

01 ;
02 ;
03 ;
04 ;
05 ;
06 ;
07 ;
08 ;
09 ;
10 ;
11 ;
12 ;
13 ;
14 ;
15 ;
16 ;
17 ;
18 ;
19 ;
20 ;
21 ;
22 ;
23 ;
24 ;
25 ;
26 ;
27 ;
28 ;
29 ;
30 ;
31 ;
32 ;
33 ;
34 ;
35 ;
36 ;
37 ;
38 ;
39 ;
40 ;
41 ;
42 ;
43 ;
44 ;
45 ;
46 ;
47 ;
48 ;
49 ;
50 ;
51 ;
52 ;
53 ;
54 ;
55 ;
56 ;
57 ;
58 ;

CM002

; KEYWORDS: MUS, COROUTINE MONITOR, LISTING
; ABSTRACT: MULTIPROGRAMMING UTILITY SYSTEM COROUTINE MONITOR
; THE MONITOR CAN BE USED BY MUSIL PROGRAMS
; AS WELL AS BY ASSEMBLER PROGRAMS.
; THE MONITOR IS AN EXTENDED VERSION OF CM000
; WITH FACILITIES FOR HANDLING OF GENERAL SEMAPHOR
; RCSL 43-GL5087 ASCII SOURCE PAPER TAPE
; RCSL 43-GL5089 REL. BINARY PAPER TAPE

```

01 ; COROUTINE MONITOR
02 ;
03 ;
04 ; ASSEMBLER PREDEFINED SYMBOLS INTRODUCED BY THE COROUTINE MON
05 ;
06 ;
07 ; PAGE 0 ENTRY POINTS:
08 ;
09 ; CDELAY      WAITS N TIME UNITS
10 ; WAITSEM    WAITS IN A SIMPLE SEMAPHORE
11 ; WAITCH     WAITS IN A CHAINED SEMAPHORE
12 ; CWANSWER   WAITS FOR AN ANSWER
13 ; CTEST      TESTOUT OF REGISTERS
14 ; CPRINT     TESTOUT OF RECORD(AC2)
15 ; CTOUT      GENERAL TEST CALL
16 ; SIGNAL     SIGNALS IN A SIMPLE SEMAPHORE
17 ; SIGCH      SIGNALS IN A CHAINED SEMAPHORE
18 ; CPASS      ALLOWS ACTIVATION OF OTHER COROUTINES
19 ; CSENDM     SENDS A MESSAGE IN A COROUTINE MESSAGE BUFFER
20 ; SIGGEN     SIGNALS IN A GENERAL SEMAPHORE
21 ; WAITGEN    WAITS IN A GENERAL SEMAPHORE
22 ;
23 ; CUROUT     PAGE 0 POINTER TO CURRENT COROUTINE
24 ; CTCP       PAGE 0 POINTER TO TESTOUTPUT PROCEDURE
25 ;
26 ;
27 ; PROCESS DESCRIPTION RELATIVES:
28 ;
29 ; CCOROUT    CURRENT COROUTINE
30 ; LATIME     LATEST TIME
31 ; HACTIV     HEAD OF ACTIVE QUEUE
32 ; HANSWER    HEAD OF ANSWER QUEUE
33 ; HDELAY     HEAD OF DELAY QUEUE
34 ; TRETURN    TEST RETURN
35 ; TRECORDER  TEST RECORD START ADDRESS
36 ; CDEVICE    DEVICECODE USED IN CENTRAL WAIT
37 ; MSEM       MESSAGE/INTERRUPT RECEIVING SEMAPHORE
38 ; MCCOROUT   MESSAGE/INTERRUPT RECEIVING COROUTINE
39 ; CUDEX      USER DEFINED EXIT BEFORE COROUTINE EXIT
40 ; CBUFFER    COROUTINE MESSAGEBUFFER SEMAPHORE
41 ;
42 ; ! NOTE !   PC (CUR+33) IS USED BY CWANSWER
43 ;
44 ; TLENGTH    LENGTH OF TEST RECORD (BYTES)
45 ;
46 ;
47 ; COROUTINE RELATIVES:
48 ;
49 ; OPMASK:    MASK FOR OPERATION TYPES
50 ; CIDENT:    PRIORITY,TEST,IDENT
51 ; REF COROUT:NEXT: LINK
52 ; CEXIT:     EXIT TO COROUTINE
53 ; CLATOP:    LATEST OPERATION
54 ; CRETURN:   COMMON PROCEDURE RETURN
55 ; CAC1SAVE:  SAVED AC1
56 ; CSPC:     SAVED PC (MUSIL)
57 ;

```

```
01 ; COROUTINE MONITOR
02 ;
03 ; FORMATS:
04 ;
05 ;
06 ; TEST RECORD RELATIVES:
07 ;
08 000000 TKIND=+0
09 000001 TPROC=TKIND+1
10 000002 TIDENT=TPROC+1
11 000003 TTIME=TIDENT+1
12 000004 TTIM1=TTIME+1
13 000005 TAC0=TTIM1+1
14 000006 TAC1=TAC0+1
15 000007 TAC2=TAC1+1
16 000010 TAC3=TAC2+1
17 000011 TOPT1=TAC3+1
18 000012 TOPT2=TOPT1+1
19 000013 TOPT3=TOPT2+1
20 000014 TOPT4=TOPT3+1
21 000015 TOPT5=TOPT4+1
22 000016 TOPT6=TOPT5+1
23 ;
24 ; TKIND LONG B0 + INHIBIT B7 + BACKWARD B8 + KIND
25 ;
```

```

01 ; COROUTINE MONITOR
02 ;
03 ; FORMATS:
04 ;
05 ; CIDENT: PRIORITY B0 + TEST B7 + IDENT
06 ; REF COROUTINE: LINK
07 ; CEXIT
08 ; CLATOP
09 ; CRETURN
10 ; CACISAVE
11 ;
12 ;
13 ; REF OPERATION:LINK (REST NOT USED BY MON
14 ; OPERATION CODE
15 ; ANSWER SEM
16 ; MESSAGE (4 WORDS)
17 ;
18 ;
19 ; COROUTINE TEST BITS: ( 1 MEANS TESTOUT ON )
20 ;
21 ; 1 CWANSWER
22 ; 2 CPASS/CDELAY
23 ; 4 EXIT
24 ; 8 CTEST/CPRINT
25 ; 16 WAITCH/WAITSEM
26 ; 32 SIGNAL
27 ; 64 SIGCH
28 ;
29 ;
30 ;
31 ;
32 ; CGROUT CURRENT COROUTINE POINTER IN PAGE 0
33 ;
34 ;

```

01 ; COROUTINE MONITOR

02 ;

03 ;

04 ;

05 ; CONVENTIONS FOR CALL :

06 ;

07 ;

08 ; AC0 AC1 AC2 AC3 LATOP TEST

09 ;-----

10 ; SIGNAL SEM LINK 32B7+2

11 ; UNCH LATOP COROUT UNCH

12 ;

13 ; SIGCH SEM OP LINK 1B0+64B7+1

14 ; UNCH LATOP COROUT UNCH

15 ;

16 ; CPASS LINK 2B7+6

17 ; UNCH LATOP COROUT UNCH

18 ;

19 ; CWANSWER BUF LINK 1B7+5

20 ; UNCH BUF COROUT BUF

21 ;

22 ; WAITSEM SEM LINK 16B7+4

23 ; UNCH LATOP COROUT UNCH

24 ;

25 ; CTEST LINK 1B0+8B7+9

26 ; UNCH UNCH UNCH COROUT UNCH

27 ;

28 ; CPRINT REF-1 LINK 1B0+8B7+10

29 ; UNDEF UNDEF UNDEF COROUT UNCH

30 ;

31 ; CTOUT LINK GENERAL

32 ; UNCH UNCH UNCH COROUT UNCH

33 ;

34 ; WAITCH SEM LINK 16B7+3

35 ; UNCH LATOP COROUT NEW OP

36 ;

37 ; CDELAY DELAY LINK 2B7+8

38 ; UNCH UNDEF COROUT UNDEF

39 ;

40 ; EXIT UNCH LATOP COROUT UNCH 1B0+4B7+7

41 ;

42 ;

43 ;

44 ;

45 ;

01 ;
02 ; COROUTINE MONITOR
03 ;
04 ; SYMBOLS NOT DEFINED BY ASMUS 10
05 ;
06 ;

08	006364	CSENDMESSAGE	=	JSR	@364
09	006365	SIGGEN	=	JSR	@365
10	006366	WAITGEN	=	JSR	@366
11	006367	CTOP	=	JSR	@367
12					
13	177777	OPTYPE	=	NEXT-1	
14	177776	ANSWSEM	=	OPTYPE-1	
15					
16	000001	NXTOP	=	NEXT+1	
17	000002	TIMEOUT	=	NXTOP+1	
18	000003	NXTCO	=	TIMEOUT+1	
19					
20	177776	OPMASK	=	CIDENT-1	
21	000005	CSPC	=	CAC1SAVE+1	
22					
23	000050	CDEVICE	=	TRECORD+1	
24	000051	MSEM	=	CDEVICE+1	
25	000052	MCOROUT	=	MSEM+1	
26	000053	CUDEX	=	MCOROUT+1	
27	000054	CBUFFER	=	CUDEX+1	

```

↑ 0007 CM002
01 ; COROUTINE MONITOR
02 ;
03 ; ASSEMBLER OPS:
04 ;
05
06 000001 .TXTM 1 ; TEXTMODE 1
07 000012 .RDX 10 ; RADIX 10
08 .TITL CM002 ; COROUTINE MONITOR VERSION
09 .NREL ;
10
11 ; COROUTINE MONITOR
12 ;
13 ; PROGRAM HEAD
14 ;
15
16 .NREL
17 00000'100001 FIRST: 1B0+1 ; FIRST: SPEC
18 00001'000000 0 ; START: 0
19 00002'000000 0 ; CHAIN: 0
20 00003'001434 LAST-FIRST ; SIZE: TOTAL PROG
21 .TXT .CM002.
00004'041515
00005'030060
00006'031000

