



DECUS

PROGRAM LIBRARY

DECUS NO.	8-402
TITLE	RESEQUENCE
AUTHOR	Howard Wolfington, Department of Defense Computer Institute, Washington Navy Yard, Washington, D. C.
COMPANY	Submitted by: W. Kieswetter Digital Equipment Corporation Washington, D. C.
DATE	December 28, 1970
SOURCE LANGUAGE	PAL-D

ATTENTION

This is a USER program. Other than requiring that it conform to submittal and review standards, no quality control has been imposed upon this program by DECUS.

The DECUS Program Library is a clearing house only; it does not generate or test programs. No warranty, express or implied, is made by the contributor, Digital Equipment Computer Users Society or Digital Equipment Corporation as to the accuracy or functioning of the program or related material, and no responsibility is assumed by these parties in connection therewith.

DECUS

1900



RESEQUENCE

DECUS Program Library Write-up

DECUS NO. 8-402

ADDR	0334	ASCBIN	1505	ASCCNT	1527	ASCLP	1511
ASCWRK	1526	ASK1	0243	ASK2	0422	ASK3	0443
BACKUP	1002	BASE10	1153	BEGIN	0242	BINLN	1152
BUFFER	3400	BUFPTR	0002	CKI	0656	CKJ	0673
CLOSE	1240	COUNT	0333	CRLF	1404	CVTBIN	1025
CVTCLR	1061	CVTCNT	1154	CVTEXT	1043	CVTNEG	1070
CVTNXT	1105	CVTOUT	1101	CVTSUB	1062	CVTTBL	1260
CVTX	1155	EMPTY	1215	EOLINE	1112	ERRWD	0336
EXTND	1732	EXTRCT	0306	GET	1600	GETGO	1605
GETINT	1633	GETIO	1613	INADDR	1654	INBCNT	1652
INBLK	1650	INBPTR	1653	INBUF	2200	INCNT	1502
INCR	0505	INERR	1655	INHOLD	1504	ININFO	1251
INPTR	1503	INPUT	1435	INREAD	1442	INX1	1453
INX2	1474	IOKILL	1641	IONAME	0330	IOPN	0326
LASTIN	0646	LINES	0507	LNACC	0412	LNCK	0465
LNCKLP	0471	LNCK1	0501	LNCNT	0510	LNGET	0403
LNHLD	0720	LNPTR	0506	LNTBL	2600	LSTSIG	1151
MARKPT	1021	MOVE	0614	MSGADR	1545	MSGCNT	1544
MSGOUT	1530	MSGWRK	1543	MSG1	0020	MSG2	0032
MSG3	0053	MSG4	0077	MSG5	0124	MSG6	0141
MSG7	0163	MSG8	0205	NAMEPK	0263	NEW	2033
NEWCNT	2035	NEWLN	2000	NEWLP	2011	NEWNOT	2027
NEWNP	2017	NEXTIN	0626	NOKILL	1244	NOMORE	1200
OLD	2034	OPENX	0314	OPRHL	1434	OPRINT	1413
OPRLP	1420	OPR4	1433	OUTBUF	2400	OUTEXT	1146
OUTHLD	1156	OUTLN	1120	OUT2OK	1137	PACK	0272
PAD	0256	PRINT	1400	PTADDR	1730	PTBLK	1724
PTCNT	1726	PTERR	1731	PTPTR	1727	PUT	1656
PUTERR	1712	PUTEXT	1717	PUTGO	1664	PUTINT	1704
SECTB	0400	SECTD	0600	SIGCHR	0707	START	0504
SUBLN	1000	TBLIN	0416	TEMP	0335	TPNAME	0723
TPOPN	0721	ZWPTR	1150				

/RESEQ WILL RESEQUENCE THE LINE NUMBERS (AND REFERENCES)
 /IN A BASIC PROGRAM. H. WOLVINGTON -- DOD COMPUTER
 /INSTITUTE, OCTOBER 1970.

/PHASE I
 / A. OBTAIN PROGRAM NAME & OPEN FILE
 / B. BUILD TBL OF LINE NUMBERS
 / C. OBTAIN STARTING LINE # AND INCREMENT

