

.TITLE PDP-9/15 EAE TEST FOR SYSTEM EXERCISER

/REV. DATE -- 1/17/71 -- (1)

.EBREL

00000 R 000000 A  
 00001 R 000000 A  
 00002 R 000000 A  
 00003 R 000000 A  
 00004 R 000075 R  
 00005 R 000032 R  
 00006 R 050105 A  
 00007 R 202462 A  
 00010 R 000040 A  
 00011 R 000000 A  
 00020 R 000000 A  
 00021 R 000000 A  
 00022 R 000000 A  
 00023 R 000000 A  
 00024 R 000000 A  
 00025 R 000000 A  
 00026 R 000000 A  
 00027 R 000000 A  
 00030 R 000000 A  
 00031 R 000000 A

UODSW 0  
 UODSW1 0  
 UODSW2 0  
 UODSW3 0  
 .DSA EAEITR  
 .DSA EAEBGN  
 .SIXBT 'EAEPT2'

/HIGH DIVIDEND.  
 /MULTIPLICAND OR LOW DIVIDEND.  
 /MULTIPLIER OR DIVISOR.

40  
 .BLOCK 7

/MASK FOR CHAIN MODE

SYSERR 0  
 ERWC 0  
 ERWD1 0  
 ERWD2 0  
 ERWD3 0  
 ERWD4 0  
 ERWD5 0  
 ERWD6 0  
 ERWD7 0  
 ERWD8 0

/ERROR INDICATOR FOR MONITOR  
 /ERROR WORD COUNT  
 /ERROR  
 /WORDS  
 /  
 /  
 /  
 /  
 /

641002 A  
 664000 A  
 652000 A  
 657122 A  
 644323 A

LACQ=641002  
 GSM=664000  
 LMQ=652000  
 MULS=657122  
 DIVS=644323

00032 R 000000 A  
 00033 R 140020 R  
 00034 R 140021 R  
 00035 R 200711 R  
 00036 R 040217 R  
 00037 R 200032 R  
 00040 R 040075 R  
 00041 R 200000 R  
 00042 R 500712 R  
 00043 R 740200 A  
 00044 R 600050 R  
 00045 R 100202 R  
 00046 R 100342 R  
 00047 R 600056 R  
 00050 R 540713 R  
 00051 R 600062 R  
 00052 R 540714 R  
 00053 R 600065 R  
 00054 R 540712 R  
 00055 R 600070 R

EAEBGN 0  
 DZM  
 DZM  
 LAC  
 DAC  
 LAC  
 LAC  
 LAC  
 LAC  
 AND  
 SZA  
 JMP  
 JMS  
 JMS  
 JMP  
 SAD  
 JMP  
 SAD  
 JMP  
 SAD  
 JMP

SYSERR  
 ERWC  
 (LAC RAN1)

/SAVE RETURN ADDRESS

.+4  
 EXMUL  
 EXDIV  
 TERMIN  
 (100)  
 TEST1  
 (200)  
 TEST2  
 (300)  
 TEST3

.EJECT

```

00056 R 777774 A   TERMIN  LAW    -4           /END OF TEST
00057 R 040020 R           DAC    SYSERR
00060 R 140021 R           DZM    ERWC
00061 R 620075 R           JMP*   EAEITR   /EXIT
00062 R 100120 R   TEST1  JMS    SETRAN
00063 R 100202 R           JMS    EXMUL
00064 R 600056 R           JMP    TERMIN

/
00065 R 100120 R   TEST2  JMS    SETRAN
00066 R 100342 R           JMS    EXDIV
00067 R 600056 R           JMP    TERMIN

/
00070 R 100120 R   TEST3  JMS    SETRAN
00071 R 100202 R           JMS    EXMUL
00072 R 100120 R           JMS    SETRAN
00073 R 100342 R           JMS    EXDIV
00074 R 600056 R           JMP    TERMIN

/
/
/
00075 R 000000 A   EAEITR 0           /MONITOR STORE RETURN
00076 R 620077 R           JMP*   EAEXIT  /GO TO NEXT TEST

/
00077 R 000000 A   EAEXIT 0           /PROGRAM STORES ADDRESS OF NEXT TEST
00100 R 777773 A           LAW    -5           /SIGNAL TO MONITOR
00101 R 040020 R           DAC    SYSERR   /NORMAL BREAKPOINT
00102 R 750000 A           CLA
00103 R 620075 R           JMP*   EAEITR

/
00104 R 000000 A   ERRXIT 0           /ADDRESS STORED WHEN ERROR TO BE PRINTED
00105 R 200104 R           LAC    ERRXIT
00106 R 040077 R           DAC    EAEXIT
00107 R 777776 A           LAW    -2
00110 R 600101 R           JMP    EAEXIT+2  /EXIT

/
00111 R 000000 A   HOLD   0
00112 R 750004 A           LAS
00113 R 500715 R           AND    (40)
00114 R 741200 A           SNA
00115 R 620111 R           JMP*   HOLD
00116 R 100077 R           JMS    EAEXIT
00117 R 600112 R           JMP    .-5
           .EJECT