```

```

01      ; COROUTINE MONITOR
02      ; ENTRY POINT CSENDMESSAGE:
03      ;
04      ; ENTRY:  AC0=ANSWER SEM.  AC1=MADDR  AC2=NAME ADDR  AC3=LINK
05      ; EXIT:   AC0,             AC1 UNCH   AC2=BUF    AC3=CUR
06
07 00007'060277 CSMESS: INTDS ; COROUTINE SEND MESSAGE:
08 00010'054565 STA 3,AC3SAVE ; DISABLE;
09 00011'004552 JSR SETEX ; SET EXIT;
10 00012'004565 JSR GENTO ; GENERATE TESTOUTPUT(
11 00013'120015 1B0+32B7+13 ; LONG, INHIBIT 32,KIND 13);
12 00014'025401 LDA 1,CEXIT,3 ;
13 00015'034040 LDA 3,CUR ;
14 00016'045446 STA 1,TRETURN,3 ; TRETURN.CUR:= CEXIT.COROUT;
15 00017'105000 MOV 0,1 ; SAVE ANSWER SEM;
16 00020'006446 JSR @PGETCM ; GET MESSAGE OPERATION;
17 00021'175005 MOV 3,3,SNR ; IF NO AVAILABLE THEN
18 00022'000436 JMP NOMOP ; GOTO RETURN ERROR NO=-3
19 00023'045776 STA 1,ANSWSEM,3 ; INITIALIZE MESS OPERATION
20 00024'024117 LDA 1,.1B14 ; AS A MESSAGE BUFFER:
21 00025'045777 STA 1,OPTYPE,3 ; ANSWERSEM.OP:= SAVED AC0;
22 00026'115000 MOV 0,3 ; OPTYPE.OP:= ANSWER TYPE;
23 00027'055400 STA 3,NEXT,3 ; NEXT.MOP:= MOP;
24 00030'055401 STA 3,PREV,3 ; PREV.MOP:= MOP;
25 00031'145000 MOV 2,1 ;
26 00032'102400 SUB 0,0 ;
27 00033'041405 STA 0,RECEIVER,3 ; RECEIVER.MOP:= 0;
28 00034'030040 LDA 2,CUR ;
29 00035'051404 STA 2,SENDER,3 ; SENDER.MOP:= CUR;
30 00036'021011 LDA 0,BUFFER,2 ; CHAIN MESS.OP TO START OF
31 00037'041402 STA 0,CHAIN,3 ; MESSAGE BUFFER CHAIN;
32 00040'055011 STA 3,BUFFER,2 ;
33 00041'021412 LDA 0,MESS3+1,3 ; AC0:= ANSWER SEM;
34 00042'131000 MOV 1,2 ; AC2:= NAME ADDR;
35 00043'034017 LDA 3,COROUT ;
36 00044'025404 LDA 1,CAC1SAVE,3 ; AC1:= MADDR;
37 00045'006004 SENDMESSAGE ; SEND MESS.OP AS MESSAGE
38 00046'151133 MOVZL# 2,2,SNR ; IF BUF>=0 THEN
39 00047'003446 JMP @TRETURN,3 ; EXIT TO COROUTINE
40 00050'145000 MOV 2,1 ; ELSE MESSAGE NOT SENT:
41 00051'031411 LDA 2,BUFFER,3 ; SAVE ERROR NUMBER;
42 00052'006415 JSR @PUNCH ; REMOVE MESSBUF FROM CHAIN AGAI
43 00053'035454 LDA 3,CBUFFER,3 ; RETURN MESSAGE OPERATION
44 00054'055000 STA 3,NEXT,2 ; TO THE QUEUE;
45 00055'034040 LDA 3,CUR ;
46 00056'051454 STA 2,CBUFFER,3 ;
47 00057'131001 MOV 1,2,SKP ; RESTORE ERROR NUMBER;
48 00060'030144 NOMOP: LDA 2,.M3 ; IF NO MESS.OP THEN ERROR NO.:=
49 00061'034017 LDA 3,COROUT ;
50 00062'025404 LDA 1,CAC1SAVE,3 ; RESTORE MESSAGE ADDRESS;
51 00063'034040 LDA 3,CUR ;
52 00064'060177 INTEN ; ENABLE;
53 00065'003446 JMP @TRETURN,3 ; EXIT TO COROUTINE;
54 ;
55 00066'001107'PGETCM: GETCMB ; REF GETCMB
56 00067'001126'PUNCH: UNCHAIN ; REF UNCHAIN

```



```

01 ; COROUTINE MONITOR
02 ;
03 ; COMMON PROCEDURES
04
05 ;
06 ; PROCEDURE COMMON SIGNAL(SEM,OP);
07 ; SIGNALS A SEMAPHORE AND EXITS IF IT IS CLOSED. RETURN
08 ; OTHERWISE. SEM IS SAVED IN SEMADR. AC2=CP (CHAINED) OR 0 (SIM
09 ;
10 ; ENTRY: AC1= SEM ADR, AC2= OP, AC3=LINK;
11 ; EXIT: AC1= OPADR , AC2= SEM VALUE>1;
12 ;
13 ; COMMON SIGNAL:
14 00070'044506 COMSIG: STA 1,SAVSEM ; SAVE SEM ADR;
15 00071'145000 MOV 2,1 ; AC1:= OP ADR;
16 00072'032504 LDA 2,@SAVSEM ; AC2:= SEM VALUE>1;
17 00073'151222 MOVZR 2,2,SZC ; IF NOT CLOSED THEN
18 00074'001400 JMP 0,3 ; RETURN;
19 ;
20 00075'021000 LDA 0,0,2 ; SEM CLOSED: AC2:= COROUT.SEM
21 00076'101125 MOVZL 0,0,SNR ; AC0:= LINK.AC2 < 1;
22 00077'101400 INC 0,0 ; SEM:= AC0 + IF END CHAIN
23 00100'042476 STA 0,@SAVSEM ; THEN 1 ELSE 0;
24 00101'125004 MOV 1,1,SZR ; IF OP<>0 THEN
25 00102'045002 STA 1,CLATOP,2 ; LATEST OP.AC2:= OP ADR;
26 00103'004403 JSR ACTCOR ; INSERT AC2 IN ACTIVE QUEUE
27 00104'002570 JMP @PEXITC ; GOTO EXITC;
28 ;
29 ;
30 ; PROCEDURE INSERT IN ACTIVE QUEUE:
31 ;
32 ; ENTRY: AC2= NEW COROUT, AC3= LINK;
33 ; EXIT: AC2= NEW COROUT;
34 ;
35 00105'030017 ACTCC: LDA 2,COROUT ; INSERT CURRENT IN ACTIVE QUE
36 00106'054465 ACTCOR: STA 3,RET ; INSERT AC2 IN ACTIVE QUEUE:
37 00107'034040 LDA 3,CUR ; SAVE RETURN;
38 00110'020416 LDA 0,XACTIV ; AC3:= REF ACTIVE QUEUE;
39 00111'117000 ADD 0,3 ;
40 00112'021377 LDA 0,CIDENT,2 ; IF PRIORITY.AC2 THEN
41 00113'101123 MOVZL 0,0,SNC ; CHAIN TO HEAD
42 00114'000404 JMP REP5 ; ELSE CHAIN TO TAIL;
43 00115'021400 LDA 0,NEXT,3 ;
44 00116'000405 JMP BYP5 ;
45 00117'115000 REP3: MOV 0,3 ; AC3:= NEXT;
46 00120'021400 REP5: LDA 0,NEXT,3 ; REP5: AC0:= NEXT.AC3;
47 00121'101004 MOV 0,0,SZR ; IF AC0<>0 THEN
48 00122'000775 JMP REP3 ; GO ON ELSE
49 00123'051400 BYP5: STA 2,NEXT,3 ; CHAIN: LINK.AC3:= NEW;
50 00124'041000 STA 0,NEXT,2 ; LINK.NEW:= NEXT;
51 00125'002446 JMP @RET ; GOTO RETURN;
52 00126'000043 XACTIV: HACTIV ; REL OF ACTIVE HEAD;
53

```

```

01 ; COROUTINE MONITOR
02 ;
03 ;
04 ; ENTRY POINT SIGNAL:
05 ;
06 ; ENTRY: AC1=SEM, AC3=LINK;
07 ; EXIT: AC1=SEM, AC2=LATOP, AC3=COROUT;
08 ;
09 00127'060277 SIGSEM: INTDS ; SIGNAL SEMAFOR: DISABLE;
10 00130'054445 STA 3,AC3SAVE ;
11 00131'004432 JSR SETEX ; SET EXIT;
12 00132'004445 JSR GENTO ; GENERATE TESTOUTPUT(
13 00133'020002 32B7+2 ; KIND 2,INHIBIT 32);
14 00134'152400 SUB 2,2 ;
15 00135'004733 JSR COMSIG ; COMMON SIGNAL(OP=0);
16 00136'151540 INCOL 2,2 ; SEM NOT CLOSED:
17 00137'052437 STA 2,@SAVSEM ; SEM:=(COUNT+1) * 2+ 1;
18 00140'002534 JMP @PEXITC ; GOTC EXITC;
19 ;
20 ;
21 ; ENTRY POINT SIGCH:
22 ;
23 ; ENTRY: AC1=SEM, AC2=OP, AC3=LINK;
24 ; EXIT: AC1=SEM, AC2=LATOP,AC3=COROUT;
25 ;
26 00141'060277 SIGNCH: INTDS ; SIGNAL CHAINED: DISABLE;
27 00142'054433 STA 3,AC3SAVE ;
28 00143'004420 JSR SETEX ; SET EXIT;
29 00144'004433 JSR GENTO ; GENERATE TESTOUTPUT(
30 00145'140001 1B0+64B7+1 ; LONG,KIND 1,INHIBIT 64);
31 00146'004722 JSR COMSIG ; COMMN SIGNAL(OP ADR);
32 00147'052427 STA 2,@SAVSEM ; SEM NOT CLOSED:
33 00150'020426 LDA 0,SAVSEM ; AC1=OP ADR; AC3:= SEM;
34 00151'115000 REP4: MOV 0,3 ; REP4: AC3:= NEXT;
35 00152'021400 LDA 0,NEXT,3 ; AC0:= NEXT.AC3;
36 00153'101004 MOV 0,0,SZR ; IF NEXT<> 0 THEN
37 00154'000775 JMP REP4 ; GO CN;
38 00155'045400 EX4: STA 1,NEXT,3 ; CHAIN.PRE:= OP ADR;
39 00156'043400 STA 0,@NEXT,3 ; CHAIN.OP:= 0;
40 00157'022417 LDA 0,@SAVSEM ;
41 00160'101140 MOVOL 0,0 ; SEM VALUE:= SEM VALUE<1+1
42 00161'042415 STA 0,@SAVSEM ; (OPEN SEM);
43 00162'002512 JMP @PEXITC ; GOTC EXITC;

```

↑ 0011 CM002

```
01 ; COROUTINE MONITOR
02 ;
03 ; PROCEDURE SET EXIT:
04 ;
05 ; ENTRY: AC3=LINK;
06 ; EXIT: AC0 UNCH, AC1 UNCH, AC2 UNCH, AC3=COROUT;
07 ;
08 00163'054410 SETEX: STA 3,RET ; SET EXIT: SAVE RETURN;
09 00164'040410 STA 0,AC0SAV ; SAVE AC0;
10 00165'020410 LDA 0,AC3SAV ; AC0:= SAVED AC3;
11 00166'034017 LDA 3,COROUT ; CEXIT.CORCUT:= AC0;
12 00167'041401 STA 0,CEXIT,3 ;
13 00170'045404 STA 1,CAC1SAVE,3 ; CAC1SAVE.CORCUT:= AC1;
14 00171'020403 LDA 0,AC0SAV ; REESTABLISH AC0;
15 00172'002401 JMP @RET ; GOTO RETURN;
16 00173'000000 RET: 0 ;
17 00174'000000 AC0SAVE: 0 ;
18 00175'000000 AC3SAVE: 0 ;
19 00176'000000 SAVSEM: 0 ;
20
```

```

01      ;
02      ; PROCEDURE GENERATE TESTOUTPUT
03      ;
04      ;   DELIVERS A TESTOUTPUT RECORD BY CALLING THE CTOP
05      ;   PROCEDURE LOCATED IN THE TESTPROCESS. IF CTOP IS NOT
06      ;   DEFINED AN IMMEDIATE RETURN IS MADE. THE PROCEDURES
07      ;   MUST BE CALLED DISABLED AND WILL RETURN WITOUT
08      ;   ENABLING THE INTERRUPT SYSTEM. ONLY THE TAC2, TAC3, AND
09      ;   TKIND FIELDS IN THE TESTRECORD, WILL BE SET BY THIS
10      ;   PROCEDURE.
11      ;
12      ; CALL:
13      ; RETURN:  AC0,  AC1,  AC2  UNCHANGED  AC3=LINK
14      ;           AC3=COROUT
15      ;
16 00177'054424 GENTO:  STA      3,TRET      ; GENERATE TESTOUTPUT:
17 00200'034367          LDA      3,CTOP-GOS  ; SAVE RETURN;
18 00201'175004          MOV      3,3,SZR    ; IF CTOP PRESENT THEN
19 00202'000404          JMP      CTPREP    ; GOTO PREPARE CALL
20 00203'010420 CTEX:   ISZ      TRET      ; ELSE
21 00204'034017          LDA      3,COROUT  ;
22 00205'002416          JMP      @TRET    ; RETURN;
23                          ;
24 00206'034040 CTPREP: LDA      3,CUR      ; PREPARE CTOP CALL:
25 00207'035447          LDA      3,TRECORD,3 ;
26 00210'175120          MOVZL   3,3      ; GET TESTRECORD ADDRESS;
27 00211'175220          MOVZR   3,3      ;
28 00212'051407          STA      2,TAC2,3  ; SAVE AC2 IN TESTRECORD;
29 00213'171000          MOV      3,2      ; AC2:= TESTRECORD ADDRESS;
30 00214'036407          LDA      3,@TRET  ;
31 00215'055000          STA      3,TKIND,2 ; SAVE TESTKIND IN TESTRECORD;
32 00216'034017          LDA      3,COROUT  ;
33 00217'035401          LDA      3,CEXIT,3 ;
34 00220'055010          STA      3,TAC3,2  ; SAVE CORCUTINE LINK IN TESTREC
35 00221'006367          CTOP      ; DELIVER TESTRECORD
36 00222'000761          JMP      CTEX    ; EXIT FROM TESTOUTPUT
37                          ;
38 00223'000000 TRET:   0              ; SAVED LINK FOR GENTO
39                          ;