/PHASE II
 / D. 'REWIND' INPUT FILE AND OPEN 'RSQ001' AS OUTPUT
 / E. READ LINES FROM INPUT, SUBSTITUTE LNS, & OUTPUT
 / F. DELETE INPUT FROM DISK AND RENAME RSQ001 AS INPUT

```

*0000
0000 5401      JMP I ,+1
0001 0242      BEGIN
0002 0000      BUFPTR, 0          /BUFFER POINTER USED IN SECT
                                   /E (BOTH PAGES)

*0020
0020 7771      MSG1,    -7          /'INPUT:' MSG BLOCK
0021 021        ,+0
0022 0240      TEXT .B
0023 0311      CI
0024 0316      CN
0025 0320      CP
0026 0325      CU
0027 0324      CT
0030 0272      B:
0031 0000      .

0032 7762      MSG2,    -16         /'FILE NOT FOUND' MSG BLOCK
0033 0033      ,+0
0034 0306      TEXT .CF
0035 0311      CI
0036 0314      CL
0037 0305      CE
0040 0240      B
0041 0316      CN
0042 0317      CO
0043 0324      CT
0044 0240      B
0045 0306      CF
0046 0317      CO
0047 0325      CU
0050 0316      CN
0051 0304      CD
0052 0000      .

0053 7757      MSG3,    -21         /'STARTING LINE # ' MSG BLOCK
0054 0054      ,+0
0055 0323      TEXT .CS
0056 0324      CT
0057 0301      CA
0060 0322      CR
0061 0324      CT
-----0003
  
```

0062 0311 CI
 0063 0316 CN
 0064 0307 CG
 0065 0240 B
 0066 0314 CL
 0067 0311 CI
 0070 0316 CN
 0071 0305 CE
 0072 0240 B
 0073 0243 B#
 0074 0240 B
 0075 0272 B:
 0076 0000 .

0077 7756 MSG4,
 0100 0100
 0101 0322
 0102 0301 CA
 0103 0316 CN
 0104 0307 CG
 0105 0305 CE
 0106 0240 B
 0107 0311 CI
 0110 0323 CS
 0111 0240 B
 0112 0261 B1
 0113 0240 B
 0114 0324 CT
 0115 0317 CO
 0116 0240 B
 0117 0262 B2
 0120 0260 B0
 0121 0264 B4
 0122 0266 B6
 0123 0000 .

-22
 .+0
 TEXT .CR

/'RANGE IS 1 TO 2046' MSG BLOCK

0124 7766 MSG5,
 0125 0125
 0126 0311
 0127 0316 CN
 0130 0303 CC
 0131 0322 CR
 0132 0305 CE
 0133 0315 CM
 0134 0305 CE
 0135 0316 CN
 0136 0324 CT
 0137 0272 B:
 0140 0000 .

-12
 .+0
 TEXT .CI

/'INCREMENT:' MSG BLOCK

0141 7761 MSG6,
 0142 0142
 0143 0304
 0144 0311 CI
 0145 0323 CS
 0146 0313 CK
 -----0004

-17
 .+0
 TEXT .CD

/'DISK I/O ERROR ' MSG BLOCK

0147 0240 B
 0150 0311 CI
 0151 0257 B/
 0152 0317 CO
 0153 0240 B
 0154 0305 CE
 0155 0322 CR
 0156 0322 CR
 0157 0317 CO
 0160 0322 CR
 0161 0240 B
 0162 0000 .

0163 7761 MSG7, -17
 0164 0164 .+0
 0165 0314 TEXT ,CL
 0166 0311 CI
 0167 0316 CN
 0170 0305 CE
 0171 0240 B
 0172 0243 B#
 0173 0240 B
 0174 0317 CO
 0175 0326 CV
 0176 0305 CE
 0177 0322 CR
 0200 0306 CF
 0201 0314 CL
 0202 0317 CO
 0203 0327 CW
 0204 0000 ,

/'LINE # OVERFLOW' MSG BLOCK

0205 7746 MSG8, -32
 0206 0206 .+0
 0207 0322 TEXT ,CR
 0210 0305 CE
 0211 0323 CS
 0212 0305 CE
 0213 0321 CQ
 0214 0325 CU
 0215 0305 CE
 0216 0316 CN
 0217 0303 CC
 0220 0305 CE
 0221 0304 CD
 0222 0240 B
 0223 0306 CF
 0224 0311 CI
 0225 0314 CL
 0226 0305 CE
 0227 0240 B
 0230 0311 CI
 0231 0323 CS
 0232 0240 B
 0233 0322 CR
 0234 0323 CS
 -----0005

/'RESEQUENCED FILE IS RSQ001' MSG

```

0235 0321 CQ
0236 0260 B0
0237 0260 B0
0240 0261 B1
0241 0000 .

