: Program Check

Listing:

```
// JOB CSFLDUPD  --EDIT/OVERLAY FIELDS IN C.S.FILE--              DATE 10/07/1997, CLOCK 20/02/45
// OPTION DUMP
// ASSGN SYS005,897  'I/P KEYFAST TAPE 80/80'
// LIBDEF PHASE,SEARCH=IJSYSRS.SYSLIB
// EXEC CSFLDUPD,SIZE=AUTO
        RUN DATE 10/07/97              C. S. FILE EDIT              -CSFLDUPD-                  PAGE  1
        SOC. SEC. #   JILT   R-CODE   DISH   TELEPHONE#   RATE OF PAY   BIRTH DATE   COPE   -*-   M E S S A G E S
        -----------   ----   ------   ----   ----------   -----------   ----------   ----   ------------------

JOBNAME=CSFLDUPD           DATE=10/07/1997  TIME=20:03:08  CPUID=A0021793 91210180  COMP=56860660615C  PAGE     1

0S03I PROGRAM CHECK INTERRUPTION  - HEX LOCATION 001C3E7A - INTERRUPTION CODE 05 - ADDRESSING EXCEPTION
0S00I JOB CSFLDUPD CANCELED
0S07I PROBLEM PROGRAM  PSW = 071D0000 801C3E7C
0S30I DUMP STARTED. MEMBER=DBG00018.DUMP IN SUBLIB=SYSDUMP.BG
1I49I DUMP LIBRARY FULL

*SYMPTOM RECORD
```




```
00330100                                  00000001 E2D9F9F1 F2F1F0F2 F1F7F9F3  00                ....SR9121021793
00330120 F2F07AF0 F37AF0F8 7AF0F0F9 F761F1F0 61F0F7F0 F07AF0F3 7AF0F87A F0F0F5F6   20:03:08:0097/10/0700:03:08:0056
00330140 F8F6F0F6 F6F0F6F1 F5C30000 E2C3D7D9 C5D84040 00000000 00000000 00400074   860660615C..SCPREQ  ......... ..
00330160 000000B4 002B00B4 000000DF 00000000 00000000 00000000 00000000 00000000   ................................
00330180 00000000 00000000 C1C261E2 F2F0F0F0 40D9C5C7 E261C6C6 C6C6C640 D4E261F0   ........AB/S2000 REGS/FFFFF MS/0
003301A0 E2F0F3C9 40D9C9C4 E261C9D2 D8E5D9D4 40D6C6C6 E261F0F0 F0F0F1C5 F5C340C1   S03I RIDS/IKQVRM OFFS/00001E5C A
003301C0 C261E2F0 F0F0F540 D1D6C26D D5C1D4C5 7EC3E2C6 D3C4E4D7 C440C4E4 D4D7C5C4   B/S0005 JOB_NAME=CSFLDUPD DUMPED
003301E0 6DC4C1E3 C17EC2C7 60D7C1D9 E3C9E3C9 D6D540                               _DATA=BG-PARTITION
```

Exercise One: Program Check

```
PSW AND REGISTERS OF ENDING TASK

PSW      071D0000 801C3E7C

GR 0-7   40401238 00000068 00413F38 00000068 00000068 00000004 801C2582 001C3DB8
   8-F   001C3EE0 801C20F2 00410228 00410068 0040AE00 00413B18 801C4392 00000000

AR 0-7   00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
   8-F   00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000

FP 0-3   40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040

CR 0-7   04B1EE40 01000002 00000000 40000000 00000000 01077040 10000000 01000002
   8-F   00000000 00000000 00000000 00000000 00000000 01000002 DF001076 000414E8

IKQVRM   SVA       ADDRESS IS 001C2020  LENGTH IS 000158EB
001C2020 47F0F07E C9D2D8E5 E2D44040 F1F5C340 C4E8F4F3 F6F5F240 F1F161F2 F061F9F6 00 00=IKQVSM  15C DY43652 11/20/96
001C2040 40F5F6F8 F660F0F6 F640C3D6 D7E8D9C9 C7C8E340 C9C2D440 C3D6D9D7 4B40F1F9    5686-066 COPYRIGHT IBM CORP. 19
001C2060 F7F940F1 F9F9F540 C1D3D340 D9C9C7C8 E3E240D9 C5E2C5D9 E5C5C440 D3C9C3C5   79 1995 ALL RIGHTS RESERVED LICE
001C2080 D5E2C5C4 40D4C1E3 C5D9C9C1 D3E260D7 D9D6D7C5 D9E3E840 D6C640C9 C2D490EC   NSED MATERIALS-PROPERTY OF IBM..
001C20A0 D00C18C1 5810FDB0 41110000 89100001 12114780 F09E9680 D00C47F0 F0D089E0   ...A........i.......0.o....00.i.

Locate the failing instruction, and back up looking for an eyecatcher.

001C3DA0 00000000 00000000 00000000 00000000 00000000 00000000 47F07010 C9D2D8D9   ........................0..IKQR
001C3DC0 E3E54040 F1F5C3F0 1BFF5820 D1745800 C00C9102 C0204780 702A5810 A01447F0   TV  15C0....J.....j...........0
001C3DE0 70364810 D1824811 20044A20 D1809108 C02147E0 70829110 A0284780 70529108   ....Jb......J.j......bj.......j.
001C3E00 B0784710 70524540 70CC4110 00045510 C0144720 70C69102 A0284780 707A9102   ....... .............Fj......:j.
001C3E20 C0204770 707A9108 B0784770 707A4122 00041810 50201000 07FE9110 A02847E0   .....:j.....:.....&.....j.....
001C3E40 70969108 B0784710 70964540 70CC5840 C0149102 A0284780 70BA9102 C0204770   .oj......o. ... ..j.......j.....
001C3E60 70BA9108 B0784770 70BA4122 00044B10 71201514 472070C6 1830E0E2 07FE41F0   ..j.....................F.......0
001C3E80 002307FE 50E0D0C0 18315810 D20C184D 18D141F0 00481A1F 58F0B090 05EF18D4   ....&.......K..(.J.0.....0.....M
001C3EA0 58E0D0C0 07FE0000 00000000 00000000 00000000 00000000 00000000           ..............
001C3EC0 00000000 00000000 00000000 00000000 00000000 00000000 0004               ..................
```

Exercise One: Program Check

Search in RETAIN for "PROGCK IKQRTV":

** SOFTWARE SUPPORT FACILITY TITLE PAGE 1 **

LIB/FILE(S) CURRENTLY SELECTED DS/AC
PROGCK IKQRTV

THE ABOVE SEARCH ARGUMENT RESULTED IN 3 MATCHES
USER PTF#,APAR#,ABS.L1 (APAR DEFAULTS)

```
1 UD47798 DY41632 PROGCK IN VSAM USING IIP (ISAM INTERFACE)
2         DY41110 PROGCK IN IKQRTV DURING GET PROCESSING WIT
3         II08262 USING THE ISAM INTERFACE PROGRAM (IIP) RES
```

APAR: II08262

-
-
-

ERROR DESCRIPTION:
Using the ISAM Interface Program (IIP) from old (backlevel) COBOL programs, may result in progck in VSAM due to invalid 31bit addresses passed to VSAM in the RPL. COBOL is using the high order byte of those fields in the DTFIS (IOASAD2, KARGAD2) that are used by IIP to modify the according RPL fields (RPLAREA and RPLARG).

Exercise One: Program Check

Since most of the customers have no source for these programs and the COBOL release can no longer be maintained, VSAM provides the following patch to phase IIPOPEN to clear the high order byte, when the addresses are moved from the second to the first part of the DTFIS:

```
// JOB PATCH                                          APAR= II08262
// EXEC MSHP,SIZE=900K
 PATCH S=IJSYSRS.SYSLIB
 AFFECTS PHASE=IIPOPEN
 ALT 00EE D20380747008 : D20280757009   /* move ioarea   addr */
 ALT 00F4 D20380707014 : D20280717015   /* move workarea addr */
 ALT 00FA D20380607010 : D20280617011   /* move key arg  addr */
 ALT 0218 D20380747008 : D20280757009   /* move ioarea   addr */
 ALT 021E D20380707014 : D20280717015   /* move workarea addr */
 ALT 0224 D20380607010 : D20280617011   /* move key arg  addr */
/*
/&
```

The symptoms of the ABEND in VSAM code are most commonly a PROGCK in phase IKQVRM, modules IKQIXS or IKQRTV or similar.

Determine VSE System Status

Determine System Status:

Tell difference between Lions and Tigers and Bears ... Oh My:

- System disabled wait state (hard wait)
- System disabled loop
- System enabled wait state (soft wait)
- Partition wait
- Partition loop

Determine VSE System Status

1. Is Attention Routine responding?

   - Immediate command:

```
REPLID
AR 0015 1I88I NO REPLIES OUTSTANDING
```

   Response? No… go to page **22** to check for hard wait or disabled loop.

2. Reply to all outstanding messages, then attempt a more complex command:

```
VOLUME 140
AR 0015 CUU  CODE DEV.-TYP   VOLID  USAGE   SHARED    STATUS    CAPACITY
AR 0015 140  6E    3390-3   DOSRES  USED                          1112  CYL
AR 0015 1I40I  READY
```

   Response? No…Enter "**RC**" and retry command. Still no response? Look for console management or vendor product problems

3. Are higher priority partitions running?

```
PRTY
AR 0015 PRTY Z,Y,P,C,BG,FB,FA,F9,F8,F7,F6,F5,F4,F2,F3,F1

POWER:
     D T
     AR 0015 1C39I COMMAND PASSED TO VSE/POWER
     F1 0001 1R46I  TIME IS 17:23:07, DATE IS 07/31/2002
     F1 0001 1R46I  019 PAGES FIXED, 023 CURRENT TASKS
```

Determine VSE System Status

CICS:

MSG F2

```
F2-0100
```

100 CEMT I TA

```
F2-0100
F2 0103
    Tas(0000023) Tra(CXPB)           Sus Tas Pri( 001 )
    Sta(S ) Use(DBDCCICS) Rec(X'B7F46C3740692D02') Hty(OPEN_ANY)
    Tas(0000025) Tra(ICVS)           Sus Tas Pri( 001 )
.
.
.
    RESPONSE: NORMAL TIME:  17.25.52  DATE: 07.31.02
    SYSID=CIC1 APPLID=DBDCCICS
```

100

VTAM:

D NET,MAJNODES

```
AR 0015 1C39I COMMAND PASSED TO ACF/VTAM
F3 0003 IST097I DISPLAY ACCEPTED
F3 0003 IST350I DISPLAY TYPE = MAJOR NODES
F3 0003 IST089I VTAMSEG  TYPE = APPL SEGMENT     , ACTIV
F3 0003 IST089I VTAMSEG  TYPE = APPL SEGMENT     , ACTIV
F3 0003 IST089I NODE0001 TYPE = PU T4/5 MAJ NODE , ACTIV
F3 0003 IST089I ISTPDILU TYPE = CDRSC SEGMENT    , ACTIV
F3 0003 IST089I ISTADJCP TYPE = ADJCP MAJOR NODE , ACTIV
```

Respond? No … Transfer analysis to higher priority partition.

Determine VSE System Status

4. Check status of failing partition:

STATUS F2

```
AR 0015 M23 F2 CICSICCF 82 WAITING FOR I/O, OR ECB POSTING
AR 0015    TCB=00056A60 TIB=0004D600 SAV=00500000
AR 0015    SCB=0004A70C PCB=0004B100 COM=00003BD8
```

-or-

```
82 WAITING FOR I/O ON DEVICE=009
```

-or-

```
82 WAITING ON TIMER INTERRUPT
```

-or-

```
82 WAITING FOR OPERATORS RESPONSE
```

-or-

```
83 READY TO RUN
```

-or-

```
85 WAITING FOR PROGRAM FETCH
```

-or-

```
8C WAITING FOR SUB-TASK TERMINATION
```

-or-

```
8E WAITING FOR LOCKED RESOURCE USED BY <task>
```

-or-

```
95 WAITING FOR DUMP/TRACE PROCESS   USED BY F4
```

-or-

```
57 VSE/POWER WAITING FOR WORK
```

Determine VSE System Status

- If "WAITING FOR I/O, ECB OR TECB", partition is waiting for an event to occur.
  See "Exercise Two, Partition in I/O Wait" on page 25.

- If "READY TO RUN", partition may be in a loop.

- If "WAITING FOR I/O ON DEVICE=<cuu>"
  i. Check using the IUI System Status screen if partition is performing I/O.
  ii. Check status of device using "**STATUS** <cuu>".
  iii. Save partition, SVA, and supervisor to tape using "**DUMP**" command
  iv. Try "**CANCEL** <cuu>". If this doesn't work, try "**CANCEL** <cuu>**,FORCE**".
  Caution: A partition may be doing productive work (i.e. copying large amounts of data) and still show waiting on I/O each time it is checked.

- If "WAITING FOR LOCKED RESOURCE USED BY <task>", check status of "<task>", which owns the resource.

- If any indication that the partition may be waiting on POWER spooled device, also enter "**D A**" and "**D TASKS**".

- If "VSE/POWER WAITING FOR WORK", previously running application has canceled, and partition is paused at end of job boundary. Check for previous error messages in this partition.

Determine VSE System Status

5. Is partition in a wait or loop? Check System Utilization using IUI:

```
IESADMSL.IESEADM       VSE/ESA FUNCTION SELECTION       APPLID: DBDCICS
Enter the number of your selection and press the ENTER key:
          1  Installation
          2  Resource Definition
   ==>    3  Operations
          4  Problem Handling
          5  Program Development
          6  Command Mode
          7  CICS-Supplied Transactions

 PF1=HELP                      3=SIGN OFF                      6=ESCAPE(U)
                               9=Escape(m)
==>
```

Select "**OPERATIONS**", "**SYSTEM STATUS**":

```
IESADMSL.IESEOPS           OPERATIONS                   APPLID: DBDCCICS
Enter the number of your selection and press the ENTER key:
         1  Console
         2  Manage Batch Queues
         3  Display Active Users/Send Message
         4  Enter News
         5  Retrieve Message
   ==>   6  System Status
         7  Backup/Restore
         8  Personal Computer Move Utilities
         9  Transfer Files and Jobs to Another System
```

Determine VSE System Status

Select "6 **SYSTEM STATUS**":

```
IESADMSL.IESESTAT          SYSTEM STATUS                APPLID: DBDCCICS
Enter the number of your selection and press the ENTER key:
   ==>  1  Display System Activity
        2  Display Channel and Device Activity
        3  Display Storage Layout
        4  Display CICS TS Storage
```

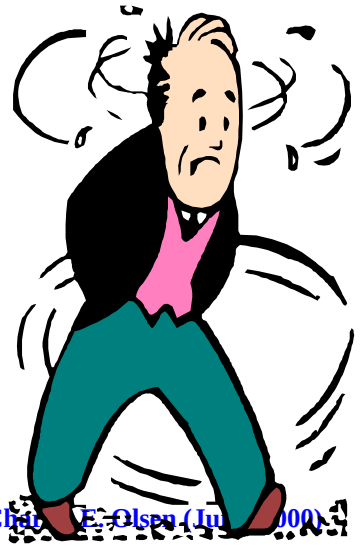
Select "**DISPLAY SYSTEM ACTIVITY**":

```
IESADMDA              DISPLAY SYSTEM ACTIVITY             15 Seconds  23:14:11
*--- SYSTEM (CPUs active:  1  / 0 ) -* *---------- CICS : DBDCCICS ---------*
|CPU      :    *   I/O/Sec:    3     | |No. Tasks:   580   Per Sec    :   *  |
|Pages In :    0   Per Sec:    *     | |Dispatchable:  4   Suspended  :   0  |
|Pages Out:    0   Per Sec:    *     | |Peak Active :  5   MXT reached:   0  |
*-----------------------------------* *-----------------------------------*
Priority: Z,Y,P,C,BG,FB,FA,F9,F8,F7,F6,F5,F4,F2,F3,F1

 ID S JOB NAME  PHASE NAME  ELAPSED     CPU TIME   OVERHEAD   %CPU       I/O
 F1 1 POWSTART    IPWPOWER   02:36:31       1.11        .33              2,268
 F3 3 VTAMSTRT    ISTINCVT   02:36:24       2.52        .93              4,352
 F2 2 CICSICCF    DFHSIP     02:36:20       9.86       2.24              2,752
 F4 4 <=WAITING FOR WORK=>                  .11        .03                 76
 F8 8 <=====STOPPED======>                  .00        .00
 F9 9 <=====STOPPED======>                  .00        .00
 FA A <=====STOPPED======>                  .00        .00
 FB B <=WAITING FOR WORK=>                  .11        .03                 77
 BG 0 TESTJOB     SHOWCAT    00:00:01       1.59        .00                 34
```

Determine VSE System Status

```
IESADMDA              DISPLAY SYSTEM ACTIVITY              16 Seconds  23:15:27
*--- SYSTEM (CPUs active:  1  / 0 ) -* *---------- CICS : DBDCCICS ---------*
|CPU      :  98%   I/O/Sec:    1     | |No. Tasks:   585   Per Sec    :   *   |
|Pages In :    0   Per Sec:    *     | |Dispatchable:  4   Suspended  :   0   |
|Pages Out:    0   Per Sec:    *     | |Most Active :  5   MXT reached:   0   |
*-----------------------------------* *-----------------------------------*
Priority: Z,Y,P,C,BG,FB,FA,F9,F8,F7,F6,F5,F4,F2,F3,F1

 ID S JOB NAME  PHASE NAME  ELAPSED     CPU TIME   OVERHEAD   %CPU       I/O
 F1 1 POWSTART    IPWPOWER   02:37:47       1.11        .33              2,268
 F3 3 VTAMSTRT    ISTINCVT   02:37:40       2.53        .93              4,357
 F2 2 CICSICCF    DFHSIP     02:37:36       9.92       2.24              2,752
 F4 4 <=WAITING FOR WORK=>                  .11        .03                 76
 F5 5 <=====STOPPED======>                  .00        .00
 F6 6 <=====STOPPED======>                  .00        .00
 F8 8 <=====STOPPED======>                  .00        .00
 F9 9 <=====STOPPED======>                  .00        .00
 FB B <=WAITING FOR WORK=>                  .11        .03                 77
 BG 0 TESTJOB     SHOWCAT    00:01:17      76.45        .02    98%          34
```

If partition is in a loop (logging CPU time), record a few PSWs from the current TCBSAVE area prior to canceling partition: **CANCEL BG,DUMP**.
If this does not cancel the job, try: **CANCEL BG,DUMP,FORCE**