```

```

01 ; CORROUTINE MONITOR
02 ;
03 ; PROCEDURE COMMON WAIT:
04 ;
05 ; RETURNS IF OPEN WITH SEM ADR SAVED. CHAINS THE CORROUTINE TO T
06 ; SEMAPHORE IF CLOSED AND ACTIVATES THE NEXT CORROUTINE.
07 ;
08 ; ENTRY: AC1=SEM, AC3=LINK;
09 ; EXIT: AC1=SEM, AC2=CHAIN.SEM, AC3=COROUT;
10 ;
11 00224'054747 CWAIT: STA 3,RET ; COMMON WAIT:
12 00225'034017 LDA 3,COROUT ; AC3:= COROUT;
13 00226'044750 STA 1,SAVSEM ; SAVE SEMAPHORE ADDR;
14 00227'032747 LDA 2,@SAVSEM ;
15 00230'151226 MOVZR 2,2,SEZ ; AC2:= SEM VALUE >1;
16 00231'002742 JMP @RET ; IF OPEN THEN RETURN;
17 00232'052744 STA 2,@SAVSEM ;SEM NOT OPEN:
18 00233'020743 LDA 0,SAVSEM ; REM SIGN BIT MAKES SEM A
19 00234'111000 REP1: MOV 0,2 ; REP1: PRE:= NEXT;
20 00235'021000 LDA 0,NEXT,2 ; AC0:=NEXT; AC2:=PRE;
21 00236'101004 MOV 0,0,SZR ; IF AC0<>0 THEN
22 00237'000775 JMP REP1 ; GOTC REP1;
23 00240'055000 BYP1: STA 3,NEXT,2 ; STORE CURRENT IN END OF C
24 00241'041400 STA 0,NEXT,3 ; STORE END OF CHAIN IN CUR
25 00242'022734 LDA 0,@SAVSEM ;
26 00243'101120 MOVZL 0,0 ; SEM VALUE:=SEM VALUE < 1+
27 00244'042732 STA 0,@SAVSEM ;
28 00245'002463 JMP @PNEXTC ; GOTC ACTIVATE NEXT COROUT

```

```

01      ; COROUTINE MONITOR
02      ;
03      ; ENTRY POINT CDELAY:
04      ;
05      ; ENTRY:          AC1=DELAY,          AC3=LINK;
06      ; EXIT:           AC1 UNCH, AC2 UNDF,  AC3=COROUT;
07      ;
08 00246'060277 CDEL:      INTDS              ; DELAY:
09 00247'054726      STA          3,AC3SAVE    ; SAVE AC3;
10 00250'004713      JSR          SETEX       ; SET EXIT;
11 00251'004726      JSR          GENTO      ; GENERATE TESTOUTPUT(
12 00252'001010      287+8                ; KIND 8,INHIBIT 2);
13 00253'125005      MOV          1,1,SNR     ; IF DELAY=0 THEN
14 00254'002420      JMP          @PEXITC    ; GOTC EXIT TO COROUTINE;
15 00255'020075      LDA          0,RTIME+1   ;
16 00256'030040      LDA          2,CUR      ;
17 00257'031042      LDA          2,LATIME,2  ; SAVE DELAY:= LATOP.COROU
18 00260'142400      SUB          2,0        ; DELAY+RTIME-LATIME;
19 00261'107023      ADDZ         0,1,SNC     ; IF OVERFLOW OR RESULT=2↑
20 00262'125415      INC#         1,1,SNR    ; THEN ADJUST TO 2↑16-2;
21 00263'126120      ADCZL        1,1        ;
22 00264'045402      STA          1,CLATOP,3  ;
23 00265'030040      LDA          2,CUR      ; INSERT CORCUT IN DELAY CHA
24 00266'021045      LDA          0,HDELAY,2  ;
25 00267'041400      STA          0,NEXT,3    ; NEXT.COROUT:= DELAY HEAD
26 00270'055045      STA          3,HDELAY,2  ; DELAY HEAD:= COROUT;
27 00271'176400      SUB          3,3        ;
28 00272'055014      STA          3,TIMER,2   ;
29 00273'002435      JMP          @PNEXTC    ; GOTC NEXTC;
30      ;
31 00274'001206'PEXITC:  EXITC           ; REF EXITC
32 00275'001212'PWEV:   CWEVE          ; REF CWEVE
33 00276'001100'PTESTU: TESTU           ; REF TESTU

```

```

01 ;
02 ; ENTRY POINT CWANSWER:
03 ;
04 ; ENTRY: AC2=BUF, AC3=LINK;
05 ; EXIT: AC1 UNCH, AC2=BUF, AC3=COROUT;
06 ;
07 00277'060277 CWA: INTDS ; COROUTINE WAIT ANSWER:
08 00300'054675 STA 3,AC3SAVE ; SAVE AC3;
09 00301'004662 JSR SETEX ; SET EXIT;
10 00302'004675 JSR GENTU ; GENERATE TESTOUTPUT(
11 00303'000405 1B7+5 ; KIND 5,INHIBIT 1);
12 00304'051402 STA 2,CLATOP,3 ; SAVE BUF IN LATEST OPERAT
13 00305'030040 LDA 2,CUR ; TEST FOR CALL FROM BASIC I/
14 00306'021401 LDA 0,CEXIT,3 ;
15 00307'034221 LDA 3,OPEN-GOS ;
16 00310'162520 SURZL 3,0 ; IF SAVED LINK < ADDR(OPEN)
17 00311'021033 LDA 0,PC,2 ; BEGIN SAVE PC: (MUSIL)
18 00312'125003 MOV 1,1,SNC ;
19 00313'000410 JMP CINAC ;
20 00314'006762 JSR @PTESTU ; IF NEW FACILITIES
21 00315'000404 JMP CCSAV ;
22 00316'135000 MOV 1,3 ; THEN CSPC.COROUT:= PC.CU
23 00317'041417 STA 0,ZFIRST,3 ;
24 00320'000403 JMP CINAC ; ELSE ZFIRST.ZONE:= PC.CU
25 00321'035041 CCSAV: LDA 3,CCOROUT,2 ;
26 00322'041405 STA 0,CSPC,3 ; END;
27 00323'035041 CINAC: LDA 3,CCOROUT,2 ; INSERT COROUT IN ANSWER CHA
28 00324'021044 LDA 0,HANSWER,2 ;
29 00325'041400 STA 0,NEXT,3 ; CHAIN.COROUT:= ANSWER HEA
30 00326'055044 STA 3,HANSWER,2 ; ANSWER HEAD:= COROUT;
31 00327'002401 JMP @PNEXTC ; GOTO NEXTC;
32
33 00330'001153'PNEXTC: NEXTC ; REF NEXTC
34 00331'001163'PTEXTIT: TEXTIT ; REF TEXTIT

```

```

01      ; COROUTINE MONITOR
02      ;
03      ;
04      ;
05      ; ENTRY POINT PASS:
06      ;
07      ;   ENTRY:                               AC3=LINK;
08      ;   EXIT:  AC1 UNCH, AC2=CLATOP, AC3=COROUT;
09      ;
10 00332'060277 PASS:      INTDS                ; PASS: DISABLE;
11 00333'054642          STA      3,AC3SAVE      ;   SAVE AC3;
12 00334'004627          JSR      SETEX          ;   SET EXIT;
13 00335'004642          JSR      GENTU          ;   GENERATE TESTOUTPUT(
14 00336'001006          287+6                  ;     KIND 6,INHIBIT 2);
15 00337'006402          JSR      @RACTCC        ;   INSERT CURRENT IN ACTIVE
16 00340'002735          JMP      @PWEV         ;   GOTC INSPECT EVENTS;
17 00341'000105'RACTCC: ACTCC                ;   REF ACTCC
18      ;
19      ;
20      ; ENTRY POINT WAITSEM:
21      ;
22      ;   ENTRY: AC1=SEM,                       AC3=LINK;
23      ;   EXIT:  AC1=SEM, AC2=LATOP, AC3=COROUT;
24      ;
25 00342'060277 SWAIT:    INTDS                ; WAIT SEMAFOR: DISABLE;
26 00343'054632          STA      3,AC3SAVE      ;   SAVE AC3;
27 00344'004617          JSR      SETEX          ;   SET EXIT;
28 00345'004632          JSR      GENTU          ;   GENERATE TESTOUTPUT(
29 00346'010004          1687+4                 ;     KIND 4,INHIBIT 16);
30 00347'004655          JSR      CWAIT         ;   COMMON WAIT;
31 00350'016626          DSZ      @SAVSEM        ;   SEM OPEN:
32 00351'016625          DSZ      @SAVSEM        ;     SEM VALUE:= SEM VALUE -2;
33 00352'002722          JMP      @PEXITC       ;   GOTC EXITC;
34      ;
35      ;
36      ; ENTRY POINT WAITCH:
37      ;
38      ;   ENTRY: AC1=SEM,                       AC3=LINK;
39      ;   EXIT:  AC1=SEM, AC2=LATOP, AC3=COROUT;
40      ;
41 00353'060277 CHWAIT:  INTDS                ; WAIT CHAINED: DISABLE;
42 00354'054621          STA      3,AC3SAVE      ;   SAVE AC3;
43 00355'004606          JSR      SETEX          ;   SET EXIT;
44 00356'004621          JSR      GENTU          ;   GENERATE TESTOUTPUT(
45 00357'010003          1687+3                 ;     KIND 16, INHIBIT 3);
46 00360'004644          JSR      CWAIT         ;   COMMON WAIT;
47 00361'021000          LDA      0,0,2         ;   SEM OPEN: AC2= OP.SEM;
48 00362'101140          MOVOL   0,0           ;
49 00363'042613          STA      0,@SAVSEM      ;   SEM VALUE:=CHAIN.OP<1+1;
50 00364'051402          STA      2,CLATOP,3     ;   SAVE LATEST OP IN COROUT;
51 00365'002744          JMP      @PTEXIT       ;   GOTC TEXIT;
52