```

```

00120 R 000000 A SETRAN 0
00121 R 200716 R LAC (JMP RAN,1)
00122 R 040217 R DAC RAN+2
00123 R 200001 R LAC UODSW+1
00124 R 040235 R DAC RAN1
00125 R 200002 R LAC UODSW+2
00126 R 040237 R DAC RAN3
00127 R 200003 R LAC UODSW+3
00130 R 040236 R DAC RAN2
00131 R 620120 R JMP* SETRAN
00132 R 000000 A BADMUL 0
00133 R 200717 R LAC (1)
00134 R 040022 R DAC ERWD1
00135 R 200236 R LAC RAN2
00136 R 040023 R DAC ERWD2
00137 R 200237 R LAC RAN3
00140 R 040024 R DAC ERWD3
00141 R 200672 R LAC HPRODH
00142 R 040025 R DAC ERWD4
00143 R 200674 R LAC LPRODH
00144 R 040026 R DAC ERWD5
00145 R 200673 R LAC HPRODS
00146 R 040027 R DAC ERWD6
00147 R 200675 R LAC LPRODS
00150 R 040030 R DAC ERWD7
00151 R 777771 A LAW -7
00152 R 040021 R DAC ERWC
00153 R 100104 R JMS ERRXIT

```

```

/
00154 R 620132 R JMP* BADMUL
00155 R 000000 A BADDIV 0
00156 R 200720 R LAC (2)
00157 R 040022 R DAC ERWD1
00160 R 200235 R LAC RAN1
00161 R 040023 R DAC ERWD2
00162 R 200237 R LAC RAN3
00163 R 040024 R DAC ERWD3
00164 R 200236 R LAC RAN2
00165 R 040025 R DAC ERWD4
00166 R 200703 R LAC QUOTH
00167 R 040026 R DAC ERWD5
00170 R 200705 R LAC REMH
00171 R 040027 R DAC ERWD6
00172 R 200704 R LAC QUOTS
00173 R 040030 R DAC ERWD7
00174 R 200706 R LAC REMS
00175 R 040031 R DAC ERWD8
00176 R 777770 A LAW -10
00177 R 040021 R DAC ERWC
00200 R 100104 R JMS ERRXIT
00201 R 620155 R JMP* BADDIV

```

.EJECT

```

/MULTIPLY RANDOM NUMBERS (RAN2 X RAN3)
00202 R 740000 A EXMUL NOP
00203 R 200721 R LAC (400000
00204 R 040666 R DAC CTRAN# /RANDOM NUMBER COUNTER 4096
00205 R 100215 R EXMAGN JMS RAN /RANDOM NUMBER GENERATOR
00206 R 100310 R JMS SOFMUL /SOFTWARE MULTIPLY
00207 R 100320 R JMS HARMUL /HARDWARE MULTIPLY
00210 R 100330 R JMS MULCOM /HARDWARE=SOFTWARE
00211 R 100077 R JMS EAEXIT
00212 R 440666 R ISZ CTRAN
00213 R 600205 R JMP EXMAGN
00214 R 620202 R JMP* EXMUL /EXIT

/RANDOM NUMBER GENERATOR
00215 R 000000 A RAN 0
00216 R 100111 R JMS HOLD
00217 R 200235 R LAC RAN1
00220 R 100240 R JMS RDGEN
00221 R 040235 R DAC RAN1 /FIRST NUMBER
00222 R 100240 R JMS RDGEN
00223 R 040236 R DAC RAN2 /SECOND NUMBER
00224 R 100240 R JMS RDGEN
00225 R 040237 R DAC RAN3 /THIRD NUMBER
00226 R 620215 R JMP* RAN

/
/
/
/
/
00227 R 440237 R RAN.1 ISZ RAN3
00230 R 620215 R JMP* RAN
00231 R 440235 R ISZ RAN1
00232 R 620215 R JMP* RAN
00233 R 100120 R JMS SETRAN
00234 R 620215 R JMP* RAN

/
/
/
00235 R 003466 A RAN1 003466
00236 R 153501 A RAN2 153501
00237 R 210762 A RAN3 210762
.EJECT