```

```

/SECTION A,
0242 4777 BEGIN, JMS CRLF
0243 1376 ASK1, TAD (MSG1) /ASK FOR PROGRAM NAME
0244 4775 JMS MSGOUT
0245 1374 TAD (BUFFER) /READ PROGRAM NAME
0246 4773 JMS INPUT
/CONVERT TO INTERNAL FILE NAME
/COMPARE CHAR COUNT TO 6
0247 1372 TAD (-6)
0250 7500 SMA
0251 5263 JMP NAMEPK /JUMP IF SIX CHARACTER NAME
0252 3333 DCA COUNT /# OF BLANKS TO PAD (NEG)
0253 1374 TAD (BUFFER) /COMPUTE ADDR FOR 1ST BLANK
0254 1771 TAD INCNT
0255 3334 DCA ADDR /SAVE IN POINTER
026 1370 PAD, TAD (240) /BLANK TO ACC
0257 3734 DCA I ADDR /MOVE TO PAD NAME
0260 2334 ISZ ADDR /BUMP POINTER
0261 2333 ISZ COUNT /BUMP COUNT
0262 5256 JMP PAD /LOOP UNLESS ALL DONE
0263 7320 NAMEPK, CLA STL /SET UP PACK LOOP (ACC=0, L=1)
0264 1367 TAD (BUFFER-1)
0265 3010 DCA Z 10 /INDEXED 10 POINTS TO ASCII BLOCK
0266 1366 TAD (IONAME-1)
0267 3011 DCA Z 11 /INDEX 11 POINTS TO IONAME BLOCK
0270 1365 TAD (-3)
0271 3333 DCA COUNT /COUNT CONTROLS LOOP 3 TIMES
0272 4306 PACK, JMS EXTRCT /EXTRACT 1ST CHARACTER
0273 7006 RTL /SHIFT LEFT 6
0274 7006 RTL
0275 7006 RTL
0276 3335 DCA TEMP
0277 4306 JMS EXTRCT /EXTRACT 2ND CHARACTER
0300 0335 AND TEMP /'OR' IN 1ST CHARACTER
0301 7040 CMA
0302 3411 DCA I Z 11 /DEPOSIT IN IONAME BLOCK
0303 2333 ISZ COUNT
0304 5272 JMP PACK /LOOP UNLESS 3 WORDS DEPOSITED
0305 5314 JMP OPENX
0306 0000 EXTRCT, 0 /SUBROUTINE FOR PICKING UP ASCII
0307 1410 TAD I Z 10 /CHARACTER TO ACC
0310 1364 TAD (40) /COMPLEMENT BIT 6
0311 0363 AND (77) /CLEAR LEFT 6 BITS
0312 040 CMA /COMPLEMENT FOR 'OR'
0313 5706 JMP I EXTRCT /RETURN
0314 4777 OPENX, JMS CRLF /**OPEN INPUT ON FILE 0**
0315 1362 TAD (IOPN) /OPEN BLOCK ADDR TO ACC
0316 6601 OPEN
0317 7450 SNA
0320 5761 JMP SECTB /JUMP IF SUCCESSFUL OPEN
0321 7200 CLA
-----00006