```


↑ 0017 CM002

```
01 ; ENTRY POINT SIGGENERAL:
02 ;
03 ; ENTRY: AC1=SEM AC2=OP AC3=LINK
04 ; EXIT: AC1=SEM AC2=CLATOP AC3=COROUT
05
06 00366'060277 SIGNGEN:INTDS ; SIGNAL GENERAL:
07 00367'054606 STA 3,AC3SAVE ; DISABLE;
08 00370'006405 JSR @XSETEX ; SET EXIT;
09 00371'004606 JSR GENTO ; GENERATE TESTOUTPUT(LONG,
10 00372'140213 1B0+64B7+1B8+11 ; INHIBIT 64,BACKWARD,KIND 11)
11 00373'004536 JSR GENSIG ; GENERAL SIGNAL;
12 00374'002700 JMP @PEXITC ; GOTO EXITC;
13 ;
14 00375'000163'XSETEX: SETEX ; REF SETEX
15 00376'000175'XAC3S: AC3SAVE ; REF AC3SAVE
16 00377'000177'XGENTO: GENTO ; REF GENTO
17 ;
18 ; ENTRY POINT WAITGENERAL:
19 ;
20 ; ENTRY: AC0=SEM AC1=NRUFNO AC2=TIMEOUT AC3=LINK
21 ; EXIT: AC0 UNDEF AC1=REM.TIME AC2=CLATOP AC3=COROUT
22 ;
23
24 00400'060277 GENWAIT:INTDS ; WAIT GENERAL:
25 00401'056775 STA @3,XAC3S ; DISABLE;
26 00402'006773 JSR @XSETEX ; SET EXIT;
27 00403'006774 JSR @XGENTO ; GENERATE TESTOUTPUT(
28 00404'010014 16B7+12 ; INHIBIT 16, KIND 12);
29 00405'051402 STA 2,CLATOP,3 ; SAVE TIMEOUT IN CLATOP.COROUT
30 00406'025776 LDA 1,OPMASK,3 ; AC1:= OPMASK.COROUT;
31 00407'111000 MOV 0,2 ; AC2:= ADDR OF GEN.SEM;
32 00410'035001 LDA 3,NXTOP,2 ; AC3:= HEAD OF OPERATION QUEUE;
33 00411'054515 STA 3,SAV2 ; SAVE ADDR OF LAST OPERATION
34 00412'050515 STA 2,SAV3 ; SAVE ADDR OF GEN.SEM;
35 00413'175005 MOV 3,3,SNR ; IF QUEUE EMPTY THEN
36 00414'000411 JMP TTOUT ; GOTO TEST TIMEOUT;
37 00415'031400 CREP6: LDA 2,NEXT,3 ; EXAMINE NEXT OPERATION:
38 00416'021377 LDA 0,OPTYPE,2 ;
39 00417'123404 AND 1,0,SZR ; IF OPTYPE AND OPMASK THEN
40 00420'000470 JMP OPFOUND ; GOTO OPERATION FOUND;
41 00421'020505 LDA 0,SAV2 ; ELSE
42 00422'155000 MOV 2,3 ;
43 00423'162404 SUB 3,0,SZR ; IF NOT LAST OPERATION THEN
44 00424'000771 JMP CREP6 ; GOTO EXAMINE NEXT;
45 00425'034017 TTOUT: LDA 3,COROUT ; OPERATION NOT FOUND:
46 00426'021402 LDA 0,CLATOP,3 ; TEST TIMEOUT:
47 00427'101005 MOV 0,0,SNR ; IF IMMEDIATE THEN
48 00430'000454 JMP RETTOP ; GOTO RETURN TIMEOUT OPERATION
49 00431'101405 INC 0,0,SNR ; IF NO TIMEOUT WANTED THEN
50 00432'041402 STA 0,CLATOP,3 ; CLATOP.COROUT:= 0;
51 00433'030040 LDA 2,CUR ;
52 00434'020127 LDA 0,.12 ; AC0:= MESS.OP AND INTR.OP TYPE
53 00435'123405 AND 1,0,SNR ; IF OPMASK.COROUT AND AC0 THEN
54 00436'000404 JMP INSC ; BEGIN
55 00437'020470 LDA 0,SAV3 ; MSEM.CUR:= SAVED ADDR OF GEN.
56 00440'041051 STA 0,MSEM,2 ; MCOROUT.CUR:= COROUT;
57 00441'055052 STA 3,MCOROUT,2 ; END;
```

```

↑ 0018 C4002
01 00442'030465 INSC: LDA 2,SAV3 ; INSERT COROUT IN GEN.SEM;
02 00443'020465 LDA 0,XNEXTC ; AC2:= ADDR OF NEXTCO FIELD
03 00444'113001 ADD 0,2,SKP ; IN GEN.SEM;
04 00445'111000 CREF7: MOV 0,2 ;
05 00446'021000 LDA 0,NEXT,2 ; AC0:= NEXT.AC2;
06 00447'101004 MOV 0,0,SZR ; IF NOT END OF CHAIN THEN
07 00450'000775 JMP CREF7 ; GO ON;
08 00451'055000 STA 3,NEXT,2 ; INSERT COROUT IN END OF CHAIN;
09 00452'041400 STA 0,NEXT,3 ; SET END OF CHAIN IN COROUT;
10 00453'030040 LDA 2,CUR ; ADJUST TIMEOUT:
11 00454'020075 LDA 0,RTIME+1 ;
12 00455'025042 LDA 1,LATIME,2 ;
13 00456'122400 SUB 1,0 ; AC0:= RTIME-LATIME.CUR;
14 00457'025402 LDA 1,CLATOP,3 ;
15 00460'125005 MOV 1,1,SNR ; IF NO TIMEOUT THEN
16 00461'002647 JMP @PNEXTC ; GOTO ACTIVATE NEXT COROUTINE;
17 00462'107023 ADDZ 0,1,SNC ; AC1:= TIMEOUT+AC0;
18 00463'125415 INC# 1,1,SNR ; IF OVERFLOW OR AC1=2↑16-1 THEN
19 00464'126120 ADCZL 1,1 ; AC1:= 2↑16-2;
20 00465'045402 STA 1,CLATOP,3 ; CLATOP.COROUT:= AC1;
21 00466'034441 LDA 3,SAV3 ;
22 00467'021402 LDA 0,TIMEOUT,3 ; AC0:= TIMEOUT.SEM
23 00470'101004 MOV 0,0,SZR ; IF AC0=0 THEN
24 00471'000405 JMP SETTO ; BEGIN
25 00472'021045 LDA 0,HDELAY,2 ; INSERT SEM IN DELAY CHAIN:
26 00473'041400 STA 0,NEXT,3 ; NEXT.SEM:= DELAY HEAD;
27 00474'055045 STA 3,HDELAY,2 ; DELAY HEAD:= SEM; AC0:= AC1;
28 00475'121000 MOV 1,0 ; END;
29 00476'122433 SETTO: SUBZ# 1,0,SNC ; SET TIMEOUT:
30 00477'002631 JMP @PNEXTC ; IF AC1<=AC0 THEN
31 00500'045402 STA 1,TIMEOUT,3 ; BEGIN
32 00501'126400 SUB 1,1 ; TIMEOUT.SEM:= AC1;
33 00502'045014 STA 1,TIMER,2 ; TIMER.CUR:= 0;
34 00503'002625 JMP @PNEXTC ; END; GOTO ACTIVATE NEXT COR;
35 ; RETURN TIMEOUT OPERATION;
36 00504'030420 RETTOP: LDA 2,TOOPA ; AC2:= ADDR OF TIMEOUT OPERATIO
37 00505'041404 TEXRET: STA 0,CAC1SAVE,3 ; CAC1SAVE.COROUT:= AC0;
38 00506'051402 STA 2,CLATOP,3 ; CLATOP.COROUT:= AC2;
39 00507'002622 JMP @PTEXTIT ; GOTO TEST EXIT;
40 ;
41 00510'021000 OPFOUND: LDA 0,NEXT,2 ; OPERATION FOUND:
42 00511'041400 STA 0,NEXT,3 ; REMOVE IT FROM CHAIN;
43 00512'156415 SUB# 2,3,SNR ; AC3:= PREVIOUS OP;
44 00513'176400 SUB 3,3 ;
45 00514'165000 MOV 3,1 ; AC1:=IF AC3=OP THEN 0 ELSE AC3
46 00515'034412 LDA 3,SAV3 ; AC3:= SAVED GEN.SEM ADDR;
47 00516'020410 LDA 0,SAV2 ;
48 00517'142405 SUB 2,0,SNR ; IF OP=LAST OP IN CHAIN THEN
49 00520'045401 STA 1,NXTOP,3 ; NXTOP.GEN.SEM:= AC1;
50 00521'034017 LDA 3,COROUT ;
51 00522'021402 LDA 0,CLATOP,3 ; AC0:= TIMEOUT (CALL VALUE);
52 00523'000762 JMP TEXRET ; GOTO TEST EXIT RETURN;
53 ;
54 00524'000526' TOOPA: .+2 ; ADDRESS OF TIMEOUT OPERATION;
55 00525'000001 1B15 ; TIMEOUT OPERATION TYPE
56 00526'000000 SAV2: 0 ; LOCAL WORK LOCATIONS:
57 00527'000000 SAV3: 0 ;
58 00530'000003 XNEXTC: NXTCO-NEXT ; REL. OF NEXTCO FIELD IN GEN.SEM

```

```

01 ;
02 ; PROCEDURE GENERAL SIGNAL
03 ;
04 ;
05 ; CALL:          AC1=SEM  AC2=OP  AC3=LINK
06 ; RETURN:  AC0, AC1,    AC2,    AC3 UNDEFINED
07 ;
08
09 00531'054776 GENSIG: STA      3, SAV3          ; GENERAL SIGNAL:
10 00532'021377          LDA      0, OPTYPE, 2    ; SAVE LINK; AC0:= OPTYPE.OP;
11 00533'050773          STA      2, SAV2          ; SAVE OP;
12 00534'030774          LDA      2, XNEXTC      ; AC2:= ADDR OF NEXTCO FIELD IN
13 00535'133000          ADD      1, 2          ; LOOK FOR COR. WAITING FOR OP:
14 00536'031000 CREP8:  LDA      2, NEXT, 2      ; AC2:= NEXT.AC2;
15 00537'151005          MOV      2, 2, SNR      ; IF CHAIN ENDS THEN BEGIN
16 00540'000413          JMP      CHOP          ;   COR. NOT FOUND: GOTO CHAIN O
17 00541'035376          LDA      3, OPMASK, 2    ;
18 00542'117405          AND      0, 3, SNR      ; IF NOT OPMASK.AC2 AND AC0 THEN
19 00543'000773          JMP      CREP8          ; GOTO EXAMINE NEXT;
20 00544'102400          SUB      0, 0          ;
21 00545'004420          JSR      REMC          ; COROUTINE FOUND: REMOVE IT
22 00546'041004          STA      0, CAC1SAVE, 2    ; FROM SEMAPHORE CHAIN;
23 00547'020757          LDA      0, SAV2          ; CAC1SAVE.CCR:= REMAINING TIMEO
24 00550'041002          STA      0, CLATOP, 2      ; CLATOP.CCR:= SAVED OP;
25 00551'006543          JSR      @PACTCOR      ; INSERT CCR IN ACTIVE QUEUE;
26 00552'002755          JMP      @SAV3          ; RETURN TO LINK;
27 ;
28 00553'131000 CHOP:  MOV      1, 2          ; CHAIN OP TO SEMAPHORE:
29 00554'020752          LDA      0, SAV2          ; AC0:= OP;
30 00555'035001          LDA      3, NXTOP, 2      ; AC3:= NEXTCP.SEM;
31 00556'175005          MOV      3, 3, SNR      ; LAST:=AC3:= IF AC3<>0 THEN AC3
32 00557'115000          MOV      0, 3          ; ELSE AC0;
33 00560'041001          STA      0, NXTOP, 2      ; NEXTOP.SEM:= OP;
34 00561'025400          LDA      1, NEXT, 3      ; NEXT.NEXTOP.SEM:=
35 00562'047001          STA      1, @NXTOP, 2      ; NEXT.LAST;
36 00563'041400          STA      0, NEXT, 3      ; NEXT.LAST:= OP;
37 00564'002743          JMP      @SAV3          ;
38 ;