```

00240	R	000000	A	R0GEN	0	
00241	R	040306	R		DAC	RWRK
00242	R	200274	R		LAC	RANDEX
00243	R	540722	R		SAD	(RANTBL+10
00244	R	741000	A		SKP	
00245	R	600255	R		JMP	RANTAD
00246	R	200723	R		LAC	(RANTBL
00247	R	040274	R		DAC	RANDEX
00250	R	200273	R		LAC	RANCON
00251	R	744010	A		CLLIRAL	
00252	R	741400	A		SZL	
00253	R	340717	R		TAD	(1
00254	R	040273	R		DAC	RANCON
00255	R	220274	R	RANTAD	LAC*	RANDEX
00256	R	340273	R		TAD	RANCON
00257	R	060274	R		DAC*	RANDEX
00260	R	440307	R		ISZ	RANCNT
00261	R	600266	R		JMP	.+5
00262	R	360724	R		TAD*	(0)
00263	R	060274	R		DAC*	RANDEX
00264	R	777700	A		LAW	-100
00265	R	040307	R		DAC	RANCNT
00266	R	200306	R		LAC	RWRK
00267	R	740020	A		RAR	
00270	R	360274	R		TAD*	RANDEX
00271	R	440274	R		ISZ	RANDEX
00272	R	620240	R		JMP*	R0GEN
00273	R	123456	A	RANCON	123456	
00274	R	000305	R	RANDEX	RANTBL+10	
00275	R	654321	A	RANTBL	654321	
00276	R	361416	A		361416	
00277	R	055363	A		055363	
00300	R	546060	A		546060	
00301	R	243035	A		243035	
00302	R	762572	A		762572	
00303	R	453237	A		453237	
00304	R	150214	A		150214	
00305	R	000000	A		0	
00306	R	000000	A	RWRK	0	
00307	R	777700	A	RANCNT	-100	
					.EJECT	

```

/SOFTWARE MULTIPLY (RAN2 X RAN3)
00310 R 740000 A SOFMUL NOP
00311 R 200236 R LAC RAN2
00312 R 100441 R JMS MULT
00313 R 200237 R LAC RAN3 /LOW ORDER IN AC HIGH ORDER IN MP5
00314 R 040675 R DAC LPRODS# /LOW ORDER
00315 R 200701 R LAC MP5
00316 R 040673 R DAC HPRODS /HIGH ORDER
00317 R 620310 R JMP* SOFMUL /EXIT

/HARDWARE MULTIPLY
00320 R 740000 A HARMUL NOP
00321 R 200236 R LAC RAN2 /MULTIPLIER
00322 R 100431 R JMS HMPY
00323 R 200237 R LAC RAN3 /MULTIPLICAND
00324 R 040672 R DAC HPRODH /HIGH ORDER
00325 R 641002 A LACQ
00326 R 040674 R DAC LPRODH /LOW ORDER
00327 R 620320 R JMP* HARMUL /EXIT

/COMPARE PRODUCT OF SOFTWARE + HARDWARE
00330 R 740000 A MULCOM NOP
00331 R 200672 R LAC HPRODH#
00332 R 540673 R SAD HPRODS#
00333 R 741000 A SKP
00334 R 600340 R JMP .+4 /HIGH ORDER NOT EQUAL
00335 R 200674 R LAC LPRODH#
00336 R 540675 R SAD LPRODS
00337 R 600341 R JMP .+2
00340 R 100132 R JMS BADMUL /LOW ORDER NOT EQUAL
00341 R 620330 R JMP* MULCOM
.EJECT