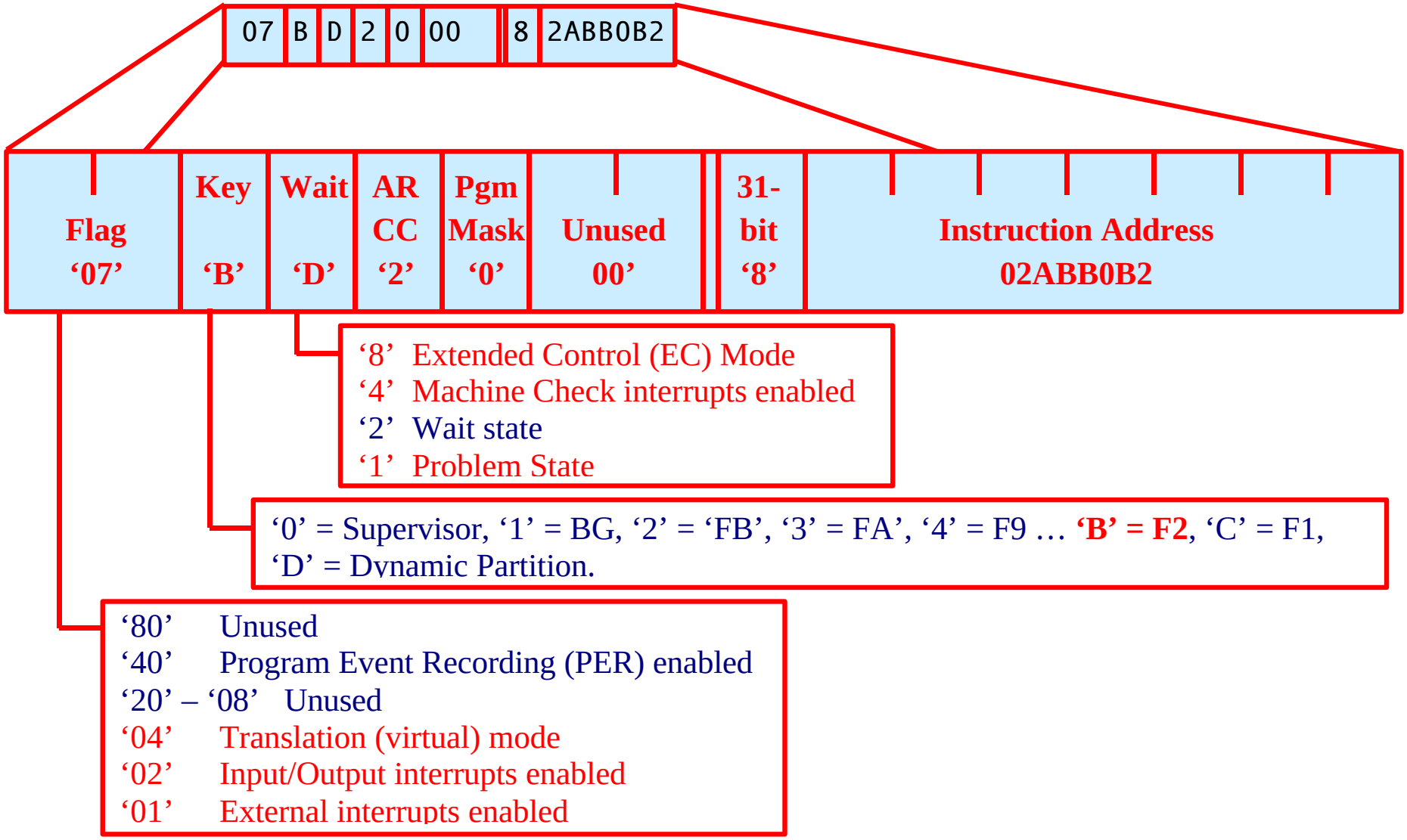


Determine VSE System Status

6. Is system in hard (disabled) wait or disabled loop?

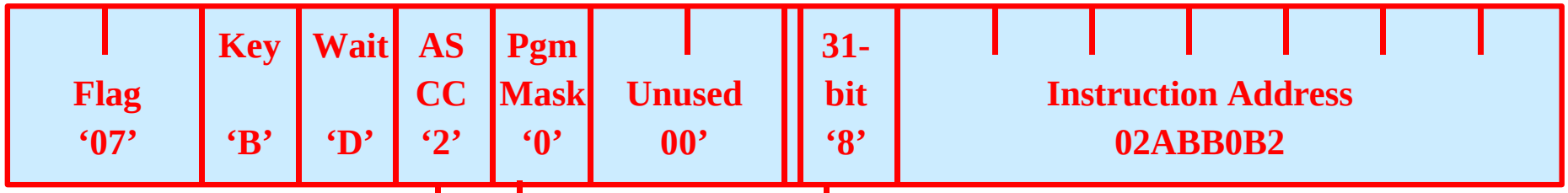
PSW: (Program Status Word, controls instruction execution)

07BD2000 82ABB0B2



Determine VSE System Status

PSW (continued):



'8' If instruction address begins with x'80', machine is operating in AMODE(31) (31-bit addressing mode)

Address Space Control (AS)

'00' = Primary mode
'10' = Access Register (AR) mode, Home Space
'11' = Access Register mode, Secondary Space

Condition Code (CC)

'00' =	Zero Both operands equal	'10' =	Greater than zero First operand high Carry
'01' =	Not zero Less than zero First operand low	'11' =	Not Zero Overflow

Programming Mask:

'8' Fixed-point overflow mask
'4' Decimal overflow mask
'2' Exponent underflow mask
'1' Significance mask

Determine VSE System Status

Hard Wait:

From the hardware console, place the system in stop mode.

If unable to stop the machine, you may be in a disabled PSW load loop
(e.g. if low core is overlaid and Progck New PSW is invalid).

Check the current PSW. Is wait bit on? No, it is probably a disabled loop (see next page).

Hardwait = PSW wait bit (14) is on, and interrupt masks (bits 6&7) are off

x'03' = zero = interrupts disabled
x'02' = on = wait bit

000A0000 00001000	Disaster error detected by VSE/AF supervisor
000A0000 00001122	Abend while processing program check (Normally indicates overlay of supervisor)
000A0000 00CExxxx	Stand-alone dump program. xxxx is rc.
010E0000 0000EEEE	SDAID address stop or DEBUG stop (can be restarted by pressing external interrupt)
010E0000 00EEEEEE	SDAID intervention required (can be restarted by making device ready)
040E0000 00002000	Stand-alone supervisor, normal completion

Determine VSE System Status

1. Check low core. Screen print it. If there is no hardwait code in low core, but the system still does not respond to attention routine commands, we may be in a disabled loop.

   See *VSE/ESA Messages and Codes, Volume 2* under "VSE/Advanced Functions Codes and SVC Errors" for a complete list of hardwait codes.

   x'0FFF' Progck in supervisor mode
   x'0FFE' I/O Error during fetch from IJSYSRS.SYSLIB
   x'0FED' Inconsistency (logic error) detected in supervisor
   x'0FFB' Page Fault during non-pageable supervisor activity
   x'0FF9' Error during Page I/O

   Special codes are placed in low core if errors are detected during IPL, including:

   x'C1 E2' Machine check on clear storage
   x'07 E6 cc uu' IPL input/output error (ccuu is device where error occurred). I/O error sense information will also be stored in low core.
   x'07 E6 C3 E2' Console router error ('CS')
   x'07 E6 C9 C3' Integrated Console error ('IC')
   x'cc 00 0F D0' Error during IPL (cc is supervisor cancel code)
   x'F0 C9 F0 F0 C1' '0I00A' … Low Core may contain message documented in "*VSE/ESA Messages and Codes, Volume 1*"

Determine VSE System Status

2. Perform Store Status (saves PSW and registers in low core)
3. Mount stand-alone tape and ipl the tape.
4. When you receive message “Standalone dump complete”, re-ipl VSE.
5. Load stand-alone tape using IUI.
6. Perform analysis of stand-alone dump.
7. Locate failing application, if not identified in previous step.
8. Contact owner of failing application (IBM, Vendor, local support group)

Disabled Loop:

1. Screen print the first x’100’ bytes of low core.
2. Write down current PSW.
3. Continue with step 3 under “Hard Wait”.
4. If low core is overlaid, this can cause a disabled PSW load loop. Depending on the processor, it may not be possible to stop the CPU to take a standalone dump. In this case, write down (or print) as much of low core as possible.

Exercise 2: Partition in I/O Wait

Exercise 2:

1. Identify the last instruction executed by a partition in a wait state

2. Identify the module where the last instruction occurred using:
   - Base Registers
   - Branch and Link registers
   - Scanning for an eyecatcher.

Documentation:

- If a loop, record a few instructions from TCBSAVE.
- Console log containing analysis.
- AR DUMP of failing partition
- AR DUMP of supervisor and SVA
- If waiting on locked resource, AR DUMP of partition which owns the resource.
- CANCEL <partition>,DUMP

Exercise 2: Partition in I/O Wait

Analysis:

1. Using status command, determine status of partition, and locate main task Task Control Block (TCB).
2. Locate current task save area in TCBSAVE
3. Locate failing PSW.
   - '0A07' (Waiting on I/O or ECB)
     Note: PSW is backed up 10 bytes on SVC 07
   - '0A6E' (Waiting on locked resource)
   - '0A01', '0A02', '0A04', or '0A41' (Waiting on program fetch)
   - Otherwise, partition may be in a loop, or waiting on page I/O.
4. Use base register, or backup from point of impact, looking for a module eyecatcher.

If partition is in a wait state, PSW usually points to a Supervisor Call (SVC = x'0Ann'):

'0A00' = Start I/O (Register 1 points at CCB)

'0A01' = Fetch a phase (Register 1 points at phase name)

'0A02' = Fetch a $B transient (Register 1 points at phase name. If "$$BOPEN" or "$$BCLOSE", Register 0 points a list of files to be processed)

'0A04' = Load a phase and return entry address (See SVC 01)

'0A06' = Cancel

'0A07' = Wait for I/O (See SVC 00)

'0A0E' = End of Job

Exercise 2: Partition in I/O Wait

‘0A29’ = Dequeue (Register 1 points at Resource Control Block = RCB.
RCB+x’04 points at ECB to be freed)

‘0A2A’ = Enqueue (See SVC x’29’)

‘0A3D’ = Request getvis storage (Register 0 contains length of area to be acquired)

‘0A3E’ = Free storage (Register 1 contains pointer to area to be freed.
Register 0 contains length of area to be freed.)

‘0A41’ = CDLOAD (load phase into GETVIS and return address) (See SVC 01)

‘0A6E’ = Lock / Unlock a resource
(Register15 = x’……03’ → Lock request,
Register 15 = x’……01’ → Unlock request
Register 1 points at DTL = Define The Lock
DTL+ 4 (length = 12) contains name of lock)

‘0A71’ = Cross Partition Communication Control (XPCC)
(Register 1 points to CRCB +x’08’ = Path id, connection token)

Normally, the PSW points just past the last instruction to be executed. There are two exceptions:

1. Program check for segment translation exception (x’10’) or page translation exception (x’11’). In this case, the PSW points at the failing instruction.
2. VSE Standard Wait coding, after a Wait on ECB or I/O (SVC 07):

```
91801002     TM   CCBFLAG,POSTED
4710rddd     BO   *+6
0A07         SVC  07
```

If I/O is not yet complete, VSE backs up PSW by 10 bytes, back to Test-Under-Mask.

Exercise 2: Partition in I/O Wait

Problem Description:

VSE/ESA 2.2: Customer installed maintenance for Power in PMR 48228 B550 and now he has jobs that look to stall out. They do not get any I/O or CPU time. They can purge the jobs then restart them and they will run. His Cobol programs calling OEM SoftwareAG look to be the jobs that have problem. Customer request call back today.

Analysis:

```
STATUS
AR 0015    T1F S1F       WAITING FOR I/O, ECB OR TECB
AR 0015    T20 AR        READY TO RUN
AR 0015    T21 BG        VSE/POWER WAITING FOR WORK
AR 0015    T4D     -F1   WAITING FOR I/O, ECB OR TECB
AR 0015    T4E     -F1   WAITING FOR I/O, ECB OR TECB
AR 0015    T55     -F1   WAITING FOR I/O, ECB OR TECB
AR 0015    T22 F1        WAITING FOR I/O, ECB OR TECB POWER MAIN TASK
AR 0015    T72     -F2   WAITING FOR I/O, ECB OR TECB
AR 0015    T73     -F2   WAITING FOR I/O, ECB OR TECB
AR 0015    T23 F2        WAITING FOR I/O, ECB OR TECB
AR 0015    T4F     -F3   WAITING FOR I/O, ECB OR TECB
AR 0015    T50     -F3   WAITING FOR I/O, ECB OR TECB
AR 0015    T51     -F3   WAITING FOR I/O, ECB OR TECB
AR 0015    T52     -F3   WAITING FOR I/O, ECB OR TECB
AR 0015    T53     -F3   WAITING FOR I/O, ECB OR TECB
     • • •
AR 0015 1I40I  READY
```

Exercise 2: Partition in I/O Wait

```
D TASKS
AR 0015 1C39I COMMAND PASSED TO VSE/POWER
F1 0001 1R48I  ***   BEGIN OF DISPLAYING VSE/POWER TCB'S       ***
F1 0001 1R48I  TID ,CUU,TCBADR,T,PHASE(ADDR),REG12 ,STATE(DETAIL)
F1 0001 1R48I  LSNA,   ,509A40, ,SN (5A0D00),5A0D5E,M(RO1=3D509C38)
F1 0001 1R48I  LLDR,   ,50B000, ,LD (5AD000),5AD304,C(RO1=3D50B034)
F1 0001 1R48I  YTES,   ,502B40, ,TV (59C700),59C7E0,C(RO1=3D502B74)
F1 0001 1R48I  XMAS,   ,505A20, ,XM (58FC00),58FF26,M(RO1=3D505BB8)
F1 0001 1R48I  JSPM,   ,506820, ,SM (59CB00),59CD08,Q(RO1=3D2EF0B8)
F1 0001 1R48I  0 CP,   ,5012A0, ,CD (531400),534376,R(----------)   (D TASKS)
F1 0001 1R48I  NTFY,   ,50C560, ,NS (598700),598C24,M(RO1=3D50C5F4)
F1 0001 1R48I  NRV ,SNA,50AB40, ,BS (5BF200),5BF2E6,B(NCB=3D509CC0) (PWRO1,CON)
F1 0001 1R48I  DPST,   ,503D80, ,DP (575600),5756EE,M(RO1=3D503BE0)
F1 0001 1R48I  E BG,FEC,507000, ,XRE(54EA00),54FC36,Q(RO1=3D507020)
F1 0001 1R48I  E F2,FEC,507280, ,XRE(54EA00),54F844,C(RO1=3D5072B4)
     • • •
F1 0001 1R48I  ***      END OF DISPLAYING VSE/POWER TCB'S      ***
```

```
D A
AR 0015 1C39I COMMAND PASSED TO VSE/POWER
F1 0001 1R48I   C-RV ,SNA,  AWAITING     NODE=3DPWRO1
F1 0001 1R48I   F2,FEC,2A,  ADA5SRC ,30667,A
F1 0001 1R48I   F3,FEC,3V,  VTAMSTRT,22872,3
F1 0001 1R48I   F4,FEC,4A,  ADA5AUT ,30670,A
F1 0001 1R48I   F5,FEC,5A,  ADA5MDL ,30673,A
F1 0001 1R48I   F6,FEC,6A,  INACTIVE,
     • • •
```

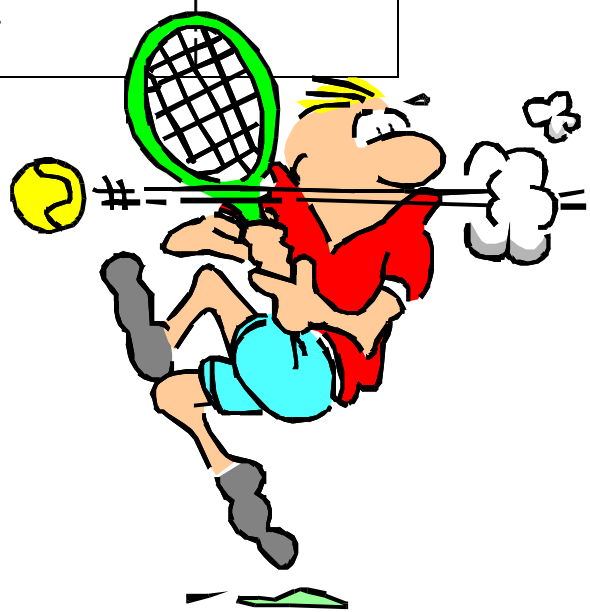
Exercise 2: Partition in I/O Wait

```
STATUS P1
AR 0015    T34 P1 DRP400 82 WAITING FOR I/O, OR ECB POSTING
AR 0015       TCB=003722E8 TIB=00372268 SAV=00540000
AR 0015       SCB=00372000 PCB=00372088 COM=003724F0

AR 0015 1I40I  READY
```

Save area: ("SAV=" points at partition save area)

```
SHOW P1,540000.70
540000        0   C4D9D7F4 F0F04040 07DD0000 0011C9BC | DRP400  ......I. |
540010       10   005DB700[9] 00583F80[A] 0011CB07[B] 0011BB08[C] | .).............. |
540020       20   00584748[D] 9011C3C2[E] 0011BB08[F] 00585432[0] | ......CB........ |
540030       30   005845F8[1] 00584630[2] 00000085[3] 005845F8[4] | ...8.......e...8 |
540040       40   50585588[5] 00000420[6] 00584080[7] 00584630[8] | ...h.....  ..... |
540050       50   0000AF75 6AE55569 40404040 40404040 | .....V..         |
540060       60   40404040 40404040 40404040 40404040 |
```

PSW: Shows last instruction executed



Exercise 2: Partition in I/O Wait

PSW, however, is not pointing in the partition, it is pointing into the SVA:

```
11BB00        0   001AF1F9 F2F0F5F0 47F0F0BA 47F0F0BE | ..192050.00..00. |
11BB10       10   47F0F0B6 47F0F0C2 47F0F0B6 47F0F0B6 | .00..00B.00..00. |
11BB20       20   47F0F0B6 47F0F098 47F0F0A6 5BC9D1C4 | .00..00q.00w$IJD |
11BB30       30   D7D9E340 C3F4F4F0 C4C1F4F1 F6F8F440 | PRT C440DA41684  |
11BB40       40   F5F6F8F6 60F0F0F7 404DC35D 40C3D6D7 | 5686-007 (C) COP |
11BB50       50   E8D9C9C7 C8E340C9 C2D440C3 D6D9D740 | YRIGHT IBM CORP  |
11BB60       60   F1F9F8F3 6BF1F9F8 F940C1D3 D340D9C9 | 1983,1989 ALL RI |
11BB70       70   C7C8E3E2 40D9C5E2 C5D9E5C5 C440D3C9 | GHTS RESERVED LI |
11BB80       80   C3C5D5E2 C5C440D4 C1E3C5D9 C9C1D3E2 | CENSED MATERIALS |
      •
      •
      •
11C990       90   07FE50E0 91E45860 90D8D501 605A908C | ....jU.-.QN.-!.. |
11C9A0       A0   4780CEB0 41109060 0A009180 10024710 | .......-..j..... |
11C9B0       B0   CEAC0A07 47F0CEBE 581090E0 91801002 | .....0......j... |
11C9C0       C0   4710CEBE 0A079108 908B4770 CEF29180 | ......n......2j. |
11C9D0       D0   909047E0 CEF29101 909247E0 CEDE5860 | .....2j..k.....- |
```