```

```

01 ;
02 ; PROCEDURE REMOVE COROUTINE FROM GENERAL SEMAPHORE QUEUE
03 ;
04 ; CALL:   AC0=COND           AC1=SEM           AC2=COROUTINE   AC3=LINK
05 ; RETURN: AC0=CLATOP.COR    AC1=RESULT   AC2=COROUTINE   AC3=CUR
06 ;
07 ; COND:   0 REMOVE, <>0 REMOVE IF 0<CLATOP.COR<=COND
08 ; IF COROUTINE(AC2) = 0 A SEARCH FOR MIN CLATOP IS DONE
09
10
11 00565'054522 REMC:   STA      3,RRET           ; REMOVE COROUTINE:
12 00566'040522      STA      0,SAVO          ; SAVE LINK SEM AND
13 00567'044522      STA      1,SAV1          ; COND;
14 00570'141000      MOV      2,0            ; AC0:= COR;
15 00571'034737      LDA      3,XNEXTC       ;
16 00572'137000      ADD      1,3            ; AC3:= ADDR OF NEXTCO.SEM;
17 00573'126000      ADC      1,1            ; MIN TIMEOUT:= 2↑16 - 1;
18 00574'031400 CREP9: LDA      2,NEXT,3     ; LOOK FOR COR:
19 00575'151005      MOV      2,2,SNR        ; AC2:= NEXT.AC2;
20 00576'000473      JMP      REMRET         ; IF NOT FOUND THEN RETURN;
21 00577'142415      SUB#     2,0,SNR         ;
22 00600'000410      JMP      UNCOR          ; IF FOUND THEN GOTO UNCHAIN
23 00601'155000      MOV      2,3            ; ELSE
24 00602'031402      LDA      2,CLATOP,3     ; BEGIN
25 00603'151005      MOV      2,2,SNR        ; IF 0<CLATOP.COR<MIN.TIMEOUT
26 00604'000770      JMP      CREP9          ; MIN.TIMEOUT:= CLATOP.COR;
27 00605'132433      SUBZ#    1,2,SNR        ; GOTO EXAMINE NEXT;
28 00606'145000      MOV      2,1            ; END;
29 00607'000765      JMP      CREP9          ;
30 00610'050502 UNCOR: STA      2,SAVCOR       ; EXAMINE CONDITION:
31 00611'021002      LDA      0,CLATOP,2     ; SAVE COROUTINE;
32 00612'030476      LDA      2,SAVO          ;
33 00613'151005      MOV      2,2,SNR        ; IF COND<>0 THEN
34 00614'000411      JMP      CHOUT         ; BEGIN
35 00615'101005      MOV      0,0,SNR        ; IF CLATOP.COR=0 THEN
36 00616'000452      JMP      NOREX         ; GOTO NOREMOVE EXIT;
37 00617'142427      SUBZ    2,0,SNR        ; REMT:= CLATOP.COR - COND;
38 00620'000404      JMP      PCOUT         ; IF REMT>0 THEN BEGIN
39 00621'030471      LDA      2,SAVCOR       ; CLATOP.COR:= REMT;
40 00622'041002      STA      0,CLATOP,2     ; GOTO NOREMOVE EXIT; END;
41 00623'000445      JMP      NOREX         ; END;
42 00624'102400 PCOUT: SUB      0,0          ; PREPARE CHAIN OUT:
43 00625'030465 CHOUT: LDA      2,SAVCOR     ; CHAIN CORROUTINE OUT OF SEM:
44 00626'031000      LDA      2,NEXT,2       ;
45 00627'051400      STA      2,NEXT,3       ;
46 00630'030461      LDA      2,SAV1         ;
47 00631'031002      LDA      2,TIMEOUT,2    ;
48 00632'112433      SUBZ#    0,2,SNR        ; IF CLATOP.COR>TIMEOUT.SEM
49 00633'000434      JMP      REMEX         ; THEN EXIT ELSE
50 00634'031400 CREP0: LDA      2,NEXT,3     ; UPDATE SEMAPHORE TIMER:
51 00635'155005      MOV      2,3,SNR        ;
52 00636'000407      JMP      UPDST         ; AC1:= MINIMUM OF
53 00637'031402      LDA      2,CLATOP,3     ; CORROUTINE TIMEOUTS;
54 00640'151005      MOV      2,2,SNR        ;
55 00641'000773      JMP      CREP0         ;
56 00642'132433      SUBZ#    1,2,SNR        ;
57 00643'145000      MOV      2,1            ;
58 00644'000770      JMP      CREP0         ;

```

```

01
02
03 00645'034040 UPDST: LDA 3,CUR ; ADJUST TIMEOUT:
04 00646'051414 STA 2,TIMER,3 ; TIMER.CUR:= 0;
05 00647'125415 INC# 1,1,SNR ;
06 00650'126400 SUB 1,1 ;
07 00651'030440 LDA 2,SAV1 ; AC2:= SEM;
08 00652'045002 STA 1,TIMEOUT,2 ; TIMEOUT.SEM:= AC1;
09 00653'125004 MOV 1,1,SZR ; IF TIMEOUT.SEM>0 THEN
10 00654'000413 JMP REXEX ; EXIT ELSE
11 00655'024436 LDA 1,XDELAY ; REMOVE SEM FROM DELAY QUEUE:
12 00656'137001 ADD 1,3,SKP ;
13 00657'135000 CREP1: MOV 1,3 ;
14 00660'025400 LDA 1,NEXT,3 ;
15 00661'125005 MOV 1,1,SNR ; IF NOT FOUND THEN
16 00662'000405 JMP REXEX ; EXIT ELSE
17 00663'146414 SUB# 2,1,SZR ; IF NOT THIS ONE THEN
18 00664'000773 JMP CREP1 ; GOTO TRY NEXT IN CHAIN;
19 00665'025000 LDA 1,NEXT,2 ; FOUND:
20 00666'045400 STA 1,NEXT,3 ; CHAIN IT OUT;
21 00667'126400 REMEX: SUB 1,1 ; EXIT FROM REMOVE:
22 00670'020422 NOREX: LDA 0,SAVCOR ;
23 00671'030040 REMRET: LDA 2,CUR ;
24 00672'031042 LDA 2,LATIME,2 ; ELAPSED TIME:=
25 00673'034075 LDA 3,RTIME+1 ; AC3:= RTIME-LATIME.CUR;
26 00674'156400 SUB 2,3 ;
27 00675'111004 MOV 0,2,SZR ; AC2:=CORCUTINE;
28 00676'021002 LDA 0,CLATOP,2 ; AC0:= CLATOP.COR;
29 00677'101005 MOV 0,0,SNR ; IF AC0=0 (NO TIMEOUT) THEN
30 00700'000405 JMP NOTOR ; AC0:=-1 (INFINITE) ELSE
31 00701'162423 SUBZ 3,0,SNC ; AC0:=AC0-AC3;
32 00702'102400 SUB 0,0 ;
33 00703'034040 RCUR: LDA 3,CUR ; AC3:=CUR;
34 00704'002403 JMP @RRET ; RETURN;
35 00705'102000 NOTOR: ADC 0,0 ;
36 00706'000775 JMP RCUR ;
37 ;
38 00707'000000 RRET: 0 ; SAVE LINK FOR REMC;
39 00710'000000 SAV0: 0 ; LOCAL WORK LOCATIONS:
40 00711'000000 SAV1: 0 ;
41 00712'000000 SAVCOR: 0 ;
42 00713'000045 XDELAY: HDELAY-NEXT ; REL. OF DELAY HEAD IN PROCESS
43 ;
00714'000106'PACTC: ACTCOR ; REF ACTCOR

```

```

01 ;
02 ; PROCEDURE SCAN DELAYQUEUE
03 ;
04 ;
05 ; AC1=DEVICE, AC2=BUF AC3=LINK
06 ; RETURN: AC0=MIN DELAY AC1, AC2 UNCHANGED AC3=CUR
07 ;
08
09 00715'060277 SCAND: INTDS ; SCAN DELAY QUEUE:
10 00716'054553 STA 3,SRET ; DISABLE; SAVE LINK
11 00717'044551 STA 1,SAC1 ; AC1 AND AC2;
12 00720'050547 STA 2,SAC2 ;
13 00721'034040 LDA 3,CUR ;
14 00722'020075 LDA 0,RTIME+1 ; AC0:= TIMEDIFF:=
15 00723'025442 LDA 1,LATIME,3 ; RTIME - LATIME.CUR;
16 00724'122404 SUB 1,0,SZR ; IF AC0=0 THEN
17 00725'000452 JMP CUPD ; BEGIN
18 00726'021414 LDA 0,TIMER,3 ; AC0:= TIMER.CUR;
19 00727'101004 MOV 0,0,SZR ; IF AC0<>0 THEN
20 00730'000417 JMP SCANEX ; GOTO RETURN;
21 00731'102000 FMIND: ADC 0,0 ; FIND MINIMUM DELAY:
22 00732'024761 LDA 1,XDELAY ;
23 00733'137000 ADD 1,3 ; MINDELAY:= 2↑16 - 1;
24 00734'035400 CREP2: LDA 3,NEXT,3 ; WHILE NOT END OF DELAY QUEUE
25 00735'175005 MOV 3,3,SNR ; DO
26 00736'000405 JMP ADJUD ;
27 00737'025402 LDA 1,CLATOP,3 ; IF CLATOP.ELEM<MINDELAY TH
28 00740'106433 SUBZ# 0,1,SNR ; MINDELAY:= CLATOP.ELEM;
29 00741'121000 MOV 1,0 ;
30 00742'000772 JMP CREP2 ;
31 00743'101415 ADJUD: INC# 0,0,SNR ; ADJUST MINDELAY:
32 00744'102120 ADCZL 0,0 ; IF NO DELAYS THEN MINDELAY:=
33 00745'034040 LDA 3,CUR ; AC0:= MAX DELAY = 2↑16 - 2;
34 00746'041414 STA 0,TIMER,3 ; TIMER.CUR:= AC0;
35 00747'025443 SCANEX: LDA 1,HACTIV,3 ; IF COROUTINES ARE WAITING
36 00750'125004 MOV 1,1,SZR ; FOR THE CPU THEN
37 00751'000405 JMP EXEVQ ; GOTO EXAMINE EVENTS;
38 00752'024516 SCANR: LDA 1,SAC1 ; RESTORE AC1;
39 00753'030514 LDA 2,SAC2 ; RESTORE AC2;
40 00754'060177 INTEN ; ENABLE; RETURN;
41 00755'002514 JMP @SRET ; END ELSE GOTO RTC UPDATED;
42 00756'024124 EXEVQ: LDA 1,EVENT ; EXAMINE EVENT QUEUE:
43 00757'167000 ADD 3,1 ; AC1:= EVENTQUEUE HEAD.CUR
44 00760'030507 LDA 2,SAC2 ;
45 00761'151005 MOV 2,2,SNR ; IF BUF=0 THEN
46 00762'131000 MOV 1,2 ; BUF:= EVENTQUEUE HEAD.CUR;
47 00763'031000 LDA 2,NEXT,2 ; BUF:= NEXT.BUF;
48 00764'132414 SUB# 1,2,SZR ; IF BUF<>EVENTQUEUE HEAD.CUR TH
49 00765'000765 JMP SCANR ; GOTO EXIT;
50 00766'024502 LDA 1,SAC1 ; DEV:= SAVED AC1;
51 00767'125005 MOV 1,1,SNR ; IF DEV=0 THEN
52 00770'000563 JMP NEXTC ; GOTO ACTIVATE NEXT COROUTINE;
53 00771'030045 LDA 2,TABLE ;
54 00772'133000 ADD 1,2 ;
55 00773'031000 LDA 2,+0,2 ; INT:= DEVTA(DEV);
56 00774'151222 MOVZR 2,2,SZC ; IF INT(15) THEN
57 00775'000755 JMP SCANR ; GOTO EXIT
58 00776'000555 JMP NEXTC ; ELSE GOTO ACTIVATE NEXT COROUT