```

```

/DIVIDE RANDOM NUMBERS (RAN1,RAN3)/(RAN2)
00342 R 740000 A EXDIV NOP
00343 R 200721 R LAC (400000
00344 R 040666 R DAC CTRAN /RANDOM NUMBER COUNTER 4096
00345 R 100215 R EXDAGN JMS RAN /RANDOM NUMBER GENERATOR
00346 R 100367 R JMS SOFDIV /SOFTWARE DIVIDE
00347 R 100355 R JMS HARDIV /HARDWARE DIVIDE
00350 R 100400 R JMS DIVCOM /HARDWARE=SOFTWARE
00351 R 100077 R JMS EAEXIT
00352 R 440666 R ISZ CTRAN
00353 R 600345 R JMP EXDAGN
00354 R 620342 R JMP* EXDIV

/HARDWARE DIVIDE
00355 R 740000 A HARDIV NOP
00356 R 200237 R LAC RAN3
00357 R 652000 A LMQ /DIVIDEND LOW ORDER
00360 R 200235 R LAC RAN1 /DIVIDEND HIGH ORDER
00361 R 100417 R JMS HDIVID
00362 R 200236 R LAC RAN2 /DIVISOR
00363 R 040705 R DAC REMH /HARDWARE REMAINDER
00364 R 641002 A LACQ
00365 R 040703 R DAC QUOTH /HARDWARE QUOTIENT
00366 R 620355 R JMP* HARDIV

/
/SOFTWARE DIVIDE (RAN1,RAN3)/(RAN2)
00367 R 740000 A SOFDIV NOP
00370 R 200235 R LAC RAN1 /HIGH ORDER DIVIDEND
00371 R 100514 R JMS DIVIDE
00372 R 200237 R LAC RAN3 /LOW ORDER DIVIDEND
00373 R 200236 R LAC RAN2 /DIVISOR
00374 R 040704 R DAC QUOTS /SOFTWARE QUOTIENT
00375 R 200667 R LAC DVD
00376 R 040706 R DAC REMS /SOFTWARE REMAINDER
00377 R 620367 R JMP* SOFDIV

/
/COMPARE QUOTIENT AND REMAINDERS
00400 R 740000 A DIVCOM NOP
00401 R 200355 R LAC HARDIV /GET LINK FROM SOFTWARE DIVIDE
00402 R 741400 A SZL /CHECK LINK FROM HARDWARE DIVIDE
00403 R 620400 R JMP* DIVCOM /IF LINK = 1 EXIT.
00404 R 741100 A SPA /AC SHOULD BE PLUS FOR EQUAL LINKS
00405 R 620400 R JMP* DIVCOM
00406 R 200703 R LAC QUOTH#
00407 R 540704 R SAD QUOTS#
00410 R 741000 A SKP
00411 R 600415 R JMP .+4 /QUOTIENT NOT EQUAL
00412 R 200705 R LAC REMH#
00413 R 540706 R SAD REMS#
00414 R 600416 R JMP .+2
00415 R 100155 R DVCMER JMS BAODIV /REMAINDER NOT EQUAL
00416 R 620400 R JMP* DIVCOM
.EJECT

```



```

/HARDWARE ARITHMETIC SUBROUTINES
/SIGNED DIVIDE SUBROUTINE
/CALLING SEQUENCE
/DIVIDE IN AC AND MQ
/JMS HDIVIDE
/PICKUP OTHER FACTOR
/

```

```

00417 R 000000 A
00420 R 040710 R
00421 R 420417 R
00422 R 664000 A
00423 R 040426 R
00424 R 200710 R
00425 R 644323 A
00426 R 000000 A
00427 R 440417 R
00430 R 620417 R

```

```

HDIVID 0 /ENTRY TO SUBROUTINE

```

```

DAC TEM#
XCT* HDIVID
GSM
DAC HDIVL
LAC TEM
DIVS

```

```

HDIVL 0 /LOCATION OF DIVISIOR

```

```

ISZ HDIVID
JMP* HDIVID

```

```

/SIGNED MULTIPLY SUBROUTINE
/CALLING SEQUENCE.
/ONE FACTOR IN AC
/JMS HMPY
/PICK UP OTHER FACTOR
/

```

```

/LACXXX ON LAC I XXX

```

```

00431 R 000000 A
00432 R 664000 A
00433 R 040436 R
00434 R 420431 R
00435 R 657122 A
00436 R 000000 A
00437 R 440431 R
00440 R 620431 R