```

0322	1360	TAD (MSG2)	/TYPE 'COMMAND ERROR'
0323	4775	JMS MSGOUT	
0324	4777	JMS CRLF	
0325	5243	JMP ASK1	/TRY AGAIN
			/OPEN IOT BLOCK
0326	0000	IOPN, 0	/INTERNAL FILE NUMBER
0327	0000		/CURRENT USER NUMBER
0330	0000	IONAME, 0	/
0331	0000		/PACKED FILE NAME (3 WORDS)
0332	0000		/
0333	0000	COUNT, 0	
0334	0000	ADDR, 0	/BUFFER POINTER FOR PACK, ETC.
0335	0000	TEMP, 0	
0336	0000	ERRWD, 0	/SAVE AREA FOR OPEN IOT ACC
0360	0032		
0361	0400		
0362	0326		
0363	0077		
0364	0040		
0365	7775		
0366	0327		
0367	3377		
0370	0240		
0371	1502		
0372	7772		
0373	1435		
0374	3400		
0375	1530		
0376	0020		
0377	1404		
		PAGE	
		/SECTION B,	
0400	7200	SECTB, CLA	
0401	1377	TAD (LNTBL)	/SET LNPTR TO POINT TO LNTBL
0402	3306	DCA LNPTR	
0403	4776	LNGET, JMS GET	/GET A DISK WORD TO ACC
0404	3706	DCA I LNPTR	/STORE LN IN TBL
0405	1706	TAD I LNPTR	/COMPARE TO 3777
0406	1375	TAD (-3777)	
0407	7450	SNA	
040	5216	JMP TBLIN	/JUMP IF MATCH (EOF)
0411	2306	ISZ LNPTR	/BUMP POINTER
0412	4776	LNACC, JMS GET	/DISK WORD TO ACC
0413	7440	SZA	
0414	5212	JMP LNACC	/JUMP IF NOT ZERO
0415	5203	JMP LNGET	/NEXT WORD IS LN
0416	1306	TBLIN, TAD LNPTR	/CALCULATE NUMBER OF LINES
0417	7041	CIA	
0420	1377	TAD (LNTBL)	
0421	3307	DCA LINES	/LINES = # OF LN
		/SECTION C,	
0422	1374	ASK2, TAD (MSG3)	/ASK FOR STARTING LINE #
0423	4773	JMS MSGOUT	
0424	1372	TAD (BUFFER-1)	/SET X10 TO POINT TO BUFFER-1
0425	3010	DCA Z 10	
	-----0007		


```

0426 1371 TAD (BUFFER)
0427 4770 JMS INPUT /READ STARTING LN
0430 4767 JMS ASCBIN /CONVERT TO BINARY
0431 3304 DCA START /SAVE IT
0432 4766 JMS CRLF
0433 1304 TAD START /CHECK FOR +
0434 7540 SZA SMA /SKIP IF NOT .GT. 0
0435 5243 JMP ASK3 /JUMP IF OK
0436 7200 CLA
0437 1365 TAD (MSG4) /TYPE ERROR MSG
0440 4773 JMS MSGOUT
0441 4766 JMS CRLF
0442 5222 JMP ASK2 /TR AGAIN
0443 7200 ASK3, CLA
0444 1364 TAD (MSG5) /ASK FOR INCREMENT
0445 4773 JMS MSGOUT
0446 1372 TAD (BUFFER-1) /SET X10 TO POINT TO BUFFER-1
0447 3010 DCA Z 10
0450 1371 TAD (BUFFER)
0451 4770 JMS INPUT /READ INCREMENT
0452 4767 JMS ASCBIN /CONVERT TO BINARY
0453 3305 DCA INCR /SAVE IT
0454 4766 JMS CRLF
0455 1305 TAD INCR
0456 7540 SZA SMA /SKIP IF NOT .GT. 0
0457 5265 JMP LNCK /JUMP IF OK
0460 7200 CLA
0461 1365 TAD (MSG4) /TYPE ERROR MSG
0462 4773 JMS MSGOUT
0463 4766 JMS CRLF
0464 543 JMP ASK3 /TRY AGAIN
0465 7200 LNCK, CLA /**CHECK LN'S FOR RANGE
0466 1307 TAD LINES /COPY COUNT OF LINES
0467 3310 DCA LNCNT
0470 1304 TAD START /STARTING LN TO ACC
0471 1305 LNCKLP, TAD INCR /ADD INCREMENT
0472 7540 SZA SMA
0473 5301 JMP LNCK1 /JUMP IF BETWEEN 1 AND 2046
0474 7200 CLA /PUT OUT ERROR MSG
0475 1363 TAD (MSG7)
0476 4773 JMS MSGOUT
0477 4766 JMS CRLF
0500 5222 JMP ASK2 /GET NEW START AND INCR
0501 2310 LNCK1, ISZ LNCNT /BUMP COUNTER
0502 5271 JMP LNCKLP /JUMP IN LOOP UNLESS ALL DONE
0503 5762 JMP SECTD /PHASE II PLEASE.
0504 0000 START, 0 /STARTING LN
0505 0000 INCR, 0 /INCREMENT
0506 0000 LNPTR, 0 /POINTER TO LNTBL ENTRIES
0507 0000 LINES, 0 /NUMBER OF LINES IN BASIC PGM
0510 0000 LNCNT, 0 /CONTROL WORD FOR LOOP
0562 0600
0563 0163
0564 0124
0565 0077
0566 1404
-----0010

```

0567 1505
 0570 1435
 0571 3400
 0572 3377
 0573 1530
 0574 0053
 0575 4001
 0576 1600
 0577 2600

```

PAGE
/PHASE II
/SECTION D,
SECTD,  CLA           /0 TO ACC
        DCA INBLK     /'REWIND' INPUT FILE
0600  7200
0601  3777
0602  3776           DCA INBCNT
0603  3775           DCA INADDR
0604  1374           TAD (TPNAME)   /CREATE 'RSQ001'
0605  6610           CRF
0606  7440           SZA
0607  5773           JMP IOKILL     /JUMP IF UNSUCCESSFUL
0610  1372           TAD (TPOPN)   /OPEN AS INTERNAL FILE 1
0611  6601           OPEN
0612  7440           SZA
0613  5773           JMP IOKILL     /JUMP ON ERROR

/SECTION E,
MOVE,   CLA           /**MOVE NEW LINE TO BUFFER
        TAD (BUFFER)  /SET BUFPTR TO ADDRESS BUFFER
0614  7200
0615  1371
0616  3002           DCA BUFPTR
0617  4770           JMS GET       /LINE NUMBER TO ACC
0620  3402           DCA I BUFPTR   /STORE AS 1ST WORD IN BUFFER
0621  1402           TAD I BUFPTR  /CHECK FOR 3777
0622  1367           TAD (-3777)
0623  7450           SNA
0624  5766           JMP NOMORE   /JUMP ON MATCH
0625  2002           ISZ BUFPTR   /UPDATE POINTER
0626  4770           NEXTIN, JMS GET   /GET NEXT WORD IN LINE
0627  7450           SNA
0630  5246           JMP LASTIN   /JUMP ON END OF LINE
0631  3320           DCA LNHL D   /SAVE IT
0632  1320           TAD LNHL D   /SHIFT RIGHT 6
0633  7012           RTR
0634  7012           RTR
0635  7012           RTR
0636  0365           AND (77)     /CLEAR LEFT 6 BITS
0637  3402           DCA I BUFPTR  /STORE IN BUFFER
0640  2002           ISZ BUFPTR   /BUMP POINTER
0641  1320           TAD LNHL D   /WORD TO ACC
0642  0365           AND (77)     /CLEAR LEFT CHAR
0643  3402           DCA I BUFPTR  /STORE THIS 2ND CHAR
0644  2002           ISZ BUFPTR   /BUMP POINTER
0645  5226           JMP NEXTIN   /LOOP FOR NEXT WORD
0646  302  LASTIN, DCA I BUFPTR  /MOVE ZERO MARKER TO BUFFER
0647  1771           TAD BUFFER   /LINE NUMBER TO ACC
0650  4764           JMS NEWLN   /CONVERT TO NEW LINE NUMBER
0651  3771           DCA BUFFER   /REPLACE LINE NUMBER
0652  1002           TAD BUFPTR   /SAVE ADDR OF 0 IN ZWPTR
-----0011