```

```
01 ; RTC HAS UPDATED RTIME;
02 00777'040473 CUPD: STA 0,STDIF ; SAVE AC0;
03 01000'024075 LDA 1,RTIME+1 ;
04 01001'045442 STA 1,LATIME,3 ; UPDATE LATIME.CUR;
05 01002'024711 LDA 1,XDELAY ;
06 01003'137000 ADD 1,3 ; AC3:= ADDR OF DELAY HEAD;
07 01004'031400 CRP3: LDA 2,NEXT,3 ; DECREASE TIMER COUNTS;
08 01005'151005 MOV 2,2,SNR ; IF END CHAIN THEN
09 01006'000457 JMP ENDSC ; GOTO END SCAN;
10 01007'025377 LDA 1,CIDENT,2 ;
11 01010'125405 INC 1,1,SNR ; IF CIDENT.ELEM=-1 THEN
12 01011'000416 JMP DECRS ; GOTO DECREASE SEM COUNTS;
13 01012'025002 LDA 1,CLATOP,2 ; COROUTINE IN DELAY QUEUE:
14 01013'106427 SUBZ 0,1,SBN ; REMT:= CLATOP.COR-TIMEDIFF;
15 01014'000404 JMP CDEXP ; IF REMT>0 THEN
16 01015'045002 STA 1,CLATOP,2 ; BEGIN
17 01016'155000 MOV 2,3 ; CLATOP.COR:= REMT; GO ON;
18 01017'000765 JMP CRP3 ; END;
```

```

01
02 01020'021000 CDEXP: LDA 0,NEXT,2 ; DELAY EXPIRED:
03 01021'041400 STA 0,NEXT,3 ; REMOVE CCR FROM DELAY QUEUE;
04 01022'054451 STA 3,SELEM ; SAVE AC3;
05 01023'006671 JSR @PACTCOR ; INSERT CCR IN ACTIVE QUEUE;
06 01024'020446 LDA 0,STDIF ;
07 01025'034446 LDA 3,SELEM ; RESTORE AC0 AND AC3;
08 01026'000756 JMP CRP3 ; GOTO EXAMINE NEXT;
09 ;
10 01027'054444 DECRS: STA 3,SELEM ; SEMAPHORE IN DELAY QUEUE:
11 01030'050444 STA 2,SSEM ; SAVE PREVIOUS ELEM AND SEM;
12 01031'145000 MOV 2,1 ; AC1:= SEM;
13 01032'034445 LDA 3,XNXTC ;
14 01033'137001 ADD 1,3,SKP ; AC3:= ADDR. OF NEXTCC FIELD;
15 01034'155000 CRP4: MOV 2,3 ; EXAMINE SEM CHAIN OF CORS:
16 01035'031400 LDA 2,NEXT,3 ;
17 01036'054437 STA 3,SPREV ; SAVE PREVIOUS;
18 01037'006466 JSR @PREMC ; DECREASE CLATOP.CCR;
19 01040'020432 LDA 0,STDIF ; RESTORE TIME DIFF
20 01041'151005 MOV 2,2,SNR ; IF END CHAIN THEN
21 01042'000413 JMP UPDSEM ; GOTO UPDATE SEM TIMEOUT;
22 01043'125004 MOV 1,1,SZR ; IF CCR NOT REMOVED THEN
23 01044'000407 JMP EXNCO ; GOTO EXAMINE NEXT;
24 01045'045004 STA 1,CAC1SAVE,2 ; CCR REMOVED:
25 01046'026430 LDA 1,@PTOOPA ; CAC1SAVE.CCR:= 0;
26 01047'045002 STA 1,CLATOP,2 ; CLATOP.CCR:= ADDR OF TIMEOUT
27 01050'006644 JSR @PACTCOR ; ACTIVATE CCR;
28 01051'030424 LDA 2,SPREV ; AC2:= SAVED PREVIOUS
29 01052'020420 LDA 0,STDIF ; AC0:= SAVED TIME DIFF;
30 01053'024421 EXNCO: LDA 1,SSEM ; AC1:= SAVED SEM;
31 01054'000760 JMP CRP4 ; GOTO EXAMINE NEXT;
32 01055'125415 UPDSEM: INC# 1,1,SNR ; UPDATE SEM TIMEOUT:
33 01056'126400 SUR 1,1 ;
34 01057'030415 LDA 2,SSEM ;
35 01060'045002 STA 1,TIMEOUT,2 ; TIMEOUT.SEM:= MIN.DELAY;
36 01061'034412 LDA 3,SELEM ; RESTORE AC3 AS PREVIOUS ELEM;
37 01062'125004 MOV 1,1,SZR ; IF SEM NOT REMOVED THEN
38 01063'155000 MOV 2,3 ; AC3:= SEM;
39 01064'000720 JMP CRP3 ; GOTO NEXT IN DELAY QUEUE;
40 ;
41 01065'034040 ENDSC: LDA 3,CUR ; END OF SCAN: AC3:= CUR;
42 01066'000643 JMP FMIND ; GOTO FIND MINIMUM DELAY;
43 ;
44 01067'000000 SAC2: 0 ; LOCAL WORK LOCATIONS:
45 01070'000000 SAC1: 0 ;
46 01071'000000 SRET: 0 ;
47 01072'000000 STDIF: 0 ;
48 01073'000000 SELEM: 0 ;
49 01074'000000 SSEM: 0 ;
50 01075'000000 SPREV: 0 ;
51 ;
52 01076'000524'PTOOP: TOOPA ; REF TOOPA
53 01077'000003 XNXTC: NXTCC-NEXT ; REL. OF NEXTCC FIELD IN GEN.SE

```



```
01 ; COROUTINE MONITOR
02 ;
03 ;
04 ;
05 ; PROCEDURE TEST USER
06 ;
07 ; DETERMINES WETHER THE PROCESS CALLING THE COROUTINE
08 ; MONITOR MAKES USE OF OLD OR BOTH OLD AND NEW FACILITIES.
09 ; RETURN IS MADE ACCORDINGLY.
10 ;
11 ;
12 ; CALL: AC3=LINK
13 ; RETURN: AC0 UNCHANGED AC1 UNCHANGED AC2=CUR AC3 UNCHANGED
14 ;
15
16 01100'030040 TESTU: LDA 2,CUR ; TEST USER: AC2:= USER PROCESS
17 01101'031047 LDA 2,TRECORD,2 ;
18 01102'151120 MOVZL 2,2 ; C:= BOTH OLD AND NEW;
19 01103'030040 LDA 2,CUR ;
20 01104'151002 MOV 2,2,SZC ;
21 01105'001400 JMP +0,3 ; IF C THEN RETURN TO LINK
22 01106'001401 JMP +1,3 ; ELSE RETURN TO LINK+1
23 ;
```

```

01 ;
02 ; PROCEDURE GET COROUTINE MESSAGE BUFFER:
03 ;
04 ;
05 ; CALL: AC3=LINK
06 ; RETURN: AC0 UNCHANGED (OP=0) AC1,AC2 UNCHANGED AC3=OP
07 ; =MESSBUF.OP (OP<>0)
08 ;
09 ;
10 01107'060277 GETCMB: INTDS ; GET COROUTINE MESSAGE BUFFER:
11 01110'054761 STA 3,SRET ; SAVE LINK
12 01111'050756 STA 2,SAC2 ; AND AC2;
13 01112'030040 LDA 2,CUR ;
14 01113'035054 LDA 3,CBUFFER,2 ; AC2:= MESSAGE BUFFER SEMAPHORE
15 01114'175005 MOV 3,3,SNR ; IF OPERATION QUEUE EMPTY THEN
16 01115'000405 JMP NORR ; GOTO RETURN WITHOUT;
17 01116'021400 LDA 0,NEXT,3 ;
18 01117'041054 STA 0,CBUFFER,2 ; REMOVE OPERATION FROM QUEUE;
19 01120'020404 LDA 0,.M12 ;
20 01121'163000 ADD 3,0 ; COMPUTE MESS.BUF ADDRESS;
21 01122'030745 NORR: LDA 2,SAC2 ; RESTORE AC2;
22 01123'002746 JMP @SRET ; RETURN TO LINK;
23 ;
24 01124'177764 .M12: -12 ;
25 01125'000565'PREMC: REMC ; REF REMC
26 ;
27 ;
28 ; PROCEDURE UNCHAIN:
29 ;
30 ;
31 ; CALL: AC2=BUF AC3=LINK
32 ; RETURN: AC0,AC1 UNCHANGED AC2=BUF.OP AC3=CUR
33 ;
34 ;
35 01126'060277 UNCHAIN:INTDS ; REMOVE BUF FROM CHAIN:
36 01127'054742 STA 3,SRET ; DISABLE;
37 01130'040737 STA 0,SAC2 ; SAVE LINK AND AC0;
38 01131'034040 LDA 3,CUR ;
39 01132'020420 LDA 0,XBUF ;
40 01133'117000 ADD 0,3 ;
41 01134'141001 MOV 2,0,SKP ; SAVE BUF
42 01135'155000 CREP5: MOV 2,3 ;
43 01136'031402 LDA 2,CHAIN,3 ;
44 01137'151005 MOV 2,2,SNR ; IF CHAIN ENDS THEN
45 01140'000405 JMP UNCRET ; GOTO RETURN;
46 01141'142414 SUR# 2,0,SZR ; IF BUF NOT FOUND THEN
47 01142'000773 JMP CREP5 ; GOTO TRY NEXT IN CHAIN;
48 01143'031002 LDA 2,CHAIN,2 ; BUF FOUND:
49 01144'051402 STA 2,CHAIN,3 ; REMOVE IT FROM CHAIN;
50 01145'030127 UNCRET: LDA 2,.12 ;
51 01146'113000 ADD 0,2 ; AC2:= MESSAGE BUFFER OPERATION
52 01147'034040 LDA 3,CUR ;
53 01150'020717 LDA 0,SAC2 ; RESTORE AC0;
54 01151'002720 JMP @SRET ; RETURN TO LINK;
55 ;
56 01152'000007 XBUF: BUFFER-CHAIN ;

```

```

01          ; COROUTINE MONITOR
02          ;
03          ; ACTIVATE NEXT COROUTINE:
04          ;
05 01153'034040 NEXTC:   LDA      3,CUR          ; NEXTC: AC3:= CURRENT PROCES
06 01154'031443          LDA      2,HACTIV,3      ; AC2:= ACTIV HEAD;
07 01155'151005          MOV      2,2,SNR        ; IF ACTIVE QUEUE EMPTY THE
08 01156'000434          JMP      CWEVE         ; GOTO WAIT EVENT;
09 01157'021000          LDA      0,NEXT,2        ; ACTIVATE FROM ACTIVE QUEUE:
10 01160'041443          STA      0,HACTIV,3      ; ACTIVE HEAD:= NEXT;
11 01161'051441 SCURC:   STA      2,CCOROUT,3     ; CORCUT:= CURR.COROUTINE:=
12 01162'050017          STA      2,COROUT       ;
13          ;
14 01163'004715 TEXTIT: JSR      TESTU         ; TEST EXIT:
15 01164'151001          MOV      2,2,SKP        ; IF ONLY OLD FACILITIES TH
16 01165'000411          JMP      PREPT         ; GOTO PREPARE TESTOUTPUT;
17 01166'021053          LDA      0,CUDEX,2        ;
18 01167'101004          MOV      0,0,SZR        ; IF CUDEX.CUR<>0 THEN
19 01170'007053          JSR      @CUDEX,2        ; CALL USEREXIT;
20 01171'030040          LDA      2,CUR          ;
21 01172'034017          LDA      3,COROUT       ;
22 01173'021052          LDA      0,MCOROUT,2     ; IF COROUT=MCOROUT.CUR THE
23 01174'162405          SUB      3,0,SNR        ; MCOROUT.CUR:= 0;
24 01175'041052          STA      0,MCOROUT,2     ;
25 01176'034017 PREPT:   LDA      3,COROUT       ; AC3:= COROUT;
26 01177'031402          LDA      2,CLATOP,3      ; AC2:= CLATOP.AC3;
27 01200'025404          LDA      1,CAC1SAVE,3     ; AC1:= CA1CSAVE.AC3;
28 01201'060277          INTDS          ;
29 01202'006577          JSR      @PGENTO        ; GENERATE TESTOUTPUT(
30 01203'102007          1B0+4B7+7          ; LONG,KIND 7,INHIBIT 4);
31 01204'060177 ENEX:   INTEN          ; ENABLE;
32 01205'003401          JMP      @CEXIT,3        ; EXIT TO COROUT;
33          ;
34 01206'034017 EXITC:  LDA      3,COROUT       ; EXITC: AC3:= COROUT;
35 01207'031402          LDA      2,CLATOP,3      ; AC2:= CLATOP.AC3;
36 01210'025404          LDA      1,CAC1SAVE,3     ; AC1:= CAC1SAVE.AC3;
37 01211'000773          JMP      ENEX          ; GOTO ENABLE AND EXIT;