```

```

HMPY 0 /ENTRY TO SUBROUTINE
GSM /FIX MULTIPLIER MAGNITUDE

```

```

DAC *+3
XCT* HMPY
MULS

```

```

0 /LOCATION OF MULTIPLER

```

```

ISZ HMPY
JMP* HMPY

```

```

.EJECT

```

/PDP-15 ONE'S COMPLEMENT SINGLE PRECISION MULTIPLICATION SUBROUTINE  
 /HARDWARE SIMULATION  
 /CALLING SEQUENCE:

/LAC MULTIPLIER

/JMS MULT

/LAC MULTIPLICAND

/RETURN; LOW ORDER PRODUCT IN AC, HIGH ORDER PRODUCT

/IN MP5

00441	R	000000	A	MULT	0
00442	R	140701	R		DZM MP#5
00443	R	741200	A		SNA
00444	R	740000	A		NOP
00445	R	745100	A		SPA!CLL
00446	R	740003	A		CMA!CML
00447	R	040676	R		DAC MP#1
00450	R	420441	R		XCT* MULT
00451	R	741200	A		SNA
00452	R	740000	A		NOP
00453	R	741100	A		SPA
00454	R	740003	A		CMA!CML
00455	R	040677	R		DAC MP#2
00456	R	200725	R		LAC (360000)
00457	R	740010	A		RAL
00460	R	040501	R		DAC MPSIGN
00461	R	777756	A		LAW -22
00462	R	040700	R		DAC MP#3
00463	R	740000	A		NOP
00464	R	200676	R	MP4	LAC MP1
00465	R	740020	A		RAR
00466	R	200701	R		LAC MP5
00467	R	745400	A		SZL!CLL
00470	R	340677	R		TAD MP2
00471	R	740020	A		RAR
00472	R	040701	R		DAC MP5
00473	R	200676	R		LAC MP1
00474	R	740020	A		RAR
00475	R	040676	R		DAC MP1
00476	R	200701	R		LAC MP5
00477	R	440700	R		ISZ MP3
00500	R	600512	R		JMP MPZ+2
00501	R	000000	A	MPSIGN	0
00502	R	040701	R		DAC MP5
00503	R	200676	R		LAC MP1
00504	R	740000	A		NOP
00505	R	400501	R		XCT MPSIGN
00506	R	040676	R		DAC MP1
00507	R	740000	A		NOP
00510	R	440441	R	MPZ	ISZ MULT
00511	R	620441	R		JMP* MULT
00512	R	740000	A		NOP
00513	R	600464	R		JMP MP4