PSW points here

Reg 12 points here

Branch shows Reg12 to be base reg

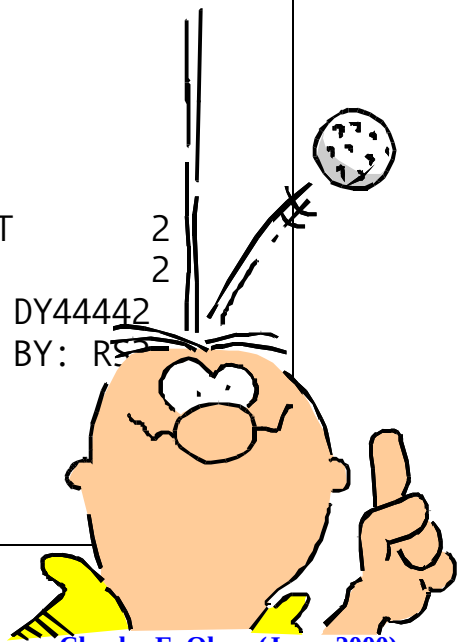
CCB (Reg1): It is not posted, but is complete (x'0C').

```
5845F8        0   00000001 0C000114 005DB710 005DB718 | .........)...).. |
584608       10   0011BB08 08B40909 00584631 00000000 | ..............   |
584618       20   07004120 E0000000 095846C1 20000084 | ..........A...d  |
584628       30   00000000 00000000 014040F5 4B40D9C5 | .........  5. RE |
```

Exercise 2: Partition in I/O Wait

Search in RETAIN for "$ijdprt wait":

```
** SOFTWARE SUPPORT FACILITY    TITLE PAGE       1     **
LIB/FILE(S) CURRENTLY SELECTED DS/AC
$IJDPRT WAIT

THE ABOVE SEARCH ARGUMENT RESULTED IN     2 MATCHES
USER PTF#,APAR#,ABS.L1                                    (APAR DEFAULTS)

    1         DY41428 VSERAS
    2 UD50251 DY44442 SERVICE RELEASE 04/1997 FOR VSE/ESA VERSIO

PROBLEM SUMMARY:                                            ACTIVE
**************************************************************  OPTIONS ARE:
* USERS AFFECTED: All.                                       *
**************************************************************
* PROBLEM DESCRIPTION: 1. SVC07 Softwait with parallel POWER  *
*                      2. ILLEGAL SVC with SVC03 fct code 06  *
*                      3. IJBAR Enhancements:                 *
*                         A. GETVIS SVA,DETAIL                *
*                         B. SIR SMF,cuu                      *
**************************************************************    HIT       2
* RECOMMENDATION:                                            *    OF        2
************************************************************** APAR= DY44442
1. Softwait with an SVC07 (WAIT) in module $ijdprt with Turbo    OWNED BY: RS
   Dispatcher active and VSE/POWER running in parallel mode.
   The dump shows a COBOL program that is waiting for a spooled
   print I/O. The traffic bit in the CCB is not posted, although
   the status looks like I/O complete.
```



Exercise Three: I/O Error

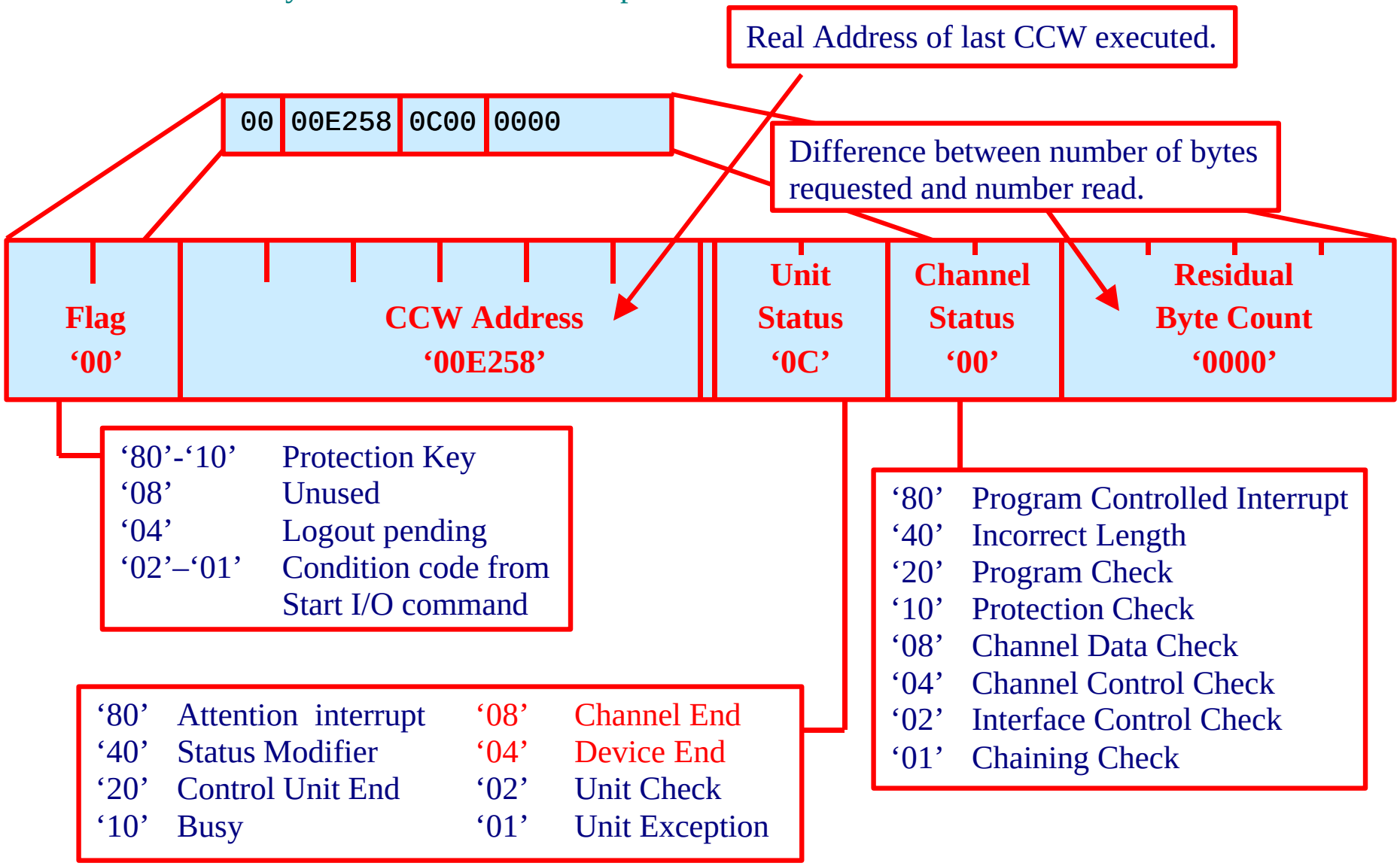
Exercise 3:

1. Interpret hardware error indicators in VSE/ESA console messages.
2. Locate CCB in I/O Error partition dump.
3. Locate failing CCW.
4. Identify dasd address

Is anyone asleep yet?

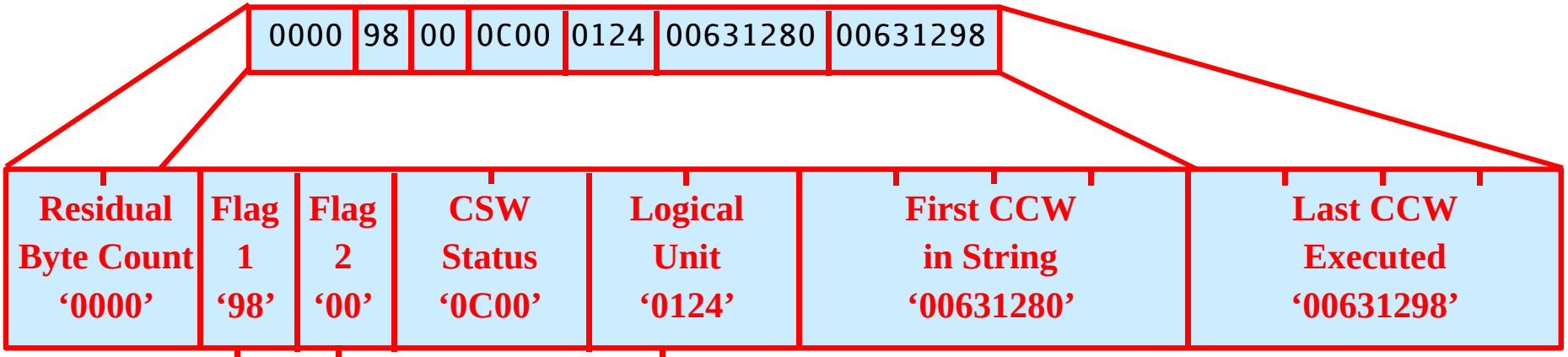
Exercise Three: I/O Error

CSW:(Channel Status Word) passed to hardware by VSE via SSCH command.
Returned by hardware when I/O complete.



Exercise Three: I/O Error

CCB: (Command Control Block) used by application to request Input/Output by VSE)



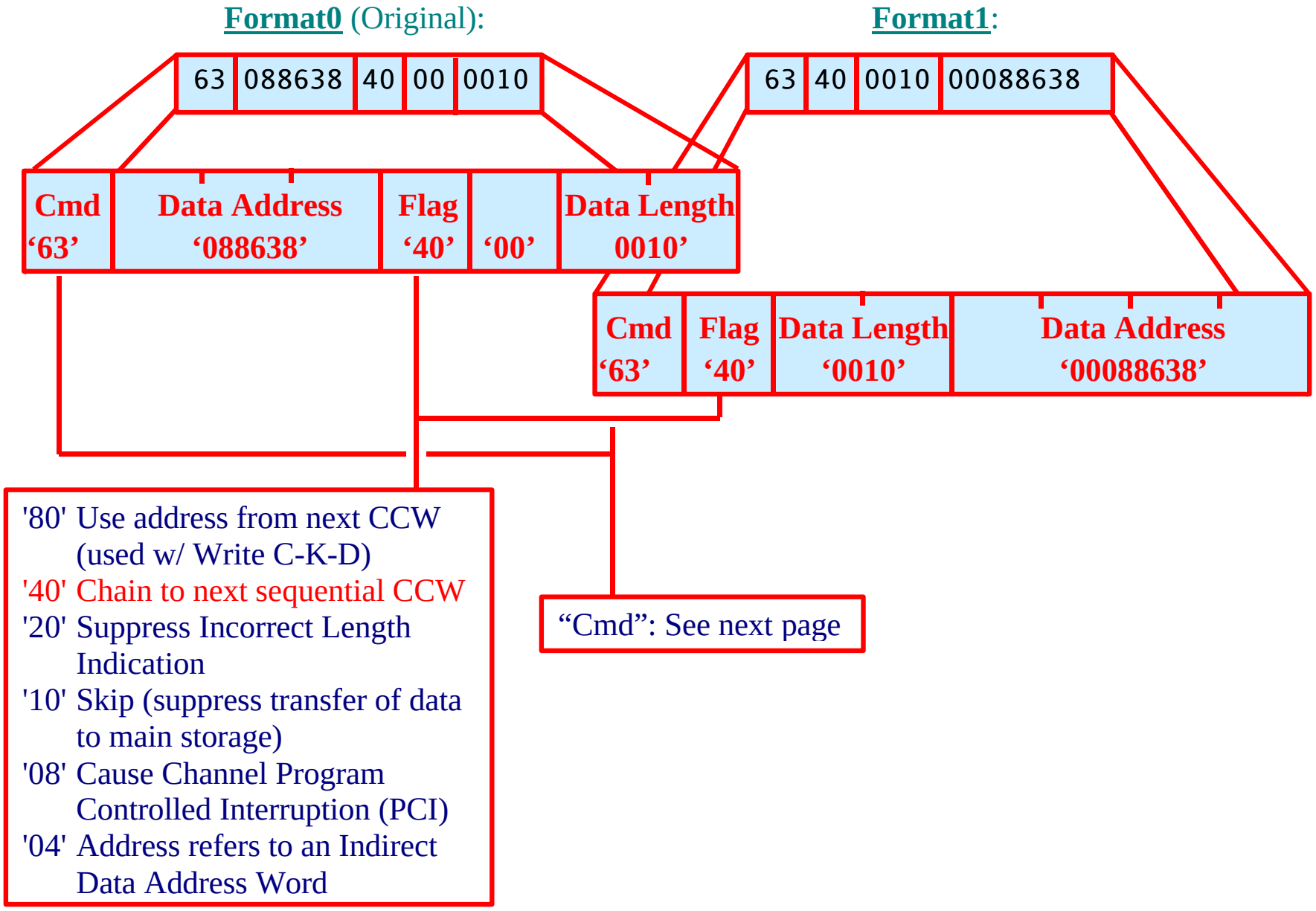
'80' Traffic Bit (Set at Channel End or Device End)
'40' End of File
'20' Unrecoverable I/O Error
'10' Accept Unrecoverable Errors
'08' Return Data Checks
'04' Post at Device End
'02' Return Data or Read Check
'01' User Error Routine

'80' Data Check in count area
'40' Track Overrun
'20' End of Cylinder
'10' Data Check
'08' No Record Found
'04' Retry No-record-found
'02' Verify Error
'01' Command Chain (retry)

x'01xx': Programmer unit (x'0124' = SYS036)
Else: '0000' = SYSRDR, '0001' = SYSIPT,
'0002' = SYSPCH, '0003' = SYSLST,
'0004' = SYSLOG, '0005' = SYSLNK,
'0006' = SYSRES etc.

Exercise Three: I/O Error

CCW: (Channel Command Word), specify I/O commands



Exercise Three: I/O Error

CCW Command Codes commonly-used for DASD:

Count-Key-Data (CKD):

```
07  Seek
1B  Seek Head
23  Set Sector
31  Search ID Equal
08  TIC
06  Read Data
86  Multi-track read
05  Write Data
```

Extended-Count-Key-Data (ECKD):

```
63  Define Extent
47  Locate Record

06  Read Data
86  Multi-track read
05  Write Data
```

CCW Op-codes commonly used for printers:

```
09  Space one line after printing
0B  Space one line immediately
63  Load FCB
89  Skip to channel one after printing
8B  Skip to channel one immediately
```

CCW Op-codes commonly used for tape:

```
01  Write                 2F  Backspace file
02  Read forward          3F  Forward space file
07  Rewind                9F  Lead display
0F  Rewind and Unload     DB  Modeset
```

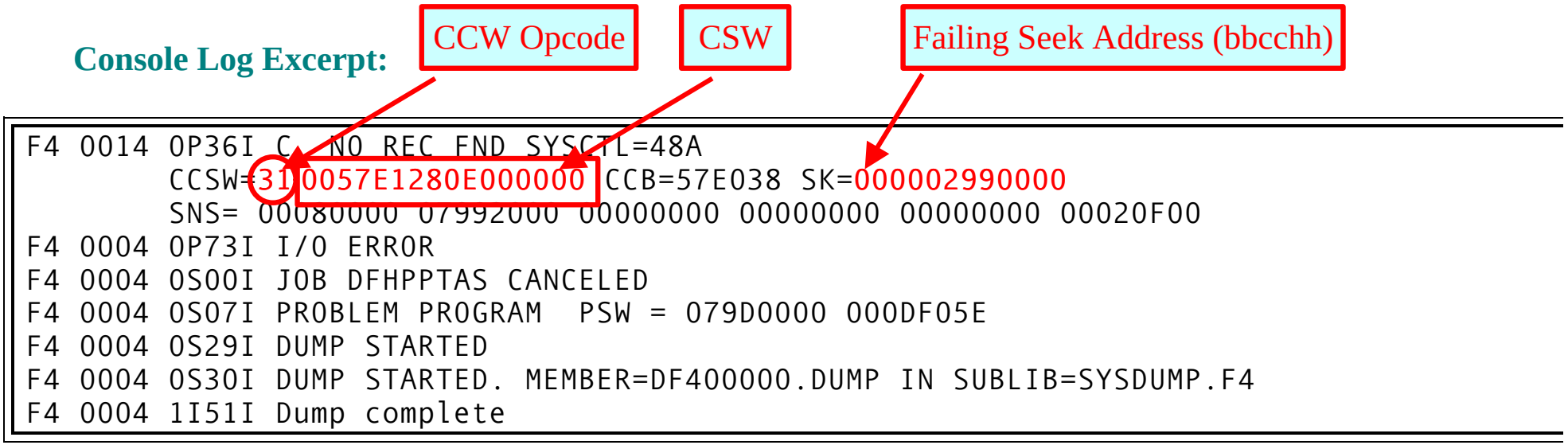
Sense CCW: 04

Exercise Three: I/O Error

Problem Description:

Message 0P36I during compilation of DFHPPT after migration to VSE/ESA 2.1.3.

Console Log Excerpt:



Input/Output errors are documented via a message to the console, 0P01 – 0P68 or 0Exx, and the application is cancelled, unless the application has requested that I/O errors be returned.
(CCB+x'02' = x'10', x'08', or x'02')

1. Check message in *VSE/ESA Messages and Codes, Volume 1*. If the message indicates a critical I/O error, contact hardware support. Many of these kinds of problems are caused by hardware.

Exercise Three: I/O Error

2. The 0Pxx message will identify the failing hardware unit. However, not all hardware errors are logged via EREP. If VSE is running on VM, it passes most error information to VM for logging. “No Record Found” (msg0P36I), “Phase not found” (msg0P62I) or “Wrong Length Record” errors are not logged.
3. Even if a dump is not created (“// OPTION NODUMP”), message 0S07I will include the failing PSW. Ascertain which phase was involved. If the phase was in the SVA, use the “**SHOW**” command to locate an eyecatcher.
4. Errors associated with file management products (LIBR, VSAM, DL/I) are often caused by file corruption. Run the following utilities:
   - LIBR: ($IJBLBR, modules start with INL*)

     Run LIBR TEST against the failing library.
   - VSAM: (modules start with IKQ*)

     Run IDCAMS REPRO against the failing file.