```

```

01      ; COROUTINE MONITOR:
02      ;
03      ; CENTRAL WAITING POINT FOR ALL COROUTINES
04      ;
05
06 01212'060177  CWEVE:  INTEN      ; CENTRAL WAIT:  ENABLE;
07 01213'102400      SUB      0,0      ; BUF:= 0;
08 01214'041446      STA      0,TRETURN,3  ; MESSAGE NO.CUR:= 0;
09 01215'126400  CWNEXT:  SUB      1,1      ; WAIT NEXT EVENT:
10 01216'004662      JSR      TESTU      ; AC1:= IF NEW FACILITIES THEN
11 01217'025050      LDA      1,CDEVICE,2  ; CDEVICE.CUR ELSE 0;
12 01220'111000      MOV      0,2      ; AC2:= BUF;
13 01221'006557      JSR      @PSCAN      ; SCAN DELAYQUEUE(AC0=MIN DELAY)
14 01222'006002      WAIT      ; WAIT EVENT, INTERRUPT, OR TIME
15 01223'000767      JMP      CWEVE      ; +0: TIMEOUT:  GOTO CENTRAL WA
16 01224'000510      JMP      CINTER      ; +1: INTERRUPT: GOTO COR. INTER
17 01225'000520      JMP      ANSWER      ; +2: ANSWER:   GOTO ANSWER;
18 01226'011446      ISZ      TRRETURN,3  ; +3: MESSAGE:  MESS-NO:= MESS-
19      ;
20      ; MESSAGE FOUND IN EVENT QUEUE:
21      ;
22 01227'024504      LDA      1,XANSWER      ;
23 01230'137000      ADD      1,3      ; AC3:= ANSWER HEAD;
24 01231'141001      MOV      2,0,SKP      ; SAVE BUF;
25 01232'155000  CREP3:  MOV      2,3      ; LOOK FOR COROUTINE WAITING
26 01233'031400      LDA      2,NEXT,3      ; FOR MESSAGE BY MEANS OF
27 01234'151005      MOV      2,2,SNR      ; CWANSWER(BUF=0);
28 01235'000423      JMP      TWGEN      ; IF NOT FOUND GOTO TEST WAITGEN
29 01236'025002      LDA      1,CLATOP,2      ;
30 01237'125004      MOV      1,1, SZR      ; IF NOT THIS ONE THEN
31 01240'000772      JMP      CREP3      ; TRY THE NEXT ELSE COROUTINE FO
32 01241'025000  EFOUND:  LDA      1,NEXT,2      ; EVENT FOUND:
33 01242'045400      STA      1,NEXT,3      ; REMOVE COROUTINE FROM CHAIN;
34 01243'041002      STA      0,CLATOP,2      ; CLATOP.CORCUT:= SAVED BUF;
35 01244'145000      MOV      2,1      ; RESTORE PC:  (MUSIL)
36 01245'004633      JSR      TESTU      ;
37 01246'131001      MOV      1,2,SKP      ; IF NEW FACILITIES
38 01247'131001      MOV      1,2,SKP      ;
39 01250'000404      JMP      CGSPC      ; THEN SAVED PC:= CSPC.CORCUT
40 01251'035004      LDA      3,CAC1SAVE,2      ;
41 01252'021417      LDA      0,ZFIRST,3      ; ELSE SAVED PC:= ZFIRST.ZONE;
42 01253'000402      JMP      CRPC      ;
43 01254'021005  CGSPC:  LDA      0,CSPC,2      ;
44 01255'034040  CRPC:   LDA      3,CUR      ;
45 01256'041433      STA      0,PC,3      ; PC.CUR:= SAVED PC;
46 01257'000702      JMP      SCURC      ; GOTO SET CURRENT COROUTINE;
47      ;
48 01260'004620  TWGEN:  JSR      TESTU      ; TEST FOR WAITGENERAL CALL:
49 01261'155001      MOV      2,3,SKP      ; IF NEW FACILITIES NOT USED THE
50 01262'000733      JMP      CWNEXT      ; GOTO WAIT FOR NEXT EVENT;
51 01263'031446      LDA      2,TRETURN,3      ;
52 01264'035452      LDA      3,MCOROUT,3      ;
53 01265'175005      MOV      3,3,SNR      ; IF NO SUCH THEN
54 01266'000727      JMP      CWNEXT      ; GOTO WAIT FOR NEXT EVENT;

```

```

↑ 0029 CM002
01 01267'025404 LDA 1,CAC1SAVE,3 ; COROUTINE WAITS FOR MESSAGE:
02 01270'132404 SUB 1,2, SZR ; IF NOT THIS MESSAGE THEN
03 01271'000724 JMP CWNEXT ; GOTO WAIT NEXT EVENT;
04 01272'111000 MOV 0,2 ;
05 01273'004614 JSR GETCMB ; DELIVER MESSAGE:
06 01274'165005 MOV 3,1,SNR ; IF NO MESSOP AVAILABLE THEN
07 01275'000423 JMP ROPER ; RETURN WITHOUT OPERATION;
08 01276'051776 STA 2,ANSWSEM,3 ; MESSOP.ANSWSEM:= BUF;
09 01277'024116 LDA 1,.1B13 ;
10 01300'045777 STA 1,OPTYPE,3 ; MESSOP.OPTYPE:= MESSTYPE;
11 01301'115000 MOV 0,3 ;
12 01302'025004 LDA 1,SENDER,2 ;
13 01303'045404 STA 1,SENDER,3 ; MESSOP.MESSAGE:= BUF.MESSAGE;
14 01304'025005 LDA 1,RECEIVER,2 ;
15 01305'045405 STA 1,RECEIVER,3 ;
16 01306'025006 LDA 1,MESS0,2 ;
17 01307'045406 STA 1,MESS0,3 ;
18 01310'025007 LDA 1,MESS1,2 ;
19 01311'045407 STA 1,MESS1,3 ;
20 01312'025010 LDA 1,MESS2,2 ;
21 01313'045410 STA 1,MESS2,3 ;
22 01314'025011 LDA 1,MESS3,2 ;
23 01315'045411 STA 1,MESS3,3 ;
24 01316'024127 LDA 1,.12 ;
25 01317'167000 ADD 3,1 ; AC1:= MESSOP;
26 01320'034040 ROPER: LDA 3,CUR ; RETURN OPERATION:
27 01321'031452 LDA 2,MCOROUT,3 ; COROUTINE:= MCOROUT.CUR;
28 01322'045452 STA 1,MCOROUT,3 ; SAVE MESSOP;
29 01323'025451 LDA 1,MSEM,3 ; AC1:= MESSAGE SEMAPHORE ADDRESS
30 01324'102400 SUB 0,0 ;
31 01325'006600 JSR @PREMC ; REMOVE MCOROUT FROM MSEM;
32 01326'041004 STA 0,CAC1SAVE,2 ; SAVED AC1.COR:= REM. TIMEOUT;
33 01327'021452 LDA 0,MCOROUT,3 ;
34 01330'041002 STA 0,CLATOP,2 ; CLATOP.CCR:= SAVED MESSOP;
35 01331'051452 STA 2,MCOROUT,3 ;
36 01332'000627 JMP SCURC ; GOTO SET CURRENT COROUTINE;
37 ;
38 01333'000044 XANSWER: HANSWER ; REL OF HEAD OF ANSWER QUEUE
39 ;
40 ;
41 ; INTERRUPT DETECTED:
42 ;
43 ;
44 01334'021452 CINTER: LDA 0,MCOROUT,3 ; INTERRUPT:
45 01335'101005 MOV 0,0,SNR ; IF NO WAITING COROUTINE
46 01336'000615 ANXTC: JMP NEXTC ; THEN GOTO ACTIVATE NEXT;
47 01337'102400 SUB 0,0 ;
48 01340'041450 STA 0,CDEVICE,3 ; CDEVICE.CUR:= 0;
49 01341'024402 LDA 1,IOPA ; AC1:= ADDR OF INTERRUPT OPERAT
50 01342'000756 JMP ROPER ; GOTO RETURN OPERATION;
51 ;
52 01343'001345 IOPA: .+2 ; ADDR OF INTERRUPT OPERATION
53 01344'000010 1B12 ; INTERRUPT OPERATION TYPE
54 ;
55 000012 BASEM= MESS3+1 ;
56 000013 BOPT= BASEM+1 ;
57 000014 BNEXT= BOPT+1 ;

```

```

01 ;
02 ; ANSWER ARRIVED:
03 ;
04 ;
05 01345'021003 ANSWER: LDA 0,SIZE,2 ; ANSWER IN EVENT QUEUE:
06 01346'024130 LDA 1,.13 ;
07 01347'122405 SUB 1,0,SNR ; IF BUF.SIZE=13 THEN
08 01350'000414 JMP CANSW ; GOTO CMESS ANSWER;
09 01351'024762 LDA 1,XANSWER ;
10 01352'137000 ADD 1,3 ; AC3:= ANSWER HEAD;
11 01353'141001 MOV 2,0,SKP ; SAVE BUF;
12 01354'155000 CREP4: MOV 2,3 ;
13 01355'031400 LDA 2,NEXT,3 ; LOOK FOR COROUTINE WAITING
14 01356'151005 MOV 2,2,SNR ; FOR BUF
15 01357'000636 JMP CWNEXT ; IF NOT FOUND THEN GOTO WAIT NE
16 01360'025002 LDA 1,CLATOP,2 ;
17 01361'106414 SUB# 0,1,SZR ; IF NOT THIS ONE THEN
18 01362'000772 JMP CREP4 ; TRY THE NEXT
19 01363'000656 JMP EFOUND ; ELSE FOUND: GOTO FIRE UP COROU
20 01364'021005 CANSW: LDA 0,RECEIVER,2 ; ANSWER TO COROUTINE MESSAGE:
21 01365'100400 NEG 0,0 ;
22 01366'041014 STA 0,BNEXT,2 ; BUF.OP.NEXT:=-BUF.RECEIVER;
23 01367'006005 WAITANSWER ;
24 01370'006407 JSR @PUNC ; RELEASE MESSAGE BUFFER;
25 01371'025376 LDA 1,ANSWSEM,2 ; AC1:= ANSWER SFMAPHORE;
26 01372'021000 LDA 0,NEXT,2 ;
27 01373'041376 STA 0,ANSWSEM,2 ; BUF.OP.ANSWSEM:= RECEIVER;
28 01374'006402 JSR @PGENSIG ; SIGNAL ANSWER SEM;
29 01375'000741 JMP ANXTC ; ACTIVATE COROUTINE;
30 ;
31 01376'000531'PGENSIG: GENSIG ; REF GENSIG
32 01377'001126'PUNC: UNCHAIN ; REF UNCHAIN
33 ;

```

```
01 ; COROUTINE MONITOR
02 ;
03 ; TEST FACILITIES
04
05
06 01400'000715'PSCAN: SCAND ; REF SCAND
07 01401'000177'PGENTO: GENTO ; REF GENTO
08 01402'000175'PAC3SAVE: AC3SAVE ; REF AC3SAVE
09 01403'000163'PSETEX: SETEX ; REF SETEX
10 ;
11 ; ENTRY POINT CTEST:
12 ;
13 ; ENTRY: AC3=LINK;
14 ; EXIT: AC0,1,2 UNCH, AC3=COROUT;
15 ;
16 01404'060277 TEST: INTDS ; TEST: DISABLE;
17 01405'056775 STA 3,@PAC3SAVE ;
18 01406'006775 JSR @PSETEX ; SET EXIT;
19 01407'006772 JSR @PGENTO ; GENERATE TESTOUTPUT(
20 01410'104011 1B0+8B7+9 ; KIND 9, INHIBIT 8);
21 01411'060177 INTEN ;
22 01412'003401 JMP @CEXIT,3 ; GOTC CEXIT.COROUT;
23 ;
24 ; ENTRY POINT CPRINT:
25 ;
26 ; ENTRY: AC2=REF, AC3=LINK;
27 ; EXIT: AC0,1,2 UNDEF, AC3=COROUT;
28 ;
29 01413'060277 TESTPRINT:INTDS ; TESTPRINT: DISABLE;
30 01414'056766 STA 3,@PAC3SAVE ;
31 01415'006766 JSR @PSETEX ; SET EXIT;
32 01416'006763 JSR @PGENTO ; GENERATE TESTOUTPUT(
33 01417'104012 1B0+8B7+10 ; LONG,KIND 10,INHIBIT 8);
34 01420'060177 INTEN ;
35 01421'003401 JMP @CEXIT,3 ; GOTC CEXIT.COROUT;
36 ;
37 ;
38 ; ENTRY POINT CTOUT:
39 ;
40 ; ENTRY: AC2=REF, AC3=LINK;
41 ; EXIT: AC0,1,2 UNCH, AC3=COROUT;
42 ;
43 01422'060277 TESTOUT: INTDS ; TESTOUT: DISABLE;
44 01423'056757 STA 3,@PAC3SAVE ;
45 01424'006757 JSR @PSETEX ; SET EXIT;
46 01425'037401 LDA 3,@CEXIT,3 ;
47 01426'054402 STA 3,TESTK ; TEST KIND:= CALL KIND;
48 01427'006752 JSR @PGENTO ; GENERATE TESTOUTPUT(
49 01430'000000 TESTK: 0 ; TESTKIND);
50 01431'011401 ISZ CEXIT,3 ; SKIP OVER CALL KIND;
51 01432'060177 INTEN ;
52 01433'003401 JMP @CEXIT,3 ; GOTC CEXIT.COROUT;
53 ;
54
55 001434'LAST= .
56
57
```