.EJECT

```

/PDP-15 ONE'S COMPLEMENT DIVIDE SUBROUTINE HARDWARE SIMULATION
/CALLING SEQUENCE,
/      LAC HIGH ORDER DIVIDEND /JMS DIVIDE
/      LAC LOW ORDER DIVIDEND
/      LAC DIVISOR
/      RETURN,QUOT. IN AC, REM. IN D#VD
/IF HIGH DIVIDENDS GREATER OF EQUAL TO DIVISOR, DIVISOR TAKES
/PLACE AND LINK IS SET TO 1.
DIVIDE 0 /HIGH ORDER DIVIDEND IN AC
00514 R 000000 A SPA!CLL /IS DIVIDEND POSITIVE
00515 R 745100 A CMA!CML /NO, COMPLEMENT IN AC AND LINK
00516 R 740003 A DAC D#VD /STORE HIGH ORDER DIVIDEND
00517 R 040667 R XCT* DIVIDE /FETCH LOW ORDER DIVIDEND
00520 R 420514 R SZL /DIVIDEND SIGN BIT POSITIVE?
00521 R 741400 A CMA /NO, COMPLEMENT LOW ORDER DIVIDEND
00522 R 740001 A DAC Q#UD /STORE LOW ORDER DIVIDEND
00523 R 040702 R JMS DV5 /DEPOSIT DIVIDEND SIGN BIT INTO DV5M
00524 R 100525 R 0 /REMAINDER HAS SIGN OF DIVIDEND
00525 R 000000 A DV5 ISZ DIVIDE
00526 R 440514 R XCT* DIVIDE /FETCH DIVISOR
00527 R 420514 R SPA /SKIP IF SIGN POSITIVE
00530 R 741100 A CMA!CML /COMPELEMENT AC AND LINK
00531 R 740003 A JMS DV4 /DEPOSIT QUOTIENT
00532 R 100533 R 0 /CONTAINS SIGN BIT OF QUOTIENT
00533 R 000000 A DV4 CLL /CLEAR LINK
00534 R 744000 A DAC D#VS /SAVE DIVISOR
00535 R 040670 R ISZ DIVIDE /INCREMENT TO EXIT ADDRESS
00536 R 440514 R LAW -23 /SET UP "STEP COUNTER"
00537 R 777755 A DAC DV#1
00540 R 040671 R LAC SVCB
00541 R 200607 R DAC SVC0 /SET SAVE CARRY SWITCH TO INITIAL
00542 R 040602 R DZM SVCRY /CLEAR SAVED CARRY.
00543 R 140707 R DSP1 NOP
00544 R 740000 A LAC DVS /FETCH DIVISOR
00545 R 200670 R JMP DV2A-2 /START DIVISION
00546 R 600563 R DV2 LAC QHIB /GET SAVED HI QUOTIENT BIT
00547 R 200601 R RAL /PUT BIT INTO LINK
00550 R 740010 A LAC DVD /GET DIVIDEND
00551 R 200667 R RAL /INSERT HI QUOT INTO DIVIDEND
00552 R 740010 A DAC DVD /STORE NEW DIVIDEND
00553 R 040667 R DSP3 NOP
00554 R 740000 A LAC DCRY /GET LAST CARRY
00555 R 200570 R RAL /PUT INTO LINK
00556 R 740010 A LAC DVS /GET DIVISOR
00557 R 200670 R SZL!CLL /IF LINK IS 1, ADD NEG DIVISOR.
00560 R 745400 A SKP
00561 R 741000 A JMP .+3
00562 R 600565 R CMA /IF LINK IS 0, ADD POS DIVISOR.
00563 R 740001 A TAD (1)
00564 R 340717 R DV2A TAD DVD /ADD DIVISOR (+,-) TO DIVIDEND
00565 R 340667 R DAC DVD /STORE NEW DIVIDEND
00566 R 040667 R JMS .+1
00567 R 100570 R DCRY 0 /SAVE CARRY
00570 R 000000 A