     (modules start with IGG*)

     Run IKQVCHK against the failing catalog.
   - DL/I: (modules start with DLZ*)

     Run DLZURGP0 to unload the database,

     IKQVCHK, to check the catalog

Exercise Three: I/O Error

CCB example: (Define Extent / Locate Record)

```
V00667B80 00001400 000005C7 00667840 01000000 B0 *.......G... ....*
V00667B90 00000000 00000000 00000000 00667BC0 B0 *.............#.*
```

First CCW in

```
V00667840 63400010 00667BC0 47400010 00667BD0 B0 *. ....#.. ....#.*
V00667850 85002000 011CCA00 03000038 00000000 B0 *................*
V00667860 00000000 00000000 00000000 00000000 B0 *................*
```

Define Extent Block

Extent definition (cchh – cchh):
From Cyl x'13', hd 0, to Cyl x'1C' Hd 14

```
V00667BC0 98C02000 00000040 00130000 001C000E B0 *...... ........*
```

Locate Record Block

Record being read (cchhr):
Cyl x'13', head 1, Record 3

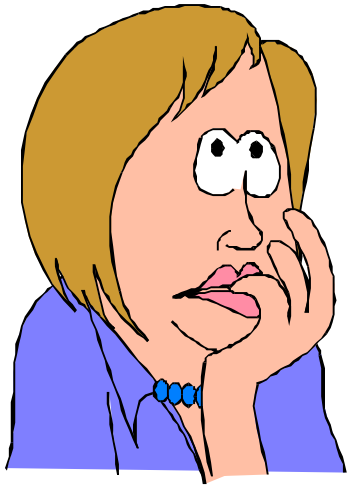
```
V00667BD0 01800001 00130001 00130001 03482000 B0 *................*
V00667BE0 00000000 13000104 00000100 00000000 B0 *................*
```

Exercise Three: I/O Error

```
// JOB DMPANA33  PRINT COMPLETE FORMATTED DUMP                    DATE 10/21/97,CLOCK 14/55/52
// EXEC PROC=DTRINFOA
// ASSGN SYS016,DISK,VOL=SYSWK1,SHR       INFO ANAL MANAGEMENT FILE
1T20I  SYS016 HAS BEEN ASSIGNED TO X'141' (TEMP)
// ASSGN SYS017,DISK,VOL=SYSWK1,SHR       INFO ANAL ROUTINES FILE
1T20I  SYS017 HAS BEEN ASSIGNED TO X'141' (TEMP)
EOP DTRINFOA
// EXEC INFOANA,SIZE=300K

BLN9018I DUMP SYSDUMP.F4.DF400000 SELECTED
         RETURN
   SELECT DUMP VIEWING
         PRINT FORMAT
      .
      .
      .
NAME = CANCLMSG COMPONENT ID = F4       TYPE = TEXT DATA

OP73I I/O ERROR
OS00I JOB DFHPPTAS CANCELED
      .
      .
      .
NAME = GREGS    COMPONENT ID = F4       TYPE = HEXADECIMAL DATA

    GP REGS    0-3    00000054 0057E038 0057E000 0057E038
               4-7    FFFFFFFF 000DF0CC 0058F380 00262028
               8-B    0040A190 0040A044 000DBE78 000E002F
               C-F    000DF030 0058F9B0 000DAA90 000DF030

      .
      .
      .
NAME = PSW      COMPONENT ID = F4       TYPE = HEXADECIMAL DATA
       PSW     079D0000 000DF05E
      .
      .
      .
```

Reg1 ⇒ CCB

Failing instruction



Exercise Three: I/O Error

```
NAME=$IJBLBR  COMPONENT ID=SVA      BASE=000BC198
 000BC180                                                          47F0F046 5BC9D1C2 00 *                        .00.$IJB
 000BC1A0  D3C2D940 C4C2F6F0 C9C9F0F0 C4F6F1D5 C4E4E3F5 F6F8F660 C0F6F640 4DC35D40 00 *LBR DB60II00D61NDUT5686-066 (C)
 000BC1C0  C3D6D7E8 D9C9C7C8 E340C9C2 D440C3D6 D9D74B40 F1F9F8F4 6B40F1F9 F9F541F0 00 *COPYRIGHT IBM CORP. 1984, 1995.0
 000BC1E0  F1601EF0 58F0F000 07FF0000 0000                                          0 *1-.0.00..........F..........?6
 000BC200  000BE848 000BEB60 000C3100 000C                                          0 *..Y....-......(.......1.........
 000BC220  000CA460 000CAD98 000CE890 000CF068 000CF478 000D6FD8 000CF9A0 000D6220 00 *...-......Y...0...4...?Q..9.....
 000BC240  000DF030 000DF0D0 000DFB18 000E0E48 000E1C00 000E2008 000E6238 000E7690 00 *..0...0.....................
 000BC260  000E5DC8 000E80D8 000F2188 000F87E8 000FB3F0 00000000 C9D5D3C3 D7E3C3C8 00 *..)H...Q.......Y...0....INLCPTCH
 000BC280  40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 00 *
 000BC2A0  TO NEXT LINE ADDRESS SAME AS ABOVE
      •
      •
      •
 000DF000  00600000 01630000 00600000 10470000 00600000 101D0000 00200000 089D0000 00 *.-.......-.......-.............
 000DF020  00200000 08FFFFFF FF000000 00000000 47F0F01C 17C9D5D3 D7C4D6C9 D6F5F6F8 00 *.................00..INLPDOI0568
 000DF040  F6F6F1C9 F9F0F8F3 F1F8F4F5 90ECD00C 18CF41B0 CFFF4150 C09C1813 0A009180 00 *661II0831845..........&........
 000DF060  10024710 C0380A07 1F001FEE 41F00047 0A6B1FFF 58E0D00C 980CD014 07FED7C1 00 *.............0...,............PA
 000DF080  E3C3C840 C1D9C5C1 406040C9 D5D3D7C4 D6C9D640 F9F54BF0 F8F3C06A C06CC06E 00 *TCH AREA - INLPDOI0 95.083...%.>
      •
      •
      •
BLN5014I DATA FROM 00268D78 TO 0026FD77 NOT AVAILABLE OR ALL ZEROS
BLN9012I PRINT FUNCTION COMPLETED
          PRINT 0 END
      •
      •
      •
```

Reg12 points here (begin of module)

Branch instructions shows Reg12 to be base reg

PSW pointed here

Exercise Three: I/O Error

```
00400000  D3D5D2C5 C4E34040 079D0000 000DF05E 0040A044 000DBE78 000E002F 000DF030 90 *LNKEDT  ......0;. ............0.
00400020  0058F9B0 000DAA90 000DF030 00000054 0057E038 0057E000 0057E038 FFFFFFFF 90 *..9.......0.....................
00400040  000DF0CC 0058F380 00262028 0040A190 0000AF67 5EA86815 40404040 40404040 90 *..0...3...... ......;...
00400060  40404040 40404040 40404040 40404040 40404040 40404040 C9D1C2D3 C5F14040 90 *                        IJBLE1
00400080  F1F5C3F0 40F2F140 C4F6F1E9 C4C8E240 F5F6F8F6 60F0F6F6 404DC35D 40C3D6D7 90 *15C0 21 D61ZDHS 5686-066 (C) COP
004000A0  E8D9C9C7 C8E340C9 C2D440C3 D6D9D74B 40F1F9F7 F96B40F1 F9F9F600 00000000 90 *YRIGHT IBM CORP. 1979, 1996.....
004000C0  D540C1E4 E3D64040 00000004 00423304 00423304 C4C6C8D7 D7E3C1E2 07000000 90 *N AUTO  ............DFHPPTAS....
004000E0  00000000 00000000 00000000 00400078 00428090 00000000 00000000 00004000 90 *.............. ............... .
00400100  00000000 C4C6C8D7 D7E3C2C1 01000040 00000003 00000000 00000000 00000000 90 *....DFHPPTBA... ................
```

-
-
-

Reg1 pointed here (CCB)

```
0057E000  C2E4C3C2 0057E110 00000017 00000006 00000400 00000000 00000000 0057E250 90 *BUCB.........................S&
0057E020  0057E250 0057E250 0057E290 0057E250 0057E1D0 00000000 0000A00C 0E000831 90 *..S&..S&..S...S&.............
0057E040  0057E110 0057E128 00000000 005F0057 E0B00000 00000000 00000000 00000000 90 *.............................
0057E060  00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 90 *.............................
0057E080  TO NEXT LINE ADDRESS SAME AS ABOVE
0057E100  00000000 00000000 00000000 00000000 0757E25E 60000006 2358FB48 60000001 90 *.................S;-.......-...
0057E120  3157E260 60000005 0857E120 60000001 8657F020 20000400 00000000 00000000 90 *..S--.......-.....0...........
0057E140  00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 90 *.............................
0057E160  TO NEXT LINE ADDRESS SAME AS ABOVE
0057E240  FFFFFFFF FFFFFFFF FFFFFFFF 0057F420 C2C8C4D9 00000003 0C600000 00510000 90 *.............4.BHDR.....-......
0057E260  02990000 01000000 00007F26 FFFFFFFF 00265DD8 FFFFFFFF FFFFFFFF FFFFFFFF 90 *.........".......)Q...........
0057E280  FFFFFFFF FFFFFFFF FFFFFFFF 0057F020 C2C8C4D9 00000004 00000000 00000000 90 *.............0.BHDR...........
```

Exercise Three: I/O Error

Summary:

1. PSW points at DF05E. (INLPDOIO + X'2E'). Last instruction executed was I/O wait (SVC 07). PSW has been backed up 10 bytes:

```
9180 1002       TM   CCBFLAG,POSTED
4710 C038       BO   *+6
0A07            SVC  07
```

2. Register 1 points at the CCB (0057E038) = `0000A00C 0E000831 0057E110 0057E128`
3. CSW Status Bytes ('0E00') indicate Channel End + Device End + Unit Check

   CCB communications bytes (`'A00C'`) indicate Posted + Unrecoverable I/O error + No Record Found.

   I/O was performed to system device x'31' (SYSCTL). Msg0P36I identified SYSCTL as being assigned to device '48A'.

   CCW chain starts at 57E110, and last CCW executed was at 57E120 (57E128 – 8)
4. CCW chain at 57E110:

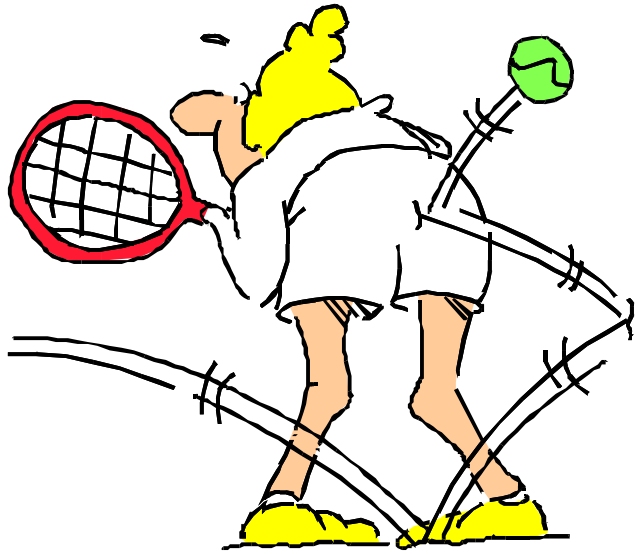
`0757E25E 60000006`	Seek to cylinder x'299' (665) {bbcc = 00000299}
`2358FB48 60000001`	Set Sector to 04 (not in handout)
`3157E260 60000005`	Search to cylinder x'299', head 0, record 1 {cchhr = 0299000001}
`0857E120 60000001`	TIC back to previous search (keeps channel program active until head is positioned properly)
`8657F020 20000400`	Multi-track read. 1024 (x'400') byte record into buffer at x'57F020'.

5. CCW chain was terminated prematurely at search with "no record found".

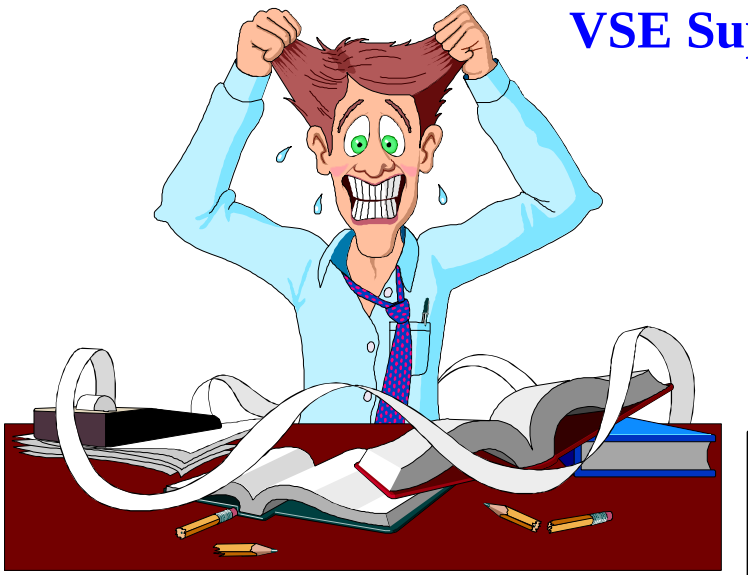
Exercise Three: I/O Error

Problem Resolution:

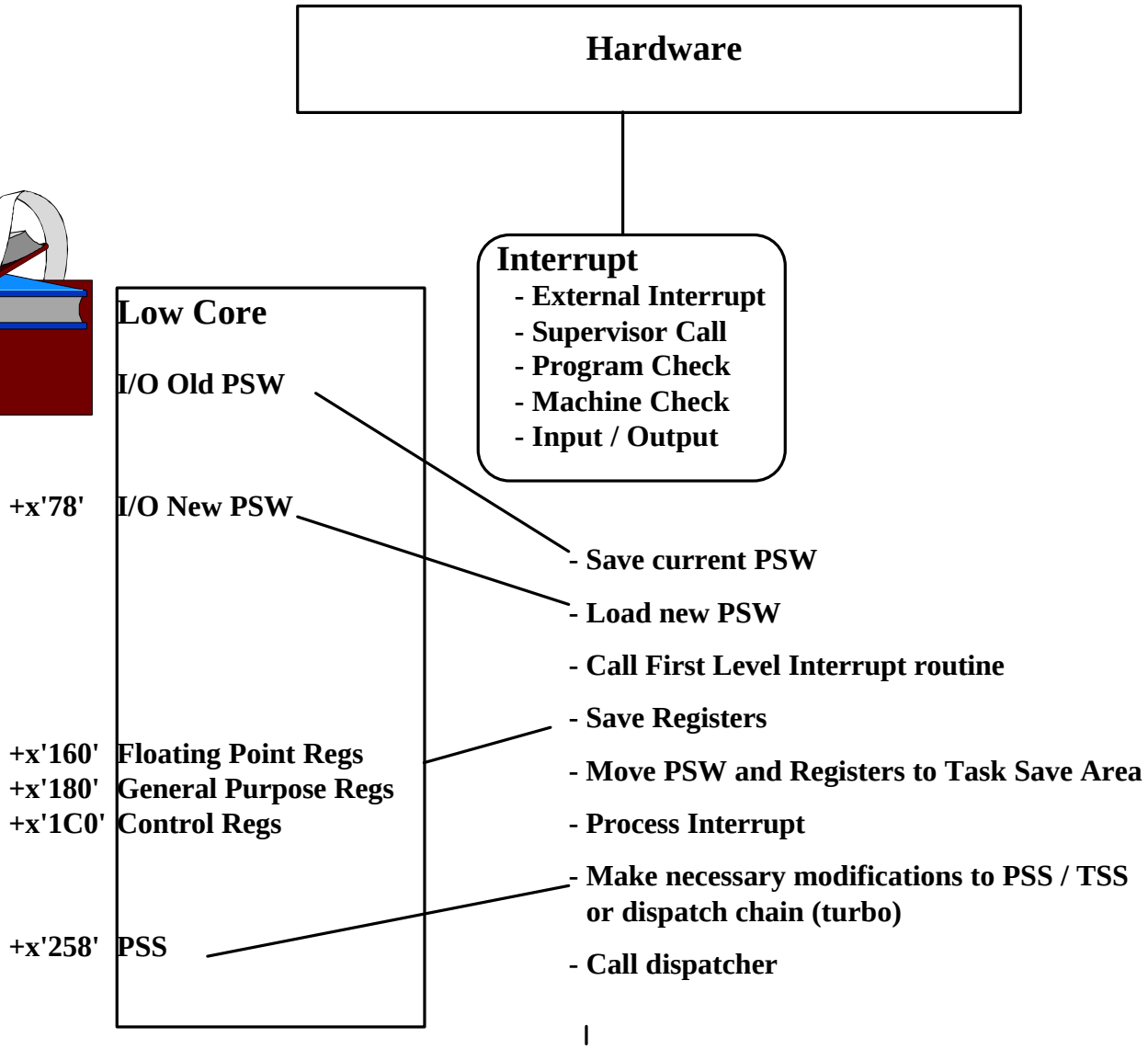
To get around the immediate problem, the library was re-built. Cylinder 665 of device x'48A' was part of the VSE library. However, when VSE/DITTO was used to display this area, it showed all records (including record 1) missing from this track (and subsequent tracks for several cylinders). Turning on SYS DASDFP=YES during IPL, identified a product initializing space outside of its defined extents.