↑ 0032 CM002

01 ; COROUTINE MONITOR

02 ;

03 ;

04 ; ENTRY POINT TABLE:

05 ;

06	000367	.LOC	CTOP-GOS	;
07	000367	000000	0	;
08	000334	.LOC	CDELAY-GOS	;
09	000334	000246'	CDEL	;
10	000335	.LOC	WAITSEM-GOS	;
11	000335	000342'	SWAIT	;
12	000336	.LOC	WAITCH-GOS	;
13	000336	000353'	CHWAIT	;
14	000337	.LOC	CWANSWER-GOS	;
15	000337	000277'	CWA	;
16	000340	.LOC	CTEST-GOS	;
17	000340	001404'	TEST	;
18	000341	.LOC	CPRINT-GOS	;
19	000341	001413'	TESTPRINT	;
20	000342	.LOC	CTOUT-GOS	;
21	000342	001422'	TESTOUT	;
22	000343	.LOC	SIGNAL-GOS	;
23	000343	000127'	SIGSEM	;
24	000344	.LOC	SIGCH-GOS	;
25	000344	000141'	SIGNCH	;
26	000345	.LOC	CPASS-GOS	;
27	000345	000332'	PASS	;
28	000364	.LOC	CSENDMESSAGE-GOS	;
29	000364	000007'	CSMESS	;
30	000365	.LOC	SIGGEN-GOS	;
31	000365	000366'	SINGEN	;
32	000366	.LOC	WAITGEN-GOS	;
33	000366	000400'	GENWAIT	;

34

35

36

37

.END

0033 CM002

ACOSA	000174'	11/09	11/14	11/17					
AC3SA	000175'	8/08	10/10	10/27	11/10	11/18	14/09	15/08	16/11
		16/26	16/42	17/07	17/15	31/08			
ACTCC	000105'	9/35	16/17						
ACTCO	000106'	9/26	9/36	21/44					
ADJUD	000743'	22/26	22/31						
ANSWE	001345'	28/17	30/05						
ANSWS	177776	6/14	8/19	29/08	30/25	30/27			
ANXTC	001336'	29/46	30/29						
BASEM	000012	29/55	29/56						
BNEXT	000014	29/57	30/22						
BOPT	000013	29/56	29/57						
BYP1	000240'	13/23							
BYP5	000123'	9/44	9/49						
CANSW	001364'	30/08	30/20						
CHUFF	000054	6/27	8/43	8/46	26/14	26/18			
CCSAV	000321'	15/21	15/25						
CDEL	000246'	14/08	32/09						
CDEVI	000050	6/23	6/24	28/11	29/48				
CDEXP	001020'	23/15	24/02						
CGSPC	001254'	28/39	28/43						
CHGP	000553'	19/16	19/28						
CHOUT	000625'	20/34	20/43						
CHWAI	000353'	16/41	32/13						
CINAC	000323'	15/19	15/24	15/27					
CINTE	001334'	28/16	29/44						
COMST	000070'	9/14	10/15	10/31					
CREP0	000634'	20/50	20/55	20/58					
CREP1	000657'	21/13	21/18						
CREP2	000734'	22/24	22/30						

CREP5	001135'	26/42	26/47		
CREP6	000415'	17/37	17/44		
CREP7	000445'	18/04	18/07		
CREP8	000536'	19/14	19/19		
CREP9	000574'	20/18	20/26	20/29	
CRP3	001004'	23/07	23/18	24/08	24/39
CRP4	001034'	24/15	24/31		
CRPC	001255'	28/42	28/44		
CSEND	006364	6/08	32/28		
CSMES	000007'	8/07	32/29		
CSPC	000005	6/21	15/26	28/43	
CTEX	000203'	12/20	12/36		
CTOP	006367	6/11	12/17	12/35	32/06
CTPRE	000206'	12/19	12/24		
CUDEX	000053	6/26	6/27	27/17	27/19
CUPD	000777'	22/17	23/02		
CWA	000277'	15/07	32/15		
CWAIT	000224'	13/11	16/30	16/46	
CWEVE	001212'	14/32	27/08	28/06	28/15
CWNEX	001215'	28/09	28/50	28/54	29/03 30/15
DECRS	001027'	23/12	24/10		
EFOUN	001241'	28/32	30/19		
ENDSC	001065'	23/09	24/41		
ENEX	001204'	27/31	27/37		
EX4	000155'	10/38			
EEVQ	000756'	22/37	22/42		
EXITC	001206'	14/31	27/34		

EXNCC	001053'	24/23	24/30						
FIRST	000000'	7/17	7/20						
FMIND	000731'	22/21	24/42						
GENSI	000531'	17/11	19/09	30/31					
GENTC	000177'	8/10	10/12	10/29	12/16	14/11	15/10	16/13	16/28
		16/44	17/09	17/16	31/07				
GENWA	000400'	17/24	32/33						
GETCM	001107'	8/55	26/10	29/05					
INSC	000442'	17/54	18/01						
IOPA	001343'	29/49	29/52						
LAST	001434'	7/20	31/55						
MCCRO	000052	6/25	6/26	17/57	27/22	27/24	28/52	29/27	29/28
		29/33	29/35	29/44					
MSEM	000051	6/24	6/25	17/56	29/29				
NEXTC	001153'	15/33	22/52	22/58	27/05	29/46			
NOBR	001122'	26/16	26/21						
NOMOP	000060'	8/18	8/48						
NOREX	000670'	20/36	20/41	21/22					
NOTOR	000705'	21/30	21/35						
NXTCC	000003	6/18	18/58	24/53					
NXTOP	000001	6/16	6/17	17/32	18/49	19/30	19/33	19/35	
OPFOU	000510'	17/40	18/41						
OPMAS	177776	6/20	17/30	19/17					
OTYP	177777	6/13	6/14	8/21	17/38	19/10	29/10		
PAC3S	001402'	31/08	31/17	31/30	31/44				
PACTC	000714'	19/25	21/44	24/05	24/27				
PASS	000332'	16/10	32/27						
PCGUT	000624'	20/38	20/42						
PEXIT	000274'	9/27	10/18	10/43	14/14	14/31	16/33	17/12	
PGENS	001376'	30/28	30/31						
PGENT	001401'	27/29	31/07	31/19	31/32	31/48			
PGETC	000066'	8/16	8/55						
PNEXT	000330'	13/28	14/29	15/31	15/33	18/16	18/30	18/34	
PREMC	001125'	24/18	26/25	29/31					
PREPT	001176'	27/16	27/25						
PSCAN	001400'	28/13	31/06						
PSETE	001403'	31/09	31/18	31/31	31/45				
PTEST	000276'	14/33	15/20						
PTEXI	000331'	15/34	16/51	18/39					
PTOOP	001076'	24/25	24/52						
PUNC	001377'	30/24	30/32						
PUNCH	000067'	8/42	8/56						
PEEV	000275'	14/32	16/16						
RACTC	000341'	16/15	16/17						
RCUR	000703'	21/33	21/36						
REMC	000565'	19/21	20/11	26/25					
REME X	000667'	20/49	21/10	21/16	21/21				
REPRE	000671'	20/20	21/23						
REP1	000234'	13/19	13/22						
REP3	000117'	9/45	9/48						
REP4	000151'	10/34	10/37						
REPS	000120'	9/42	9/46						
RET	000173'	9/36	9/51	11/08	11/15	11/16	13/11	13/16	
RETTC	000504'	17/48	18/36						
ROPER	001320'	29/07	29/26	29/50					
RRET	000707'	20/11	21/34	21/38					
SAC1	001070'	22/11	22/38	22/50	24/45				
SAC2	001067'	22/12	22/39	22/44	24/44	26/12	26/21	26/37	26/53
SAVU	000710'	20/12	20/32	21/39					

SAV1	000711'	20/13	20/46	21/07	21/40				
SAV2	000526'	17/33	17/41	18/47	18/56	19/11	19/23	19/29	
SAV3	000527'	17/34	17/55	18/01	18/21	18/46	18/57	19/09	19/26
		19/37							
SAVCO	000712'	20/30	20/39	20/43	21/22	21/41			
SAVSE	000176'	9/14	9/16	9/23	10/17	10/32	10/33	10/40	10/42
		11/19	13/13	13/14	13/17	13/18	13/25	13/27	16/31
		16/32	16/49						
SCAND	000715'	22/09	31/06						
SCANE	000747'	22/20	22/35						
SCANR	000752'	22/38	22/49	22/57					
SCURC	001161'	27/11	28/46	29/36					
SELEM	001073'	24/04	24/07	24/10	24/36	24/48			
SETEX	000163'	8/09	10/11	10/28	11/08	14/10	15/09	16/12	16/27
		16/43	17/14	31/09					
SETTO	000476'	18/24	18/29						
SIGGE	006365	6/09	32/30						
SIGNC	000141'	10/26	32/25						
SIGNG	000366'	17/06	32/31						
SIGSE	000127'	10/09	32/23						
SPREV	001075'	24/17	24/28	24/50					
SRET	001071'	22/10	22/41	24/46	26/11	26/22	26/36	26/54	
SSEM	001074'	24/11	24/30	24/34	24/49				
SODIF	001072'	23/02	24/06	24/19	24/29	24/47			
SWAIT	000342'	16/25	32/11						
TACO	000005	3/13	3/14						
TAC1	000006	3/14	3/15						
TAC2	000007	3/15	3/16	12/28					
TAC3	000010	3/16	3/17	12/34					
TEST	001404'	31/16	32/17						
TESTK	001430'	31/47	31/49						
TESTO	001422'	31/43	32/21						
TESTP	001413'	31/29	32/19						
TESTU	001100'	14/33	25/16	27/14	28/10	28/36	28/48		
TEXTT	001163'	15/34	27/14						
TEXRE	000505'	18/37	18/52						
TIDEN	000002	3/10	3/11						
TIMEO	000002	6/17	6/18	18/22	18/31	20/47	21/08	24/35	
TKIND	000000	3/08	3/09	12/31					
TOCPA	000524'	18/36	18/54	24/52					
TOPT1	000011	3/17	3/18						
TOPT2	000012	3/18	3/19						
TOPT3	000013	3/19	3/20						
TOPT4	000014	3/20	3/21						
TOPT5	000015	3/21	3/22						
TOPT6	000016	3/22							
TPROC	000001	3/09	3/10						
TRET	000223'	12/16	12/20	12/22	12/30	12/38			
TTIM1	000004	3/12	3/13						
TTIME	000003	3/11	3/12						
TTOUT	000425'	17/36	17/45						
TWGEN	001260'	28/28	28/48						
UNCHA	001126'	8/56	26/35	30/32					
UNCOR	000610'	20/22	20/30						
UNCRE	001145'	26/45	26/50						
UPDSE	001055'	24/21	24/32						
UPDST	000645'	20/52	21/03						
WAITG	006366	6/10	32/32						
XAC3S	000376'	17/15	17/25						

XACTI	000126'	9/38	9/52		
XANSW	001333'	28/22	29/38	30/09	
XBUF	001152'	26/39	26/56		
XDELA	000713'	21/11	21/42	22/22	23/05
XCENT	000377'	17/16	17/27		
XEXT	000530'	18/02	18/58	19/12	20/15
XNXTC	001077'	24/13	24/53		
XSETE	000375'	17/08	17/14	17/26	
.M12	001124'	26/19	26/24		