```

```

00571 R 740000 A DSP2 NOP
00572 R 200707 R LAC SVCRY /CHECK LAST
00573 R 741100 A SPA /CARRY,=1 IF OVERFLOW
00574 R 740002 A CML /IF OVERFLOW ERROR LAST CYCLE
/COMPLEMENT THIS INSERT BIT
00575 R 200702 R LAC QUO /GET QUOTIENT
00576 R 740010 A RAL /INSERT CARRY INTO QUOTIENT
00577 R 040702 R DAC QUO /STORE NEW QUOTIENT
00600 R 100601 R JMS .+1 /SAVE HI BIT FOR
00601 R 000000 A QHIB 0 /INSERTION INTO DIVIDEND
00602 R 740040 A SVC0 HLT
00603 R 600617 R JMP SVC2 /1ST - SAVE EXTRA SIGN BIT
00604 R 600614 R JMP SVC1 /2ND - SAVE SIGN BIT, CHECK XSIGN BIT
00605 R 600610 R JMP SVC1A /3RD - CHECK SIGN BIT.
00606 R 600622 R JMP SVC3 /OTHERS - CONTINUE
00607 R 600603 R SVCB JMP SVC0+1
00610 R 200707 R SVC1A LAC SVCRY
00611 R 741100 A SPA
00612 R 600654 R JMP OVRFLO
00613 R 600621 R JMP SVC3-1
00614 R 200707 R SVC1 LAC S#VCRY /TEST SAVED SIGN BIT
00615 R 741100 A SPA /MUST=0
00616 R 600654 R JMP OVRFLO /NOT=0, OVERFLOW
00617 R 200570 R SVC2 LAC DCRY
00620 R 040707 R DAC SVCRY /SAVE CARRY FOR TEST NEXT CYCLE
00621 R 440602 R ISZ SVC0 /INCREMENT SWITCH
00622 R 440671 R SVC3 ISZ DV1 /INCREMENT STEP COUNTER
00623 R 600547 R JMP DV2 /GO TO NEXT DIVIDE CYCLE
/STEP COUNTER=0
00624 R 200570 R LAC DCRY /TEST LAST CARRY
00625 R 741100 A SPA
00626 R 600633 R JMP DV3 /IF=1 NO CORRECTIONS NEEDED
00627 R 740000 A DSP4 NOP
00630 R 200670 R LAC DVS /WAS 0
/ADD (+) DIVISOR TO CORRECT
00631 R 340667 R TAD DVD /DVD VALUE FOR REMAINDER
00632 R 040667 R DAC DVD
00633 R 200525 R DV3 LAC DV5 /CHECK DIVIDEND SIGN
00634 R 740010 A RAL /
00635 R 740000 A DSP5 NOP
00636 R 200667 R LAC DVD /
00637 R 741401 A SZL!CMA /IF MINUS, COMPELEMENT REMAINDER
00640 R 040667 R DAC DVD /
00641 R 200533 R LAC DV4 /CHECK DIVISOR SIGN
00642 R 740010 A RAL
00643 R 200702 R LAC QUO
00644 R 741400 A SZL /IF MINUS, COMPLEMENT QUOTIENT
00645 R 740001 A CMA
00646 R 040702 R DAC QUO
00647 R 200707 R LAC SVCRY /SET LINK TO DETERMINED VALUE
00650 R 740010 A RAL
00651 R 200702 R LAC QUO
00652 R 740000 A DSP6 NOP

```

```
00653 R 620514 R          JMP*   DIVIDE
                          /OVERFLOW OCCURRED
00654 R 200671 R          OVRFLO LAC    DV1
00655 R 340717 R          TAD     (1)
00656 R 740100 A          SMA
00657 R 600633 R          JMP     DV3
00660 R 200601 R          LAC     QHIB      /GET SAVED HI QUOTIENT BIT
00661 R 740010 A          RAL
00662 R 200667 R          LAC     DVD      /PUT INTO DIVIDEND
00663 R 740010 A          RAL
00664 R 040667 R          DAC     DVD      /STORE NEW DIVIDEND
00665 R 600633 R          JMP     DV3      /GO TO ADJUST SIGNS
                          .EJECT
```

.END UODSW

00711 R 000000 R \*L  
00711 R 200235 R \*L  
00712 R 000300 A \*L  
00713 R 000100 A \*L  
00714 R 000200 A \*L  
00715 R 000040 A \*L  
00716 R 600227 R \*L  
00717 R 000001 A \*L  
00720 R 000002 A \*L  
00721 R 400000 A \*L  
00722 R 000305 R \*L  
00723 R 000275 R \*L  
00724 R 000000 A \*L  
00725 R 360000 A \*L