```

0653	3763		DCA ZWPTR	
0654	1371		TAD (BUFFER)	/RESET BUFPTR TO START OF LINE
0655	3002		DCA BUFPTR	
0656	4307	CKI,	JMS SIGCHR	/GET NEXT SIGNIFICANT CHARACTER
0657	7450		SNA	/COMPARE TO ZEROES
0660	5762		JMP EOLINE	/JUMP IF 0 (END-OF-LINE)
0661	1361		TAD (-11)	/COMPARE TO 'I'
0662	7440		SZA	
0663	5273		JMP CKJ	/JUMP IF NOT 'I'
0664	4307		JMS SIGCHR	/GET NEXT CHARACTER
0665	7450		SNA	
0666	5762		JMP EOLINE	/JUMP IF 0
0667	1360		TAD (-06)	/COMPARE TO 'F' (IF STMT)
0670	7440		SZA	
0671	5762		JMP EOLINE	/JUMP IF NOT F
0672	5757		JMP SUBLN	/PROCESS 'IF' STATEMENT
0673	7200	CKJ,	CLA	
0674	1402		TAD I BUFPTR	/1ST CHARACTER TO ACC
0675	1356		TAD (-07)	/COMPARE TO 'G'
0676	7440		SZA	
0677	5762		JMP EOLINE	/JUMP IF NOT 'G'
0700	4307		JMS SIGCHR	/GET 2ND CHARACTER
0701	7450		SNA	
0702	5762		JMP EOLINE	/JUMP IF NONE THERE
0703	1355		TAD (-17)	/COMPARE TO 'O'
0704	7440		SZA	
0705	5762		JMP EOLINE	/JUMP IF NOT 'GO'
				/'IF', 'GOTO', OR 'GOSUB' STMT
0706	5757		JMP SUBLN	/GO CHANGE LINE # REFERENCE
0707	0000	SIGCHR,	0	/**RETURNS NEXT NON BLANK CHAR
0710	2002		ISZ BUFPTR	/BUMP POINTER
0711	1402		TAD I BUFPTR	/NEXT CHAR TO ACC
0712	1354		TAD (-40)	/COMPARE TO BLANK
0713	7450		SNA	
0714	5310		JMP SIGCHR+1	/JUMP IF BLANK
0715	7200		CLA	
0716	1402		TAD I BUFPTR	/CHARACTER TO ACC
0717	5707		JMP I SIGCHR	/BYE
0720	0000	LNHLD,	0	/HOLD AREA FOR LN'S
				/OPEN IOT BLOCK FOR 'RSQ001'
0721	0001	TPOPN,	1	/INTERNAL FILE NUMBER
0722	0000		0	/CURRENT USER NUMBER
				/PACKED FILE NAME (3 WORDS)
0723	6263	TPNAME,	TEXT ,23	
0724	6120		1P	
0725	2021		PQ	
0726	0000		.	
0754	7740			
0755	7761			
0756	7771			
0757	1000			
0760	7772			
0761	7767			
0762	1112			
0763	1150			
-----0012				

0764 2000
 0765 0077
 0766 1200
 0767 4001
 0770 1600
 0771 3400
 0772 0721
 0773 1641
 0774 0723
 0775 1654
 0776 1652
 0777 1650

PAGE

/THE FOLLOWING CODE FINDS AND SUBSTITUTE A LN
 /REFERENCE WITHIN A LINE.