VSE Supervisor Control Blocks

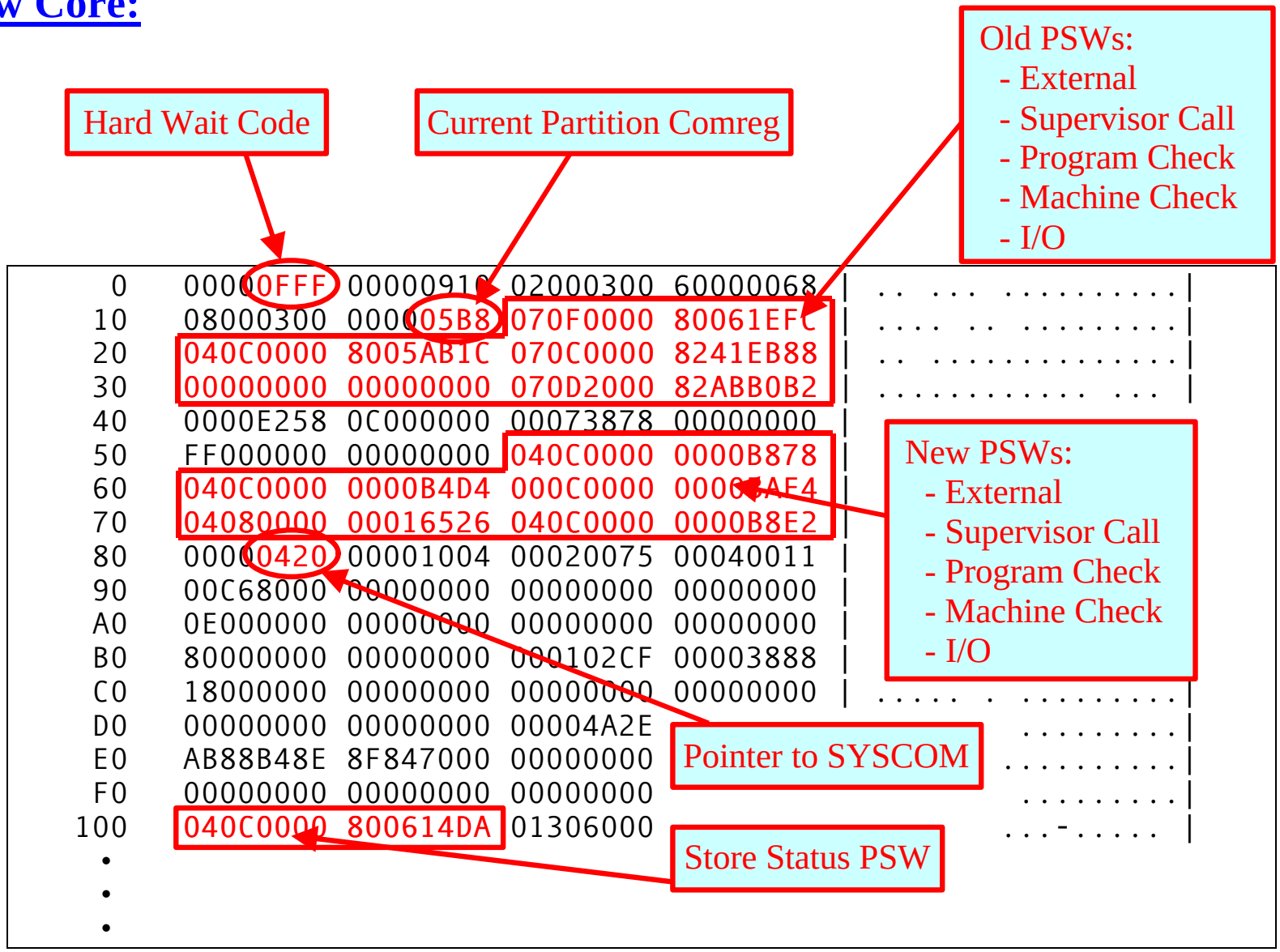


Uh oh, now we're really getting in deep!

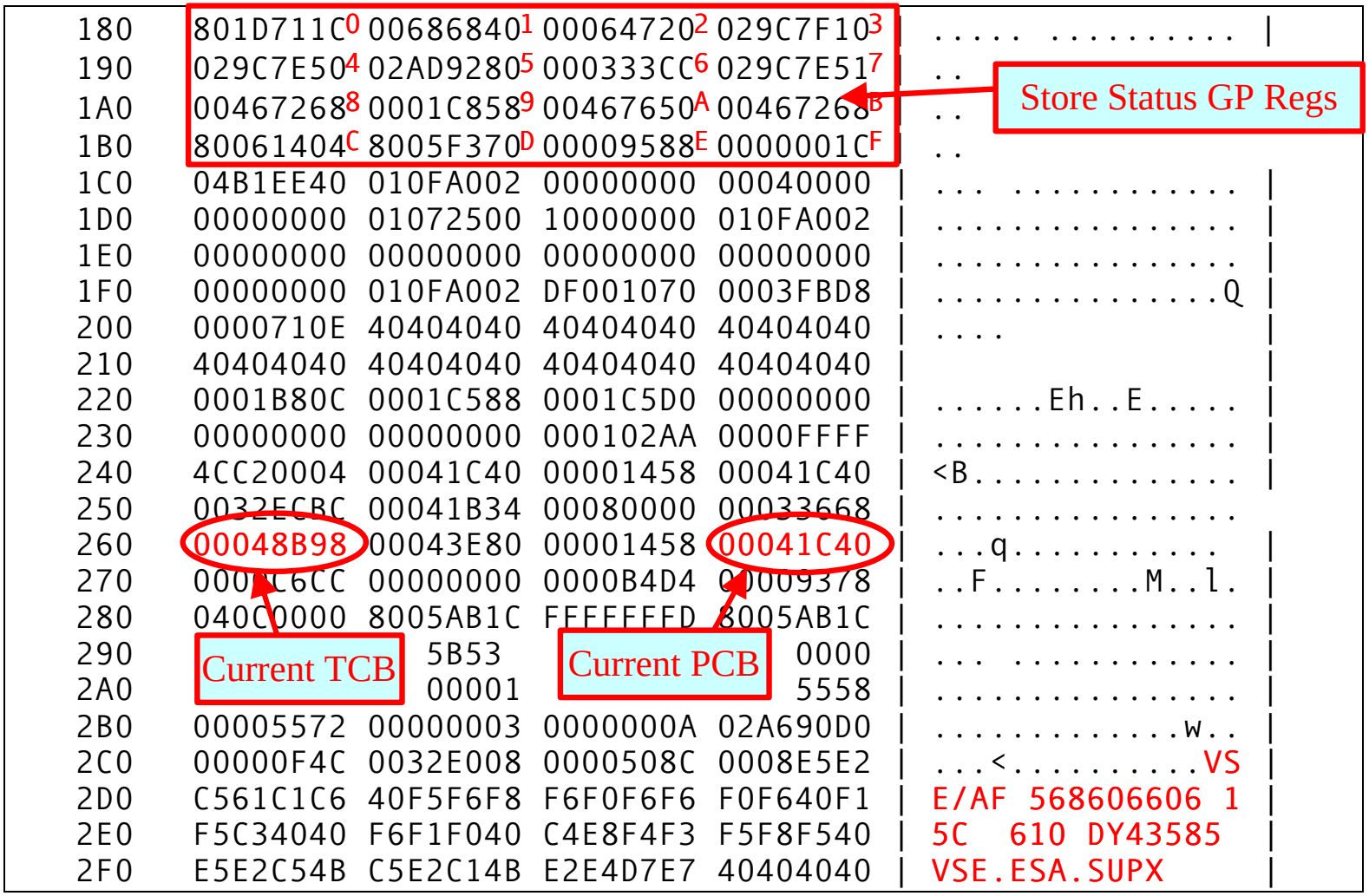


VSE Supervisor Control Blocks

Low Core:



VSE Supervisor Control Blocks



VSE Supervisor Control Blocks

Low Core:

+x'00' Hard Wait code (4 bytes) (Also in SYSCOM +x'04')
+x'14' Partition Comreg pointer for active partition

Old PSWs:

+x'18' External Old PSW
+x'20' Supervisor Call Old PSW
+x'28' Program Check Old PSW
+x'30' Machine Check Old PSW
+x'38' I/O Old PSW

+x'40' Channel Status Word (CSW)
+x'48' Channel Address Word (CAW)

New PSWs:

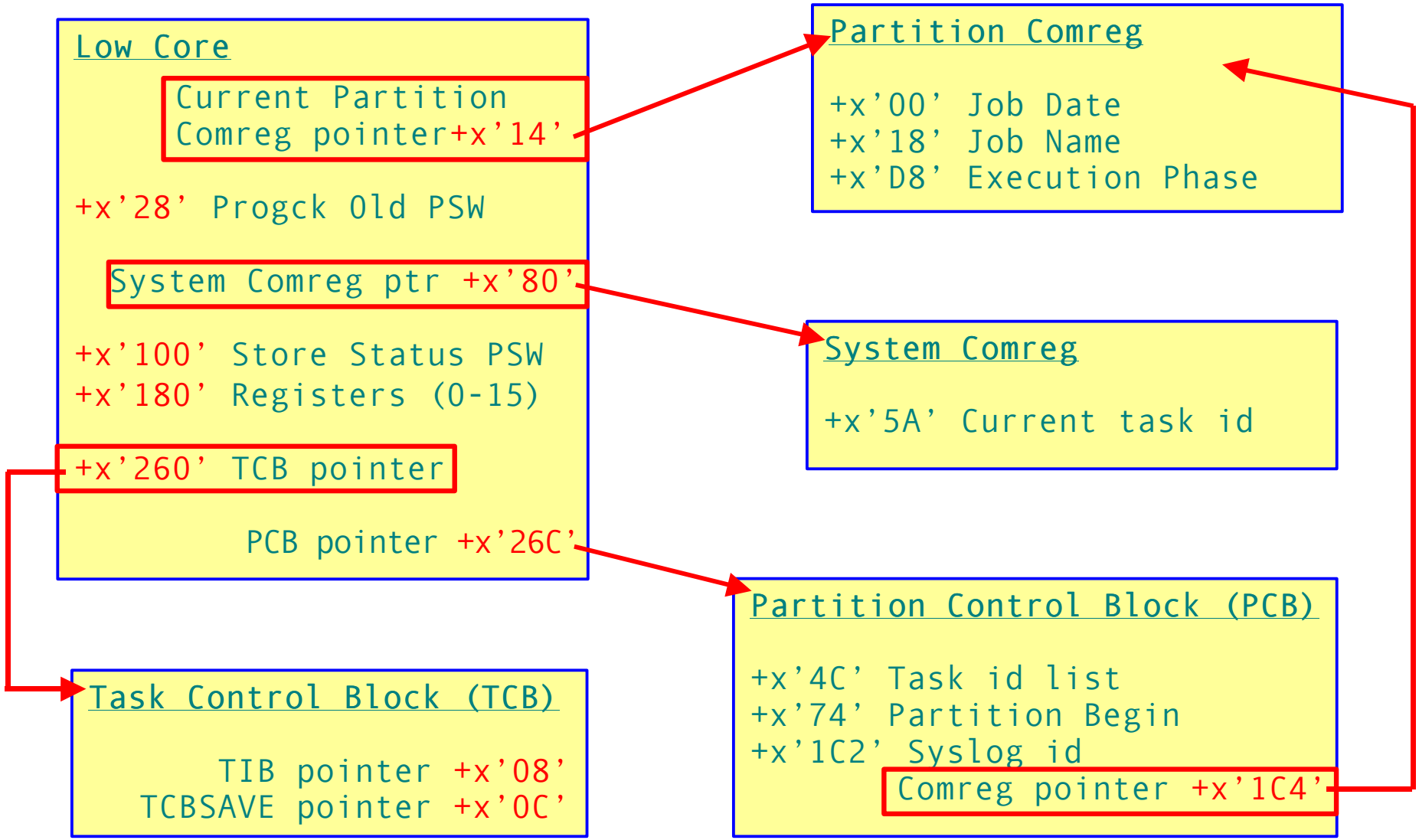
+x'58' External New PSW
+x'60' Supervisor Call New PSW
+x'68' Program Check New PSW
+x'70' Machine Check New PSW
+x'78' I/O New PSW

+x'80' Pointer to SYSCOM (always 00000420 for VSE/ESA version 2)

VSE Supervisor Control Blocks

+x'8A'	Last SVC interrupt code
+x'8C'	Last Program Check interrupt info (ILC + interrupt code)
+x'BA'	Last I/O Interrupt address (moved to x'23E' when hard wait set)
+x'100'	**Store Status PSW** (If no Store Status was performed, search for "SNS-TASK" and back up x'48' bytes)
+x'180'	**Store Status General Purpose Registers**
+x'1C0'	Store Status Control Registers
+x'23E'	Last I/O Interrupt address prior to hard wait (moved from x'BA')
+x'241'	x'10' indicates that the turbo dispatcher controls more than 1 CPU.
+x'258'	Partition Selection String (PSS)
+x'25C'	Address of related CPU control block (turbo dispatcher)
+x'260'	**Address of currently active Task Control Block (TCB)**
+x'264'	Address of currently active Task Information Block (TIB)
+x'268'	Address of currently active Partition Information Block Extension (PIB2)
+x'26C'	**Address of currently active Partition Control Block (PCB)**
+x'270'	Address of DEBUG control header
+x'280'	Save area for General Purpose Registers (9-8)
+x'2C0'	Address of TIB table (TIBATAB)
+x'2C4'	Address of PCB Table (PCBATAB)
+x'2CE'	**Supervisor Level Identification information**

VSE Supervisor Control Blocks



VSE Supervisor Control Blocks

Task Control Block (TCB)

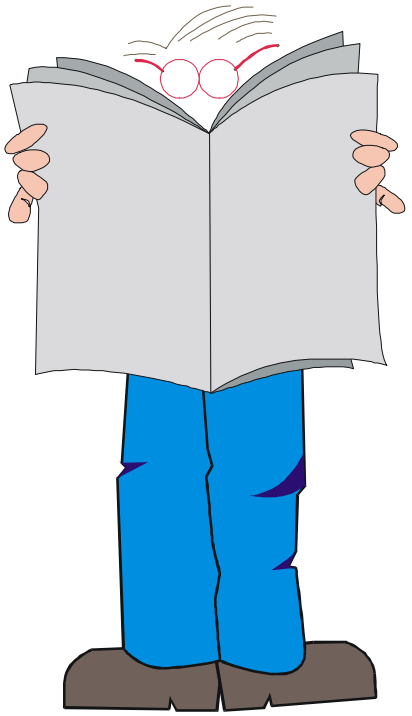
TIB pointer +x'08'
TCBSAVE pointer +x'0C'

Task Information Block (TIB)

+x'14' Task id
+x'2C' Task status

TCBSAVE

+x'00' Phase name
+x'08' PSW
+x'10' Registers 9-8

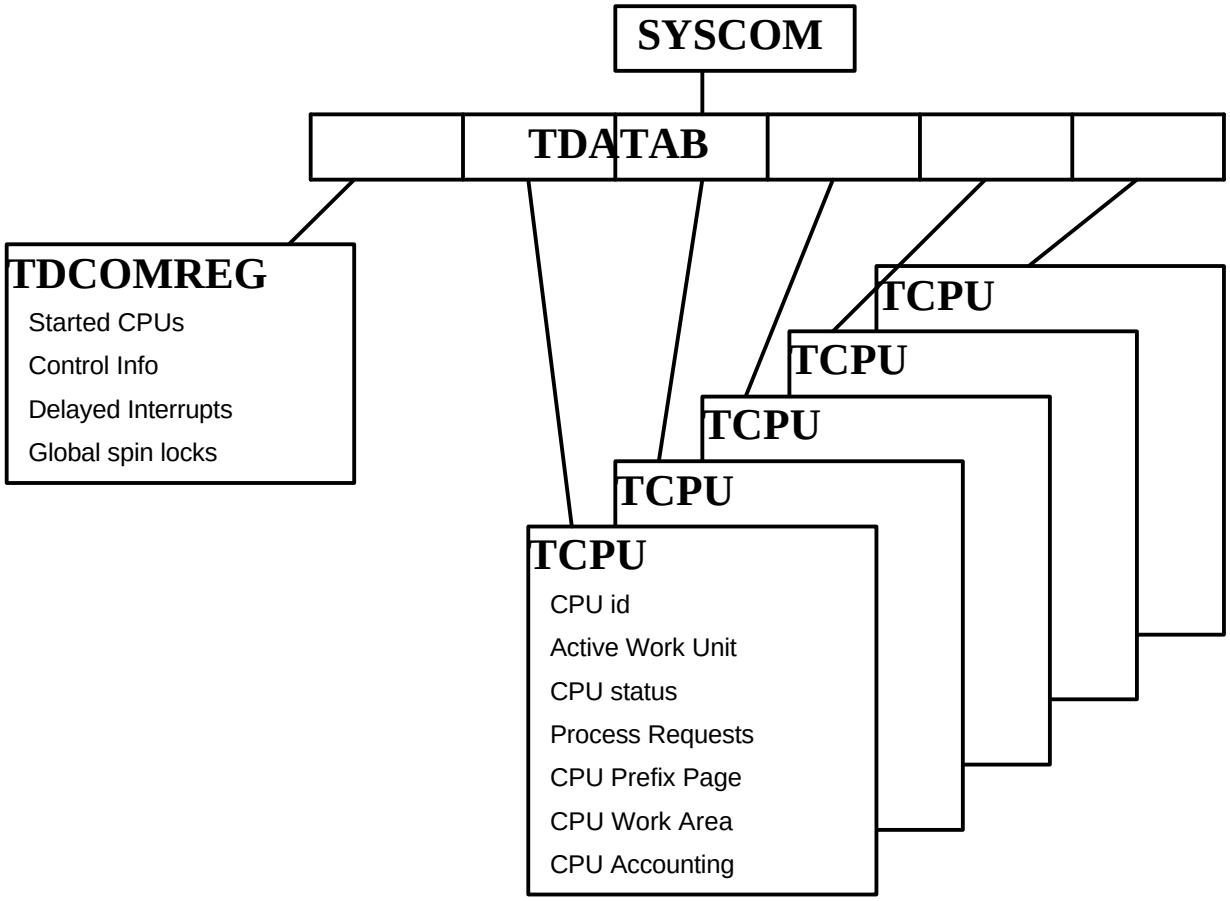


VSE Supervisor Control Blocks

Turbo Dispatcher:

When running in multi-processor mode, there is one more step. Each processor has its own low core, and system errors are only documented in the low core page (page zero) which was active at the time when the error was detected.