SIZE=00726

NO ERROR LINES

BADDIV	00155 R	BADMUL	00132 R	CTRAN	00666 R	DCRY	00570 R
DIVCOM	00400 R	DIVIDE	00514 R	DIVS	644323 A	DSP1	00544 R
DSP2	00571 R	DSP3	00554 R	DSP4	00627 R	DSP5	00635 R
DSP6	00652 R	DVCMER	00415 R	DVD	00667 R	DVS	00670 R
DV1	00671 R	DV2	00547 R	DV2A	00565 R	DV3	00633 R
DV4	00533 R	DV5	00525 R	EAEBGN	00032 R	EAEITR	00075 R
EAEEXIT	00077 R	ERRXIT	00104 R	ERWC	00021 R	ERWD1	00022 R
ERWD2	00023 R	ERWD3	00024 R	ERWD4	00025 R	ERWD5	00026 R
ERWD6	00027 R	ERWD7	00030 R	ERWD8	00031 R	EXDAGN	00345 R
EXDIV	00342 R	EXMAGN	00205 R	EXMUL	00202 R	GSM	664000 A
HARDIV	00355 R	HARMUL	00320 R	HDIVID	00417 R	HDIVL	00426 R
HMPY	00431 R	HOLD	00111 R	HPRODH	00672 R	HPRODS	00673 R
LACQ	641002 A	LMQ	652000 A	LPRODH	00674 R	LPRODS	00675 R
MPSIGN	00501 R	MPZ	00510 R	MP1	00676 R	MP2	00677 R
MP3	00700 R	MP4	00464 R	MP5	00701 R	MULCOM	00330 R
MULS	657122 A	MULT	00441 R	OVRFLO	00654 R	QHIB	00601 R
QUO	00702 R	QUOTH	00703 R	QUOTS	00704 R	RAN	00215 R
RANCNT	00307 R	RANCON	00273 R	RANDEX	00274 R	RANTAD	00255 R
RANTBL	00275 R	RAN.1	00227 R	RAN1	00235 R	RAN2	00236 R
RAN3	00237 R	RDGEN	00240 R	REMH	00705 R	REMS	00706 R
RWRK	00306 R	SETRAN	00120 R	SOFDIV	00367 R	SQFMUL	00310 R
SVCB	00607 R	SVCRY	00707 R	SVC0	00602 R	SVC1	00614 R
SVC1A	00610 R	SVC2	00617 R	SVC3	00622 R	SYSERR	00020 R
TEM	00710 R	TERMIN	00056 R	TEST1	00062 R	TEST2	00065 R
TEST3	00070 R	UODSW	00000 R	TEST1	00062 R	TEST2	00065 R
UODSW3	00003 R			UODSW1	00001 R	UODSW2	00002 R

UODSW	00000 R	UODSW1	00001 R	UODSW2	00002 R	UODSW3	00003 R
SYSERR	00020 R	ERWC	00021 R	ERWD1	00022 R	ERWD2	00023 R
ERWD3	00024 R	ERWD4	00025 R	ERWD5	00026 R	ERWD6	00027 R
ERWD7	00030 R	ERWD8	00031 R	EAEBGN	00032 R	TERMIN	00056 R
TEST1	00062 R	TEST2	00065 R	TEST3	00070 R	EAEITR	00075 R
EAEXIT	00077 R	ERRXIT	00104 R	HOLD	00111 R	SETRAN	00120 R
BADMUL	00132 R	BADDIV	00155 R	EXMUL	00202 R	EXMAGN	00205 R
RAN	00215 R	RAN.1	00227 R	RAN1	00235 R	RAN2	00236 R
RAN3	00237 R	RDGEN	00240 R	RANTAD	00255 R	RANCON	00273 R
RANDEX	00274 R	RANTBL	00275 R	RWRK	00306 R	RANCNT	00307 R
SOFMUL	00310 R	HARMUL	00320 R	MULCOM	00330 R	EXDIV	00342 R
EXDAGN	00345 R	HARDIV	00355 R	SOFDIV	00367 R	DIVCOM	00400 R
DVCMER	00415 R	HDIVID	00417 R	HDIVL	00426 R	HMPY	00431 R
MULT	00441 R	MP4	00464 R	MPSIGN	00501 R	MPZ	00510 R
DIVIDE	00514 R	DV5	00525 R	DV4	00533 R	OSP1	00544 R
DV2	00547 R	DSP3	00554 R	DV2A	00565 R	DCRY	00570 R
DSP2	00571 R	QHIB	00601 R	SVC0	00602 R	SVCB	00607 R
SVC1A	00610 R	SVC1	00614 R	SVC2	00617 R	SVC3	00622 R
DSP4	00627 R	DV3	00633 R	DSP5	00635 R	DSP6	00652 R
OVRFLO	00654 R	CTAN	00666 R	DVD	00667 R	DVS	00670 R
DV1	00671 R	HPRDGH	00672 R	HPRODS	00673 R	LPRDGH	00674 R
LPRODS	00675 R	MP1	00676 R	MP2	00677 R	MP3	00700 R
MP5	00701 R	QUO	00702 R	QUOTH	00703 R	QUOTS	00704 R
REMH	00705 R	REMS	00706 R	SVCRY	00707 R	TEM	00710 R
LACQ	641002 A	DIVS	644323 A	LMQ	652000 A	MULS	657122 A
GSM	664000 A						