1000	1350	SUBLN,	TAD ZWPTR	/SET BUFPTR TO 0 ADDR
1001	3002		DCA BUFPTR	
				/MOVE POINTER BACK TO 1ST NON-
				/ZERO, NON BLANK CHAR
				/BUFPTR .EQ, BUFPTR-1
1002	7240	BACKUP,	STA	
1003	1002		TAD BUFPTR	
100	3002		DCA BUFPTR	
1005	1402		TAD I BUFPTR	/CHARACTER TO ACC
1006	1377		TAD (-40)	/COMPARE TO BLANK
1007	7450		SNA	
1010	5202		JMP BACKUP	/JUMP IF BLANK
1011	7200		CLA	
1012	1402		TAD I BUFPTR	/CHARACTER TO ACC
1013	1376		TAD (-60)	/COMPARE TO ZERO
1014	7510		SPA	
1015	5221		JMP MARKPT	/JUMP IF LESS THAN 0
1016	1375		TAD (-12)	
1017	7510		SPA	
1020	5202		JMP BACKUP	/JUMP IF .LE, 9 (NUMERIC)
1021	7200	MARKPT,	CLA	/CHAR IS NON-NUMERIC
1022	1002		TAD BUFPTR	/SAVE BUFPTR IN LSTSIG
1023	3351		DCA LSTSIG	
1024	3352		DCA BINLN	/CLEAR BINLN
1025	4774	CVTBIN,	JMS SIGCHR	/FIND NEXT (RIGHT) SIGNIFICANT
1026	7450		SNA	
1027	5243		JMP CVTEXT	/JUMP IF ZERO WORD (END OF LN)
1030	7300		CLA CLL	/MULTIPLY BINLN BY 10
1031	1352		TAD BINLN	
1032	7006		RTL	
1033	1352		TAD BINLN	
1034	7004		RAL	
1035	3352		DCA BINLN	
1036	1402		TAD I BUFPTR	/NEW CHAR TO ACC
1037	0373		AND (17)	/CONVERT TO BINARY
1040	1352		TAD BINLN	/ADD TO BINLN
1041	3352		DCA BINLN	
1042	5225		JMP CVTBIN	/LOOK FOR NEXT CHAR
1043	1352	CVTEXT,	TAD BINLN	/BINARY LINE NUMBER TO ACC
1044	4772		JMS NEWLN	/CONVERT TO NEW LINE NUMB
1045	3352		DCA BINLN	
1046	1351		TAD LSTSIG	/RESTORE BUFPTR

-----0013

1047	3002		DCA	BUFPTR	
1050	2002		ISZ	BUFPTR	/MOVE A BLANK TO THE BUFFER
1051	1371		TAD	(40)	
1052	3402		DCA	I BUFPTR	
1053	2002		ISZ	BUFPTR	/SET BUFPTR FOR LN CONVERSION
1054	7100		CLL		/CONVERT BINLN TO DECIMAL AND MV
1055	1370		TAD	(CVTTBL)	/CLEAR LINK (LEADING 0 SWITCH)
1056	3355		DCA	CVTX	/SET POINTER TO CVT CONSTANTS
1057	1367		TAD	(-4)	/SET LOOP CNTRL
1060	3354		DCA	CVTCNT	
1061	3353	CVTCLR,	DCA	BASE10	/CLEAR BASE 10 ACCUMULATOR
1062	1352	CVTSUB,	TAD	BINLN	/BINARY # TO ACC
1063	1755		TAD	I CVTX	/SUBTRACT CONSTANT
1064	7510		SPA		
1065	5270		JMP	CVTNEG	/JUMP IF # LESS THAN CONSTANT
1066	2353		ISZ	BASE10	/UMP DECIMAL COUNTER
1067	5263		JMP	CVTSUB+1	
1070	7041	CVTNEG,	CIA		/RESTORE ACC TO LAST + VALUE
1071	1755		TAD	I CVTX	
1072	7041		CIA		
1073	3352		DCA	BINLN	/SAVE IN BINLN
1074	1353		TAD	BASE10	/COMPARE FOR LEADING ZERO
1075	7440		SZA		
1076	5301		JMP	CVTOUT	/JUMP IF NOT ZERO
1077	7420		SNL		
1100	5305		JMP	CVNXT	/JUMP IF LEADING (L=0)
1101	1366	CVTOUT,	TAD	(60)	/CHANGE TO INTERNAL CODE
1102	3402		DCA	I BUFPTR	/STORE IN BUFFER
1103	7120		STL		/SET LINK TO MARK 0 NOT LEAD
1104	2002		ISZ	BUFPTR	/BUMP POINTER
1105	2355	CVTNXT,	ISZ	CVTX	/INCREMENT CONSTANT POINTER
1106	2354		ISZ	CVTCNT	/CHECK COUNT OF LOOP
1107	5261		JMP	CVTCLR	/DO NEXT DIGIT UNLESS FINISHED
					/CONVERSION COMPLETE
1110	7200		CLA		
1111	3402		DCA	I BUFPTR	/STORE 0 AS END OF LINE
1112	7200	EOLINE,	CLA		/END LINE--WRITE TO OUTPUT
1113	1365		TAD	(BUFFER)	/SET BUFPTR TO BUFFER
1114	3002		DCA	BUFPTR	
1115	1402		TAD	I BUFPTR	/SEND OFF LN
1116	4764		JMS	PUT	
1117	2002		ISZ	BUFPTR	/BUFFER POINTER TO 1ST TEXT
1120	1402	OUTLN,	TAD	I BUFPTR	/1ST WORD TO ACC
1121	7450		SNA		
1122	5346		JMP	OUTEXT	/JUMP IF ZERO WORD
1123	7006		RTL		
1124	7006		RTL		/SHIFT LEFT 6
1125	7006		RTL		
1126	0363		AND	(7700)	/CLEAR 6 RIGHT BITS
1127	7040		CMA		/COMPLEMENT
1130	3356		DCA	OUTHLD	/SAVE IT
1131	2002		ISZ	BUFPTR	/SET 2ND WORD
1132	1402		TAD	I BUFPTR	
1133	7440		SZA		