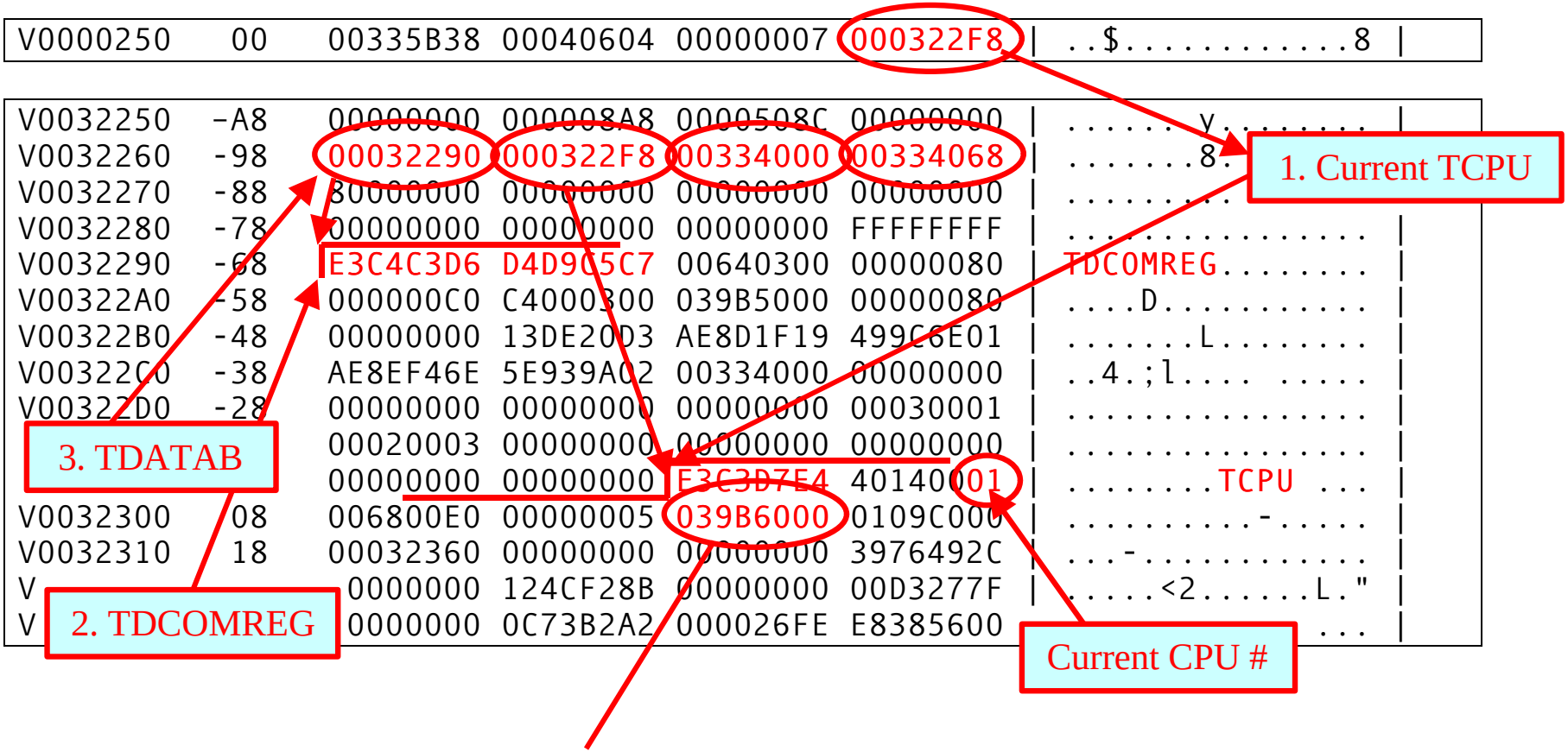
Low core is used to communicate between hardware and software. VSE control blocks which may be changed by all active processors are also saved in page zero. Each active processor is allocated a unique 4k page (page zero). All software changes to one page zero must be synchronized in all other copies. Therefore, these changes may only be made by one CPU at a time (non-parallel work unit). Control blocks and area which can be changed independent on the status of the system have been moved from page 0 (prefix page). This allows these control blocks to be updated as a parallel work unit. Since they are the anchor on which many other control blocks are hung, SYSCOM and BG Comreg have been left in page 0. Turbo Dispatcher uses a locking (test-and-set) mechanism to ensure that only one CPU at a time can update control blocks in page 0.



VSE Supervisor Control Blocks

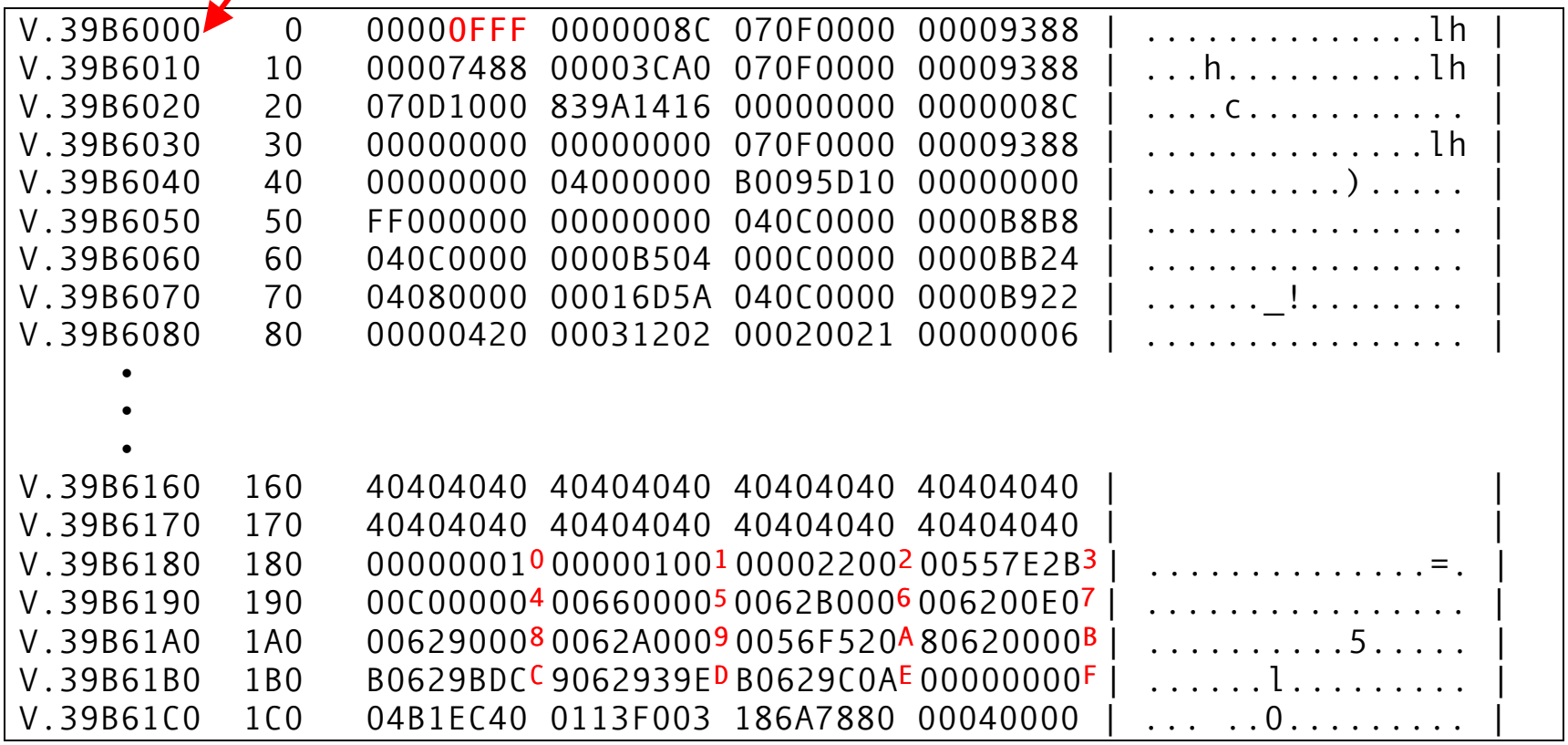
Locate all copies of page zero:

1. Current TCPU: Follow pointer in lowcore + x‘25C’ to current TCPU.
2. TDCOMREG: Approx x’68’ prior to TCPU(1), is the TDCOMREG, which is the main Turbo Dispatcher control block.
3. TDATAB: Just prior to that you will find TDATAB (a list of TCPUs for all active processors). The list starts with a pointer to the TDCOMREG, and ends with x’80000000’.



VSE Supervisor Control Blocks

4. TCPU: There is a TCPU control block for each active processor. TCPU+x'07' contains the CPU number. TCPU+x'10' contains a pointer to the prefix page (low storage page 0) for this processor. Prefix pages for all active processors are contiguous in 31-bit storage, so once you have found one, you can find them all.



```
V.39B6000    0   0000[0]FFF 0000008C 070F0000 00009388 | ..............lh |
V.39B6010   10   00007488 00003CA0 070F0000 00009388 | ...h.........lh |
V.39B6020   20   070D1000 839A1416 00000000 0000008C | ....c........... |
V.39B6030   30   00000000 00000000 070F0000 00009388 | ..............lh |
V.39B6040   40   00000000 04000000 B0095D10 00000000 | .........)..... |
V.39B6050   50   FF000000 00000000 040C0000 0000B8B8 | ................ |
V.39B6060   60   040C0000 0000B504 000C0000 0000BB24 | ................ |
V.39B6070   70   04080000 00016D5A 040C0000 0000B922 | ......_!........ |
V.39B6080   80   00000420 00031202 00020021 00000006 | ................ |
      •
      •
      •
V.39B6160  160   40404040 40404040 40404040 40404040 |                  |
V.39B6170  170   40404040 40404040 40404040 40404040 |                  |
V.39B6180  180   00000001[0] 00001001[1] 00002200[2] 00557E2B[3] | ............=. |
V.39B6190  190   00C00000[4] 00660000[5] 0062B000[6] 006200E0[7] | ................ |
V.39B61A0  1A0   00629000[8] 0062A000[9] 0056F520[A] 80620000[B] | .........5..... |
V.39B61B0  1B0   B0629BDC[C] 9062939E[D] B0629C0A[E] 00000000[F] | ......l........ |
V.39B61C0  1C0   04B1EC40 0113F003 186A7880 00040000 | ... ..0........ |
```

Exercise 4: System Disabled Wait

Exercise 4:

From a non-formatted Stand-alone dump:

1. Identify partition (task) currently in control at time of system hang.
2. Identify hard wait code (either low core or SYSCOM +x'04')
3. Locate PSW / Registers saved with "**STORE STATUS**" command.
4. Identify module in control at point of failure.

Problem Description:

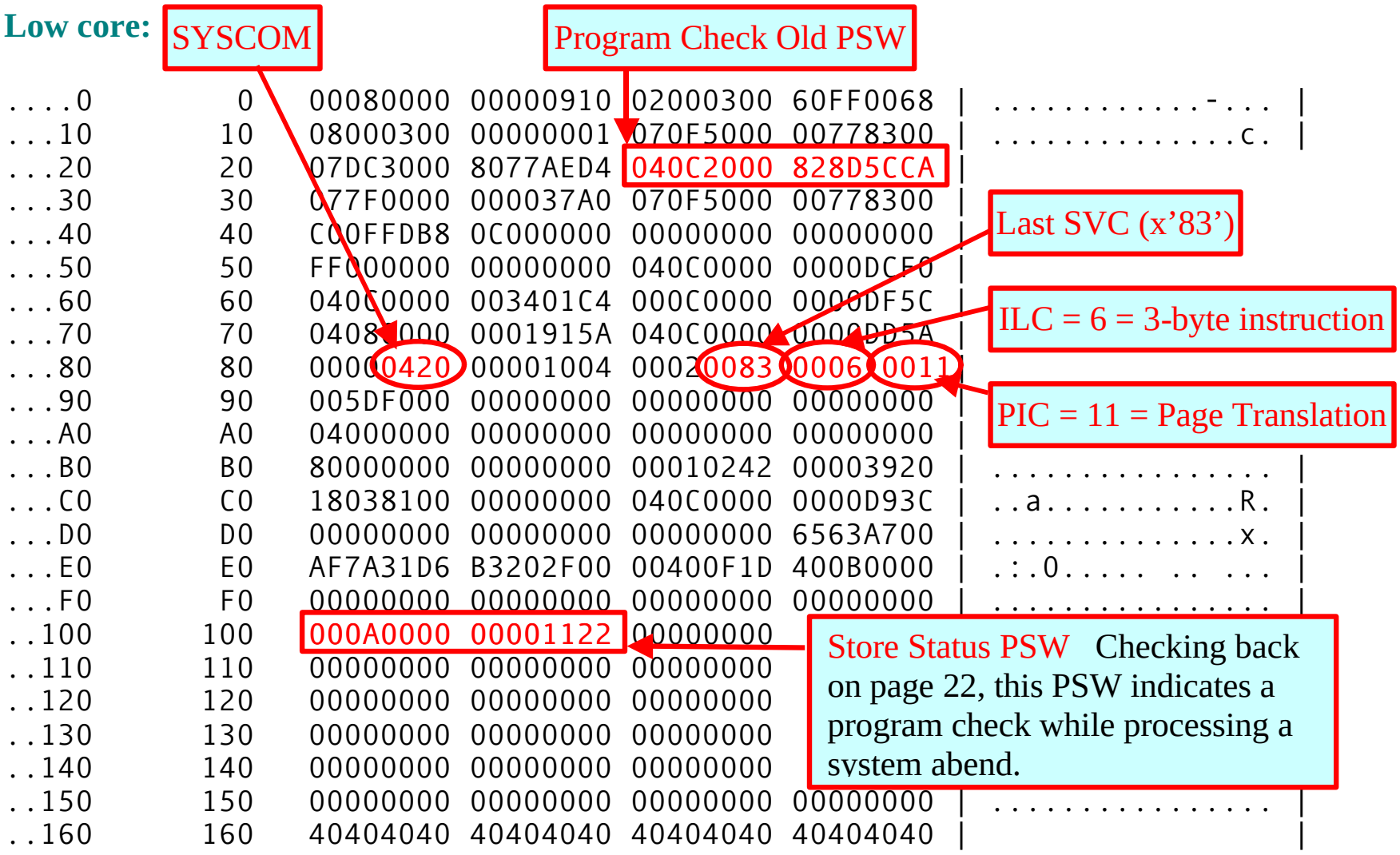
Pat was working with Level2 VSE. They had him issue a `GETVIS SVA,ALL` command and it hung up his system. Pat is taking a SADUMP and will send it in.

Unable to format stand-alone dump using Interactive Interface "*Analyze SA Dump*" function.

Exercise 4: System Disabled Wait

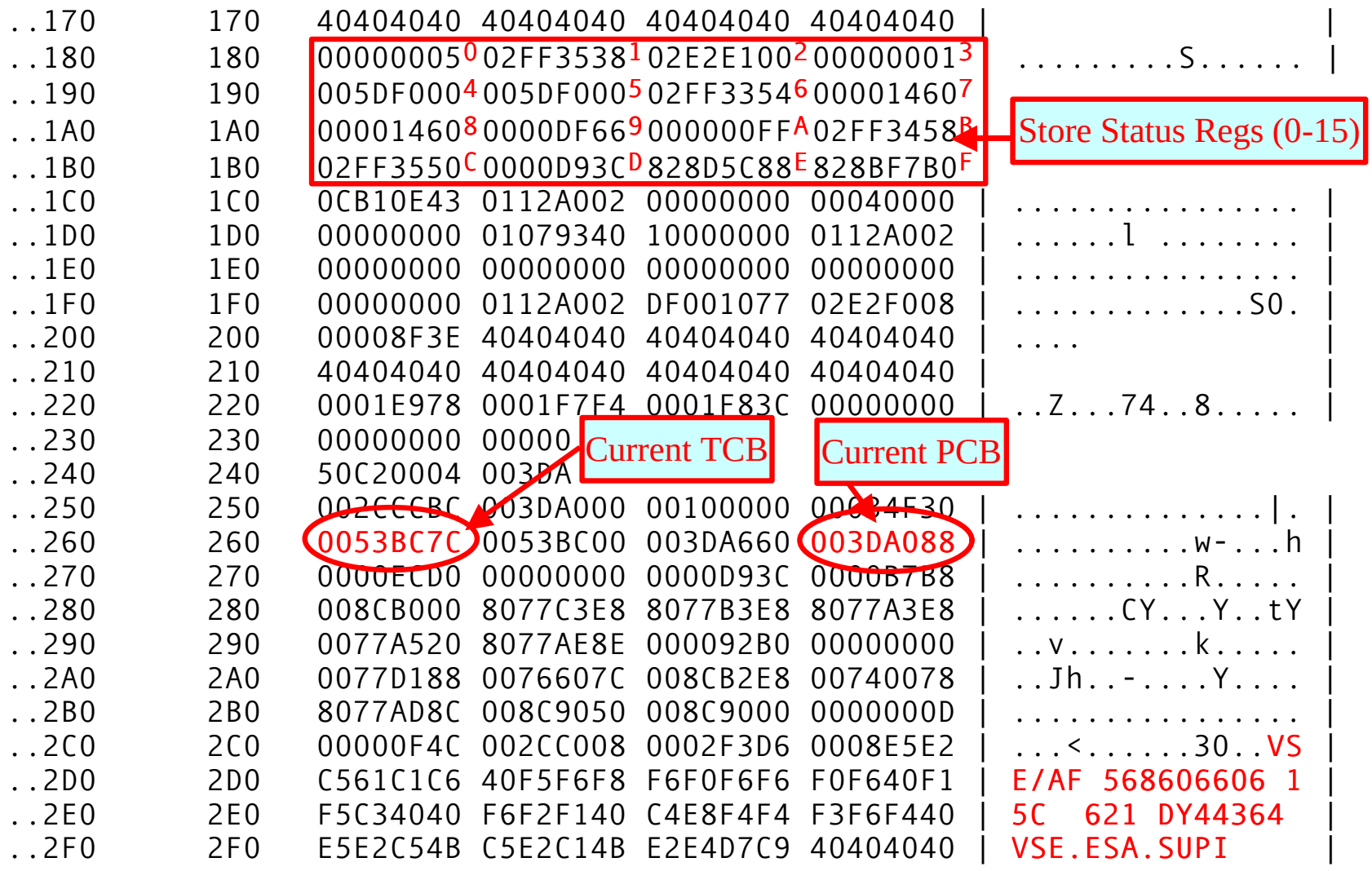
Stand-alone dump analysis failed. So we are left to our own devices

Low core:




```
....0         0   00080000 00000910 02000300 60FF0068 | ............-... |
...10        10   08000300 00000001 070F5000 00778300 | ..............c. |
...20        20   07DC3000 8077AED4 040C2000 828D5CCA |
...30        30   077F0000 000037A0 070F5000 00778300 |
...40        40   C00FFDB8 0C000000 00000000 00000000 |
...50        50   FF000000 00000000 040C0000 0000DCF0 |
...60        60   040C0000 003401C4 000C0000 0000DF5C |
...70        70   04080000 0001915A 040C0000 0000DD5A |
...80        80   00000420 00001004 00020083 00060011 |
...90        90   005DF000 00000000 00000000 00000000 |
...A0        A0   04000000 00000000 00000000 00000000 |
...B0        B0   80000000 00000000 00010242 00003920 | ................ |
...C0        C0   18038100 00000000 040C0000 0000D93C | ..a...........R. |
...D0        D0   00000000 00000000 00000000 6563A700 | ..............x. |
...E0        E0   AF7A31D6 B3202F00 00400F1D 400B0000 | .:.0.... .. ... |
...F0        F0   00000000 00000000 00000000 00000000 | ................ |
..100       100   000A0000 00001122 00000000
..110       110   00000000 00000000 00000000
..120       120   00000000 00000000 00000000
..130       130   00000000 00000000 00000000
..140       140   00000000 00000000 00000000
..150       150   00000000 00000000 00000000 00000000 | ................ |
..160       160   40404040 40404040 40404040 40404040 |                  |
```