-----0014

1134	5337		JMP OUT20K	/JUMP IF NOT ZERO
1135	1371		TAD (40)	/INSERT BLANK IN STRING
1136	7410		SKP	/SKIP BUMP INSTRUCTION
1137	200	OUT20K,	ISZ BUFPTR	
1140	0362		AND (77)	/CLEAR 6 LEFT BITS
1141	7040		CMA	/COMPLEMENT
1142	0356		AND OUTHLD	/OR WITH 1ST WORD
1143	7040		CMA	
1144	4764		JMS PUT	/SEND TO OUTPUT FILE
1145	5320		JMP OUTLN	/NEXT PAIR
1146	4764	OUTEXT,	JMS PUT	/SEND ZERO WORD
1147	5761		JMP MOVE	/GO DO NEXT LINE
1150	0000	ZWPTR,	0	/POINTER TO ZERO MARKER
1151	0000	LSTSIG,	0	/POINTER TO LAST SIZ CHAR IN LINE
1152	0000	BINLN,	0	/BINARY LN REFERENCE
1153	0000	BASE10,	0	/DECIMAL VALUE DURING CVT
1154	0000	CVTCNT,	0	/CVT LOOP COUNTER
1155	0000	CVTX,	0	/CVT LOOP COUNTER
1156	0000	OUTHLD,	0	/TEMP AREA DURING OUTPUT PACKING
1161	0614			
1162	0077			
1163	7700			
1164	1656			
1165	340			
1166	0060			
1167	7774			
1170	1260			
1171	0040			
1172	2000			
1173	0017			
1174	0707			
1175	7766			
1176	7720			
1177	7740			

PAGE

1200	1402	NOMORE,	TAD I BUFPTR	/ALL INPUT LINES PROCESSED
1201	4777		JMS PUT	/LAST LN IS 3777
1202	4777		JMS PUT	/SEND IT TO OUTPUT
				/SEND ZERO TO OUTPUT
		/SECTION F,		
1203	1776		TAD PTCNT	/COMPARE WORD COUNT IN OUTBUF TO
1204	1375		TAD (200)	/-128 (BUFFER EMPTY)
1205	7450		SNA	
1206	5215		JMP EMPTY	/JUMP IF EMPTY
1207	7200		CLA	/DUMP OUTBUF ON DISK
1210	1374		TAD (OUTBUF+177)	
1211	3773		DCA PTPTR	
1212	7240		STA	
1213	3776		DCA PTCNT	
1214	4777		JMS PUT	
1215	1372	EMPTY,	TAD (ININFO)	/DETERMINE STATUS OF THE
1216	6613		FINF	/ INPUT FILE
1217	7200		CLA	/COPY PROTECT CODE TO RSQ001
1220	1256		TAD ININFO+5	
1221	0371		AND (7637)	
1222	1370		TAD (0040)	