Exercise 4: System Disabled Wait




```
..170     170   40404040 40404040 40404040 40404040 |                    |
..180     180   00000005 02FF3538 02E2E100 00000001 |  .........S.....   |
..190     190   005DF000 005DF000 02FF3354 00001460 |                    |
..1A0     1A0   00001460 0000DF66 000000FF 02FF3458 |                    |
..1B0     1B0   02FF3550 0000D93C 828D5C88 828BF7B0 |                    |
..1C0     1C0   0CB10E43 0112A002 00000000 00040000 | ................   |
..1D0     1D0   00000000 01079340 10000000 0112A002 | ......l ........   |
..1E0     1E0   00000000 00000000 00000000 00000000 | ................   |
..1F0     1F0   00000000 0112A002 DF001077 02E2F008 | ............S0.    |
..200     200   00008F3E 40404040 40404040 40404040 | ....               |
..210     210   40404040 40404040 40404040 40404040 |                    |
..220     220   0001E978 0001F7F4 0001F83C 00000000 | ..Z...74..8....    |
..230     230   00000000 00000                      |                    |
..240     240   50C20004 003DA                      |                    |
..250     250   002CCCBC 003DA000 00100000 00084F30 | ..............|.   |
..260     260   0053BC7C 0053BC00 003DA660 003DA088 | ..........w-...h   |
..270     270   0000ECD0 00000000 0000D93C 0000B7B8 | ..........R.....   |
..280     280   008CB000 8077C3E8 8077B3E8 8077A3E8 | ......CY...Y..tY   |
..290     290   0077A520 8077AE8E 000092B0 00000000 | ..v.......k.....   |
..2A0     2A0   0077D188 0076607C 008CB2E8 00740078 | ..Jh..-....Y....   |
..2B0     2B0   8077AD8C 008C9050 008C9000 0000000D | ................   |
..2C0     2C0   00000F4C 002CC008 0002F3D6 0008E5E2 | ...<......30..VS   |
..2D0     2D0   C561C1C6 40F5F6F8 F6F0F6F6 F0F640F1 | E/AF 568606606 1   |
..2E0     2E0   F5C34040 F6F2F140 C4E8F4F4 F3F6F440 | 5C  621 DY44364    |
..2F0     2F0   E5E2C54B C5E2C14B E2E4D7C9 40404040 | VSE.ESA.SUPI       |
```

Exercise 4: System Disabled Wait

System Communications Region:

Low Core Error Code (not set yet)

```
...420          0   0002EEA8 00000F00 0000C152 000037D0 | ...y......A.....  |
...430         10   40023850 00000814 0004E4C0 00007388 |  .........U....h  |
...440         20   00000000 FF086A20 039A003C 000CE657 | ..............W.  |
...450         30   00095C00 000415D0 00000000 00054C9C | ..*...........<.  |
...460         40   C0601CFC 00200060 04000080 000090D8 |
...470         50   00000000 00000000 00000088 003AE000 |
...480         60   80009250 000099D0 0001D0A2 0000C218 |
...490         70   0001B240 0014BED0 000052B4 00000918 | ...  ............  |
...4A0         80   00000000 FFFF0000 0000B480 00000000 | ................  |
...4B0         90   0001E24C 00007F30 01BD0074 00A000C0 | ..S<..".........  |
...4C0         A0   00000000 00000178 0000038B 03010301 | ................  |
...4D0         B0   00000000 0000A724 000549F0 00000000 | ......x....0....  |
```

Current Task id

Low core + x'260' points at the current **Task Control Block (TCB)**:

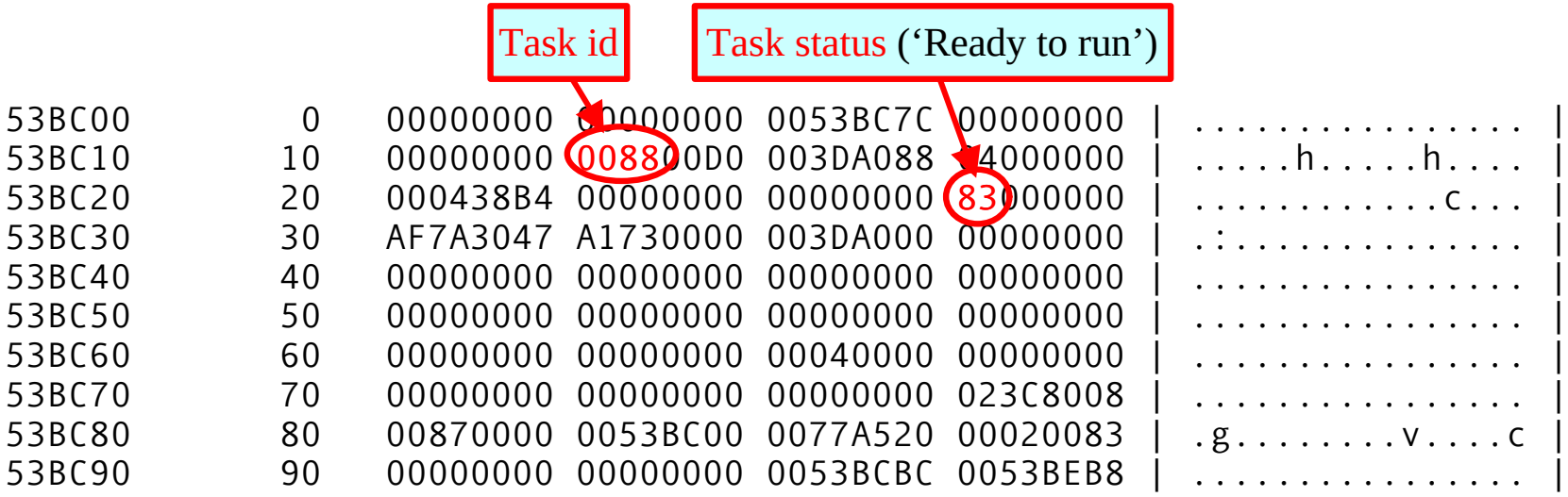
```
53BC7C          0   023C8008 00870000 0053BC00 0077A520 |                .  |
53BC8C         10   00020083 00000000 00000000 0053BCBC |                .  |
53BC9C         20   0053BEB8 02E2D624 00000000 80000000 |                .  |
53BCAC         30   00000000 02E2D7C0 00000000 00000000 | .....SP.........  |
53BCBC         40   00000000 00000000 040C2000 828D5CCA | ............b.*.  |
53BCCC         50   828D58F8 000000FF 02FF3458 02FF3550 | b..8..........  |
53BCDC         60   02FF3458 828D5C88 828BF7B0 00000005 | ....b.*hb.7.....  |
53BCEC         70   02FF353                 00000001 00583000 | .....S.........  |
53BCFC         80   0058300                 00001460 00001460 | ..........-...-  |
53BD0C         90   0000000                 000167F6 00000000 | ..........6....  |
53BD1C         A0   828D58F8 00000000 02FF3458 02FF3550 | b..8..........  |
```

TCBSAVE

TIB

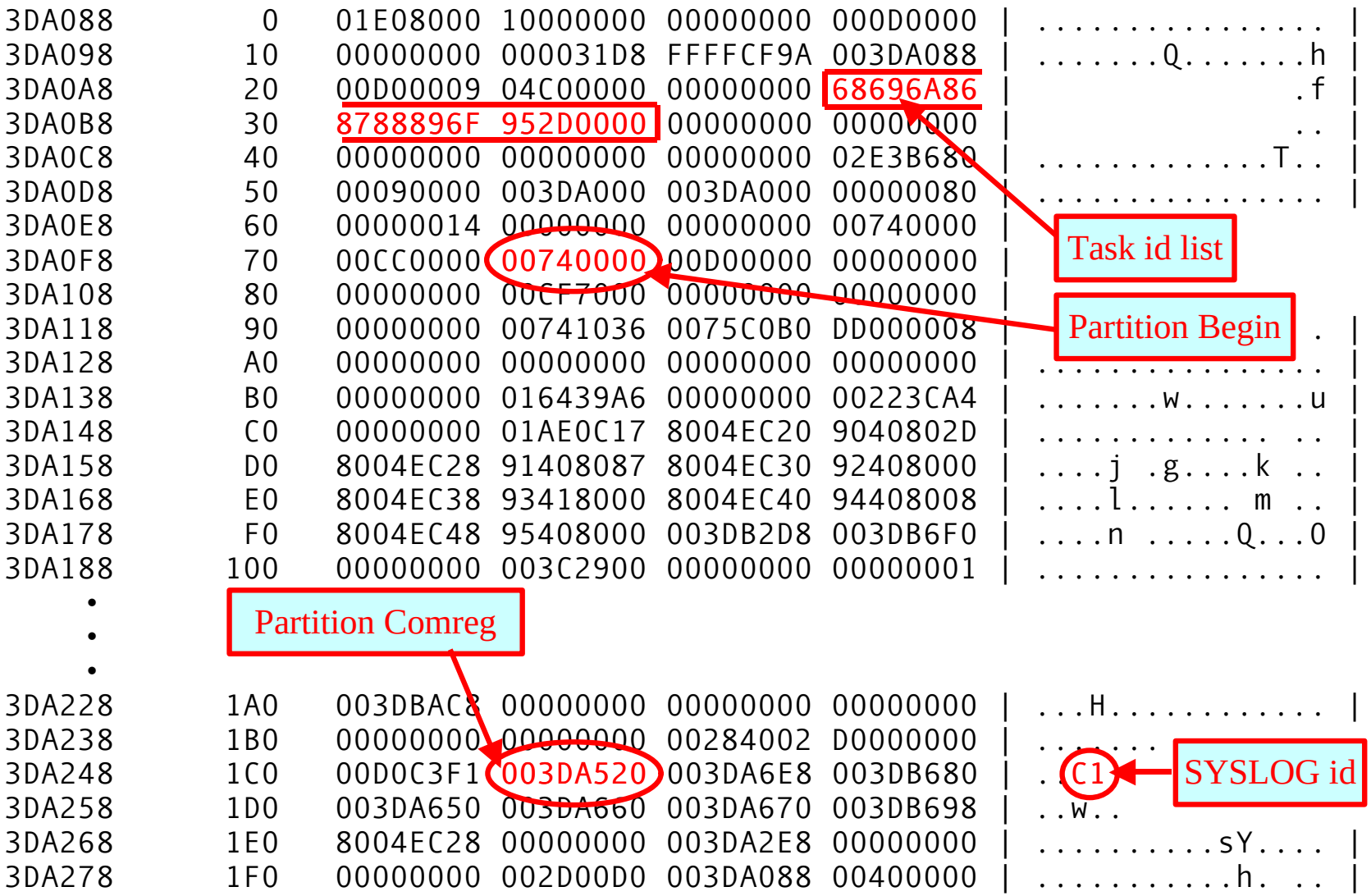
Exercise 4: System Disabled Wait

Task Information Block (TIB) contains the task id and status. Cross-check taskid with SYSCOM+x'5A'



Exercise 4: System Disabled Wait

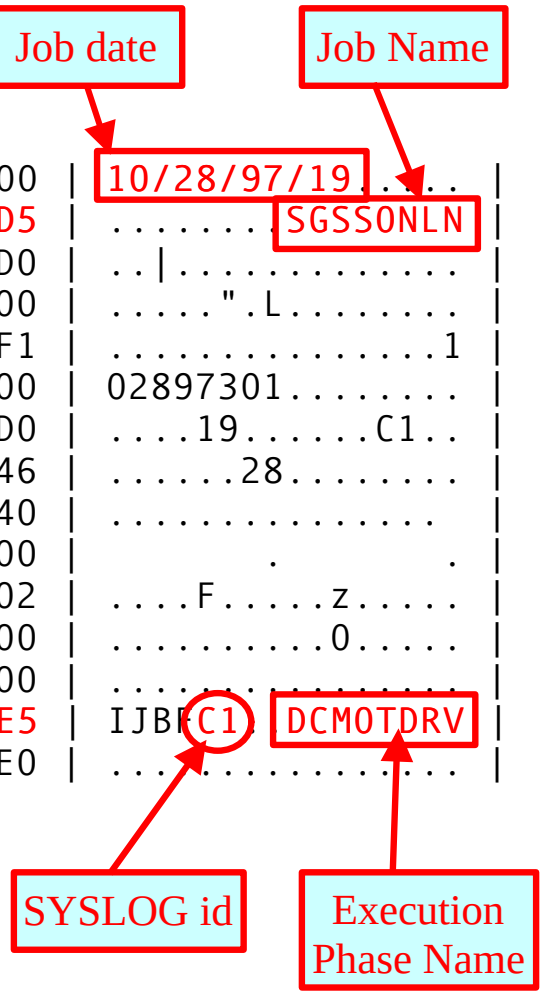
The PCB: tells you what partition is active (which file on stand-alone dump tape)



Exercise 4: System Disabled Wait

Partition Comreg:

```
3DA520        0   F1F061F2 F861F9F7 61F1F900 00000000 | 10/28/97/19....  |
3DA530       10   00000000 00000000 E2C7E2E2 D6D5D3D5 | ........SGSSONLN |
3DA540       20   00744FFF 007446BE 007446BE 000000D0 | ..|.............  |
3DA550       30   02FFFFFF FF7F54D3 200054D0 19000000 | .....".L........  |
3DA560       40   32B0AF79 DC6E6373 012C0139 014638F1 | ................1 |
3DA570       50   F0F2F8F9 F7F3F0F1 000007FE 00000000 | 02897301........  |
3DA580       60   08CE0000 F1F90000 000000D0 C3F100D0 | ....19......C1..  |
3DA590       70   003DB6E0 003CF2F8 0000092C 072E0946 | ......28........  |
3DA5A0       80   00000000 05B811E1 00702000 68680F40 | ..............    |
3DA5B0       90   40404040 40404000 40404040 40404000 |         .         .  |
3DA5C0       A0   003DB718 C6000000 003DA950 02040202 | ....F.....z.....  |
3DA5D0       B0   00000000 00000000 003CF000 0103B900 | ..........0.....  |
3DA5E0       C0   0B01000B 00000000 00CF7000 00000000 | ................  |
3DA5F0       D0   C9D1C2C6 C3F10066 C4C3D4D6 E3C4D9E5 | IJBFC1..DCMOTDRV  |
3DA600       E0   00706A50 00000000 00000000 007400E0 | ...&............  |
```

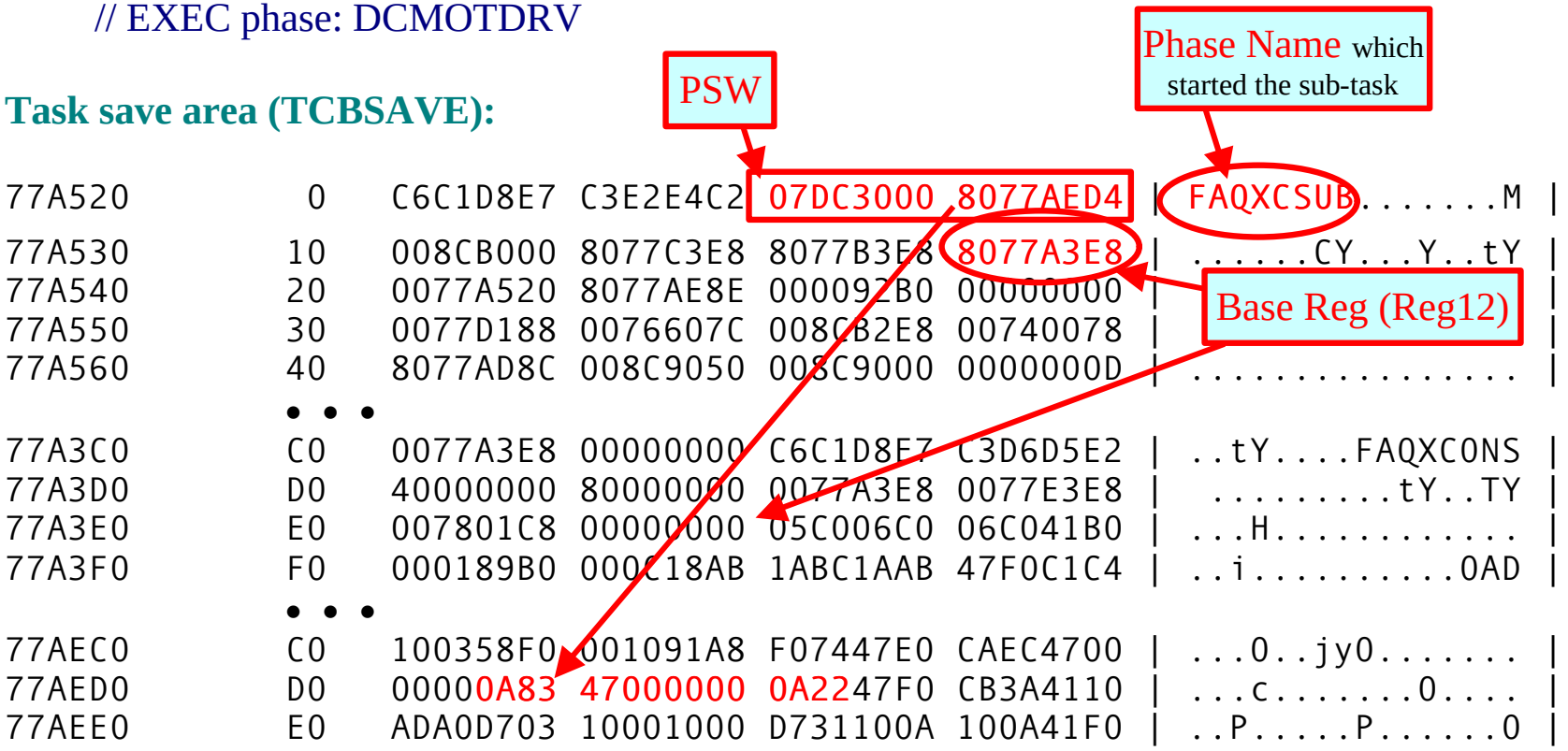


Exercise 4: System Disabled Wait

Ok, let's summarize what we have so far:

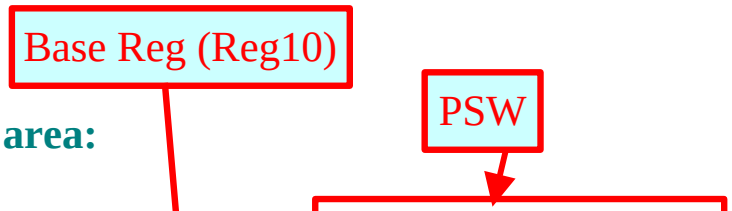
Current Task id: x'0088'
Current Task Save Area: 77A520
Current Partition: C1
Current Job: SGSSONLN
// EXEC phase: DCMOTDRV

Task save area (TCBSAVE):




```
77A520         0   C6C1D8E7 C3E2E4C2 07DC3000 8077AED4 | FAQXCSUB.......M |
77A530        10   008CB000 8077C3E8 8077B3E8 8077A3E8 | .....CY...Y..tY |
77A540        20   0077A520 8077AE8E 000092B0 00000000 |                 |
77A550        30   0077D188 0076607C 008CB2E8 00740078 |                 |
77A560        40   8077AD8C 008C9050 008C9000 0000000D | ................ |
              • • •
77A3C0        C0   0077A3E8 00000000 C6C1D8E7 C3D6D5E2 | ..tY....FAQXCONS |
77A3D0        D0   40000000 80000000 0077A3E8 0077E3E8 |  .........tY..TY |
77A3E0        E0   007801C8 00000000 05C006C0 06C041B0 | ...H..........  |
77A3F0        F0   000189B0 000C18AB 1ABC1AAB 47F0C1C4 | ..i.........0AD |
              • • •
77AEC0        C0   100358F0 001091A8 F07447E0 CAEC4700 | ...0..jy0.......|
77AED0        D0   00000A83 47000000 0A2247F0 CB3A4110 | ...c.......0....|
77AEE0        E0   ADA0D703 10001000 D731100A 100A41F0 | ..P.....P.....0 |
```

Exercise 4: System Disabled Wait



Partition (main task) save area:

```
740000        0   C4C3D4D6 E3C4D9E5 07DD1000 807406B0 | DCMOTDRV........ |
740010       10   007435A8 00740078 00744FFF 0076E900 | ...y......|...Z. |
740020       20   0075C000 0075C208 0075C298 0075C410 | ......B...Bq..D. |
740030       30   0075C27C 0075C204 0074AD80 00CBFFFF | ..B...B......... |
740040       40   00000000 00000000 00702618 00741078 | ................ |
740050       50   0000AF79 DC6E6373 40404040 40404040 | ........         |
740060       60   40404040 40404040 40404040 40404040 |                  |
740070       70   40404040 40404040 47F0F12C C4C3D4D6 |         .01.DCMO |
740080       80   E3C4D9E5 F44BF24B F0404040 F0F461F2 | TDRV4.2.0   04/2 |
740090       90   F261F9F5 E2E4D740 40404040 C7E2E5E2 | 2/95SUP     GSVS |
7400A0       A0   C5404040 4040C396 97A89989 8788A340 | E     Copyright  |
7400B0       B0   4D835D40 F1F9F9F4 40D38587 8595A340 | (c) 1994 Legent  |
7400C0       C0   E29686A3 A6819985 6B40C995 834B6B40 | Software, Inc.,  |
7400D0       D0   F1F9F8F7 6B40F1F9 F9F46B40 81A24081 | 1987, 1994, as a |
7400E0       E0   9540A495 97A48293 89A28885 8440A696 | n unpublished wo |
7400F0       F0   99920000 00000000 D7C1E3C3 C8404040 | rk......PATCH    |
                    •
                    •
                    •
740690       90   41EE0004 47F0A5D8 4100D410 500F0000 | .....0vQ..M..... |
7406A0       A0   9680F000 D203F004 AE504110 D27C0A1D | o.0.K.0.....K... |
7406B0       B0   9180801A 4710A40C 58E0D074 07FE50E0 | j.....u......... |
7406C0       C0   D0745820 D4144122 00045920 D4144780 | ....M.......M... |
```

Exercise 4: System Disabled Wait

Progck old psw: (Store Status registers) document the actual failure**:**

```
28D58F0        70   00000120 00000120 47F0F018 13C3D6C9 | .........00..COI |
28D5900        80   D5D7C8C3 6BF1F5C3 60C4E8F4 F3F7F5F9 | NPHC,15C-DY43759 |
28D5910        90   90ECD00C 189F41F0 00005800 96A458FD | .......0....ou.. |
28D5920        A0   0000581F 00041A01 590F0008 4740903C | ............. .. |
28D5930        B0   500F0008 590F000C 47409046 0000500F | ........ ..... . |
28D5940        C0   000450F1 000018B1 50D0B004 50B0D008 | ...1........... |
28D5950        D0   98F1D010 18DBD207 B0901000 D203B0FC | q1....K.....K... |
28D5960        E0   967C5880 96345850 B0945820 50044120 | o...o....m..... |
28D5970        F0   20D45830 96345030 B0E05020 B0E44140 | .M..o........U.  |
                 •
                 •
                 •
28D5C00         0   96401FAA 186A5840 200095F2 40064770 | o ..... ..n2 ... |
28D5C10        10   936C5850 00805860 501458C0 60805830 | l......-....-... |
28D5C20        20   C02C5850 30245450 960C4780 933E4160 | ........o...l..- |
28D5C30        30   000147F0 936C5860 404041C0 00011826 | ...0l..-  ..... |
28D5C40        40   1E2C5830 96041846 185C9200 40000E24 | ....o....*k. ... |
28D5C50        50   58C00080 58C0C014 58C0C080 58C0C                       |
28D5C60        60   50A0C024 18261256 47709394 58609                       |
28D5C70        70   5060B0E0 5070B0F4 41C0B0F8 50C0B                       |
28D5C80        80   4110B0E0 45E0958C 5850B0F8 41A000FF |                  |
28D5C90        90   155A47D0 93B05860 B0905820 60001845 | .!..l..-....-... |
28D5CA0        A0   50402090 47F093D2 5820B094 D20BB0E0 | . ...0lK...mK... |
28D5CB0        B0   95F05020 B0E458E0 96244110 B0E005E                     |
28D5CC0        C0   D203B0FC 9698 47F0 951A D203 4000967                   |
28D5CD0        D0   50804004 58702034 50704008 5850B09                     |
```

Reg14 from Store Status

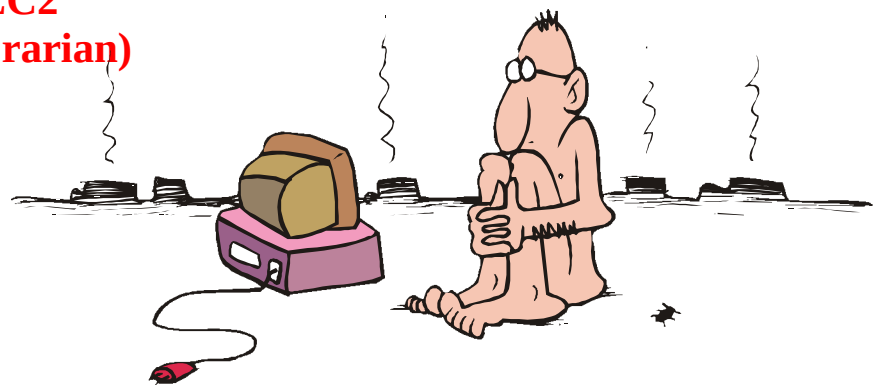
Progck Old PSW Reg9 (from Store Status) points to an invalid page

Exercise 4: System Disabled Wait

Problem Summary:

1. Last call from “main-line” code is found in partition save area, and was at address 7406B0. This was in Legend module DCMOTDRV (Base Reg12), and was an SVC 29 (x’1D’ = Wait multiple). This is documented in the partition save area.
2. This call resulted in a call to FAQS, which then issued an SVC 131 (x’83’ = MVS SVC simulation) at 77AED4 from module FAXSCONS (Base Reg10). This is documented in TCBSAVE.
3. The actual abend, however, was documented via the program check old psw, and was in IBM code. Failing registers are in low core +x’180’ (Store Status).

So, much as we would like to pass this problem to FAQS, the abend occurred in an IBM csect: 'COINPHC' which is part of module IJBCSEC2 and phase $IJBCSIO (IBM Component: Librarian)



Process Stand-alone Dump Tape using INFOANA

Following JCL can be used to define a new SYSDUMP dataset in non-VSE/VSAM space:

```
// JOB DEFINE SYSDUMP AND ASSOCIATED WORK FILES
/* NEW DUMP LABELS
// DLBL SYSDUMP,'VSE.DUMP.LIBRARY',99/365,SD
// EXTENT SYS010,,1,0,nn,mmm
// ASSGN SYS010,DISK,TEMP,VOL=vvvvvv,SHR
// DLBL BLNDMF,'INFO.ANALYSIS.DUMP.MGNT.FILE',0
// EXTENT SYS010,,1,0,nn,mmm
// DLBL BLNXTRN,'INFO.ANALYSIS.EXT.RTNS.FILE',99/365,SD
// EXTENT SYS010,,1,0,nn,mmm
/* END OF DUMP LABELS
// EXEC LIBR
   DEFINE L=SYSDUMP
   DEFINE S=SYSDUMP.DYN
   DEFINE S=SYSDUMP.BG
   DEFINE S=SYSDUMP.F1
   DEFINE S=SYSDUMP.F2
/*
// EXEC INFOANA,SIZE=300K
   SELECT DUMP MANAGEMENT
   UTILITY
   RETURN
   SELECT END
/*
```

Process Stand-alone Dump Tape using INFOANA

```
// UPSI 1
// LIBDEF *,SEARCH=(PRD1.BASE)
// EXEC DITTO
$$DITTO CSQ FILEOUT=BLNXTRN,BLKFACTOR=1
ANEXIT IJBXDBUG ANALYZE STANDALONE DUMP ROUTINE
ANEXIT IJBXCSMG ANALYZE CONSOLE BUFFER
ANEXIT IJBXSDA  SDAID BUFFER FORMATTING ROUTINE
ANEXIT DFHPDAP  CICS/VSE (COEXISTANCE) ANALYZER
ANEXIT DFHPD410 CICS DUMP ANALYZER TS 1.1
/*
/&
```

If BLNDMF file gets out of sequence with the VSEDUMP library, INFOANA may report that the library is full, but in fact there are no dumps in the library at all. The following, quick, job will often resynchronize the library, allowing you to continue to use it. If this does not work, the library is probably corrupted, and will need to be re-built.

```
// JOB INIT SYSDUMP MANAGEMENT FILE
/* NEW DUMP LABELS
... SYSDUMP, BLNDMF and BLNXRTN DLBL / EXTENT / ASSGN ...
/* END OF DUMP LABELS
// EXEC INFOANA,SIZE=300K
   SELECT DUMP MANAGEMENT
   UTILITY
   RETURN
   SELECT END
/*
/&
```

Process Stand-alone Dump Tape using INFOANA

Scan the stand-alone dump tape (and identify the dumps to be onloaded)

```
R  RDR,PAUSEBG
BG 0000 // JOB PAUSEBG
BG-0000 // PAUSE
0 // EXEC DOSVSDMP
BG 0000 4G01D SELECT ONE OF THE FOLLOWING FUNCTIONS:

  1   CREATE STAND-ALONE DUMP PROGRAM
  2   SCAN DUMP TAPE/DISK
  3   PRINT DUMP TAPE/DISK
  4   PRINT SDAID TAPE
  5   PRINT IPL DIAGNOSTICS
  R   END DOSVSDMP PROCESSING
0  2
BG 0000 4G04D SPECIFY ADDRESS OF DUMP DEVICE (CUU OR SYSNNN)
BG-0000 0C66D READY

0  7A8
```

Process Stand-alone Dump Tape using INFOANA

SYSLST Output:

```
PRINTOUT OF VSE DUMP TAPE
DIRECTORY OF VSE DUMP TAPE
DUMP FILE  DUMP TYPE  NAME      DATE      DATA DUMPED
---------  ---------  --------  --------  --------------------
   001                              DOES NOT CONTAIN DUMP DATA
   002                              DOES NOT CONTAIN DUMP DATA
   003     SADUMP                          SUPERVISOR+SVA
   004     SADUMP                99/04/14  PMRAS-R
   005     SADUMP                99/04/14  PMRAS-OO
   006     SADUMP     T77VDA     99/04/14  Z1-PARTITION
   007     SADUMP     T77VEA     99/04/14  Z2-PARTITION
   009     SADUMP     VSECVTIE   99/04/14  F3-PARTITION
   010     SADUMP     POWSTART   99/04/14  F1-PARTITION
END OF DUMP
```

Process Stand-alone Dump Tape using INFOANA

Load file three (Supervisor and SVA) from a stand-alone dump tape.

```
// JOB INFOANAL ONLOAD S/A DUMP FROM TAPE
// ASSGN SYS002,181
// MTC REW,SYS002
/* NEW DUMP LABELS
// DLBL SYSDUMP,'VSE.DUMP.LIBRARY',99/365,SD
// EXTENT SYS010,,1,0,nn,mmm
// ASSGN SYS010,DISK,TEMP,VOL=vvvvvv,SHR
// DLBL BLNDMF,'INFO.ANALYSIS.DUMP.MGNT.FILE',0
// EXTENT SYS010,,1,0,nn,mmm
// DLBL BLNXTRN,'INFO.ANALYSIS.EXT.RTNS.FILE',99/365,SD
// EXTENT SYS010,,1,0,nn,mmm
/* END OF DUMP LABELS
// EXEC INFOANA,SIZE=300K
  SELECT DUMP MANAGEMENT            *** DELETE PREVIOUS DUMPS ***
    DUMP NAME SYSDUMP.F2.SUPSVA
    DELETE
    RETURN
  DUMP NAME SYSDUMP.F2.SUPSVA     *** LOAD NEW DUMP ***
  SELECT DUMP ONLOAD
     VOLID 111111 SYS002
     FILE 3 LAST
     RETURN
  SELECT END
/*
/&
```

Process Stand-alone Dump Tape using INFOANA

Create a list of all dumps currently in the VSEDUMP Library.

```
// JOB INFOANAL LIST ALL DUMPS IN SYSDUMP LIBRARY
/* NEW DUMP LABELS
// DLBL SYSDUMP,'VSE.DUMP.LIBRARY',99/365,SD
// EXTENT SYS010,,1,0,nn,mmm
// ASSGN SYS010,DISK,TEMP,VOL=vvvvvv,SHR
// DLBL BLNDMF,'INFO.ANALYSIS.DUMP.MGNT.FILE',0
// EXTENT SYS010,,1,0,nn,mmm
// DLBL BLNXTRN,'INFO.ANALYSIS.EXT.RTNS.FILE',99/365,SD
// EXTENT SYS010,,1,0,nn,mmm
/* END OF DUMP LABELS
// EXEC INFOANA,SIZE=300K
  SELECT DUMP MANAGEMENT            *** LIST ALL DUMPS IN LIBRARY ***
    PRINT DATA
    RETURN
  SELECT END
/*
/&
```

Process Stand-alone Dump Tape using INFOANA

Print Formatted analysis of dump:

```
// JOB INFOANAL ANALYZE S/A DUMP
/* NEW DUMP LABELS
// DLBL SYSDUMP,'VSE.DUMP.LIBRARY',99/365,SD
// EXTENT SYS010,,1,0,nn,mmm
// ASSGN SYS010,DISK,TEMP,VOL=vvvvvv,SHR
// DLBL BLNDMF,'INFO.ANALYSIS.DUMP.MGNT.FILE',0
// EXTENT SYS010,,1,0,nn,mmm
// DLBL BLNXTRN,'INFO.ANALYSIS.EXT.RTNS.FILE',99/365,SD
// EXTENT SYS010,,1,0,nn,mmm
/* END OF DUMP LABELS
// EXEC INFOANA,SIZE=300K
   SELECT DUMP MANAGEMENT
     DUMP NAME SYSDUMP.F2.P9141003
          RETURN
   SELECT DUMP VIEWING
          CALL IJBXDBUG
          CALL IJBXSDA
          CALL IJBXCSMG
          RETURN
   SELECT DUMP VIEWING
     PRINT FORMAT
   SELECT END
/*
/&
```

Process Stand-alone Dump Tape using INFOANA

Format CICS TS System Dump, or CICS TS partition from Stand-alone dump:

```
// JOB DFHPD410 FORMAT CICS TS SYSTEM DUMP
/* NEW DUMP LABELS
// DLBL SYSDUMP,'VSE.DUMP.LIBRARY',99/365,SD
// EXTENT SYS010,,1,0,nn,mmm
// ASSGN SYS010,DISK,TEMP,VOL=vvvvvv,SHR
// DLBL BLNDMF,'INFO.ANALYSIS.DUMP.MGNT.FILE',0
// EXTENT SYS010,,1,0,nn,mmm
// DLBL BLNXTRN,'INFO.ANALYSIS.EXT.RTNS.FILE',99/365,SD
// EXTENT SYS010,,1,0,nn,mmm
/* END OF DUMP LABELS
// LIBDEF *,SEARCH=(PRD1.BASE)
// EXEC INFOANA,SIZE=300K
  DUMP NAME SYSDUMP.F2.F2DUMP
  SELECT DUMP SYMPTOMS
    PRINT DATA
    RETURN
  SELECT DUMP VIEWING
    CALL DFHPD410 DATA DEF=0,KE=3,AP=3,DS=3,TR=1
    RETURN
  SELECT END
/*
/&
```

Process Stand-alone Dump Tape using INFOANA

Off-load dump to tape to send to VSE/ESA Level2, or for off-line archiving.

```
// JOB OFFLOAD DUMPS FROM VSEDUMP LIBRARY
// ASSGN SYS009,182
// MTC REW,182
/* NEW DUMP LABELS
// DLBL SYSDUMP,'VSE.DUMP.LIBRARY',99/365,SD
// EXTENT SYS010,,1,0,nn,mmm
// ASSGN SYS010,DISK,TEMP,VOL=vvvvvv,SHR
// DLBL BLNDMF,'INFO.ANALYSIS.DUMP.MGNT.FILE',0
// EXTENT SYS010,,1,0,nn,mmm
// DLBL BLNXTRN,'INFO.ANALYSIS.EXT.RTNS.FILE',99/365,SD
// EXTENT SYS010,,1,0,nn,mmm
/* END OF DUMP LABELS
// EXEC INFOANA,SIZE=300K
  DUMP NAME SYSDUMP.F1.DF100064  *** OFFLOAD DUMP TO TAPE ***
  SELECT DUMP OFFLOAD
    VOLID 111111 SYS009
    ERASE NO      <<<Specify "ERASE YES" if dump is no longer required on-line >>>
    RETURN
  DUMP NAME SYSDUMP.F1.DF100065  *** OFFLOAD DUMP TO TAPE ***
  SELECT DUMP OFFLOAD
    VOLID 111111 SYS009
    ERASE NO
    RETURN
  SELECT END
/*
/&
```

Process Stand-alone Dump Tape using INFOANA

Delete dumps no longer required on-line.

```
// JOB INFOANAL DELETE S/A DUMP FROM SYSDUMP LIBRARY
/* NEW DUMP LABELS
// DLBL SYSDUMP,'VSE.DUMP.LIBRARY',99/365,SD
// EXTENT SYS010,,1,0,nn,mmm
// ASSGN SYS010,DISK,TEMP,VOL=vvvvvv,SHR
// DLBL BLNDMF,'INFO.ANALYSIS.DUMP.MGNT.FILE',0
// EXTENT SYS010,,1,0,nn,mmm
// DLBL BLNXTRN,'INFO.ANALYSIS.EXT.RTNS.FILE',99/365,SD
// EXTENT SYS010,,1,0,nn,mmm
/* END OF DUMP LABELS
// EXEC INFOANA,SIZE=300K
  SELECT DUMP MANAGEMENT             *** DELETE DUMP ***
    DUMP NAME SYSDUMP.BG.DBG00006
      DELETE
    DUMP NAME SYSDUMP.BG.DBG00005
      DELETE
    DUMP NAME SYSDUMP.F2.DF200002
      DELETE
    DUMP NAME SYSDUMP.F2.DF200001
      DELETE
  SELECT END
/*
/&
```