-----0015

```

1223 6604 PROT
1224 7200 CLA /MOVE # SGMTS TO REDUCE BLOCK
1225 3767 DCA EXTND
1226 1257 TAD ININFO+6
1227 3766 DCA EXTND+1
1230 1367 TAD (EXTND) /ERASE INPUT FILE
1231 6612 RED
1232 7440 SZA
1233 5244 JMP NOKILL /JUMP IF NOT POSSIBLE TO ERASE
1234 7201 CLA IAC /RENAME RSQ001 AS INPUT
1235 3765 DCA IONAME-1
1236 1365 TAD (IONAME-1)
1237 6600 REN
120 7200 CLOSE, CLA /CLOSE INPUT AND OUTPUT FILES
1241 1364 TAD (6000)
1242 6602 CLOS
1243 7402 HLT /END OF PROCESSING
1244 7200 NOKILL, CLA /PUT OUT RSQ001 RESULT MSG
1245 1363 TAD (MSG8)
1246 4762 JMS MSGOUT
1247 4761 JMS CRLF
1250 5240 JMP CLOSE
1251 0000 ININFO, 0 /7 WORD FILE INFO BLOCK
1252 0000 0 /FILE 0 ON CURRENT ACCOUNT
1253 0000 0 /3 WORD FILE NAME
1254 0000 0
1255 0000 0
1256 0000 0 /PROTECT CODE
1257 0000 0 /NUMBER OF SEGMENTS IN THE FILE
DECIMAL /CONVERSION CONSTANTS
1260 6030 CVTTB, -1000
1261 7634 -100
1262 7766 -10
1263 7777 -1
OCTAL

1361 1404
1362 1530
1363 0205
1364 6000
1365 0327
1366 1733
1367 1732
1370 0040
1371 7637
1372 1251
1373 1727
1374 2577
1375 020
1376 1726
1377 1656

```

PAGE

```

/*****
/*IOCS SYSTEM AND UTILITY ROUTINES*
/*****

```

-----0016

```

/PRINT THE ASCII CHARACTER IN THE ACC AND CLEAR IT
1400 0000 PRINT, 0
1401 6046     TLS
1402 7200     CLA
1403 5600     JMP I PRINT

```

```

/CRLF GENERATES A CARRIAGE RETURN AND A LINE FEED
1404 0000 CRLF, 0
1405 7200     CLA
1406 1377     TAD (215)       /CR TO ACC
1407 4200     JMS PRINT
1410 1376     TAD (212)       /LF TO ACC
1411 4200     JMS PRINT
1412 5604     JMP I CRLF     /BYE

```

```

/OPRINT PRINTS THE CONTENTS OF ACC (IN OCTAL)
1413 0000 OPRINT, 0
1414 7004     RAL
1415 3234     DCA OPRHLD     /SAVE ACC CONTENTS
1416 1375     TAD (-4)      /SET UP LOOP
1417 3233     DCA OPR4
1420 1234     OPRLP, TAD OPRHLD /PICK UP CURRENT WORD
1421 7006     RTL          /ROTATE LEFT 3
1422 7004     RAL
1423 3234     DCA OPRHLD     /SAVE IT
1424 1234     TAD OPRHLD
1425 0374     AND (7)       /CLEAR ALL BUT LOW ORDER 3 BITS
1426 1373     TAD (260)     /ADD ASCII '0'
1427 4200     JMS PRINT     /PRINT IT
1430 2233     ISZ OPR4      /BUMP COUNTER AND SKIP IF DONE
1431 5220     JMP OPRLP
1432 5613     JMP I OPRINT  /BYE
1433 0000     OPR4, 0       /-4 WILL BE HERE
1434 0000     OPRHLD, 0

```

/THIS UBROUTINE ACCEPTS A STRING OF CHARACTERS FROM THE /TTY AND PLACES THEM IN CORE. BACKSPACING MAY BE DONE /WITH RUBOUT AND RETURN ENDS INPUT, ON CALL, ACC SHOULD /POINT TO CORE AREA. ON RETURN, ACC=COUNT OF CORRECT /CHARACTERS TRANSMITTED, THE RETURN CHARACTER IS NOT /MOVED TO CALLER'S AREA, BUT OTHER CHARACTERS IN IT /MAY BE DISTURBED BY THE BACKSPACE PROCESS.

```

1435 0000 INPUT, 0
1436 3303     DCA INPTR     /BUFFER ADDR TO INPTR
1437 3302     DCA INCNT     CLEAR COUNT OF CHARACTERS
1440 1372     TAD (0204)    /MASK FOR BREAK ON RETURN OR RUB
1441 6400     KSB
1442 6036     INREAD, KRB   /READ NEXT CHARACTER
1443 3304     DCA INHOLD    /SAVE IN INHOLD
1444 1304     TAD INHOLD    /COMPARE FOR RETURN
1445 7041     CIA
1446 1377     TAD (215)
1447 7440     SZA
1450 5253     JMP INX1      /JUMP IF NOT RETURN
1451 1302     TAD INCNT     /COUNT TO ACC
1452 5635     JMP I INPUT   /BYE

```

-----0017


```

1453 7200 INX1, CLA /COMPARE FOR RUBOUT
1454 1304 TAD INHOLD
1455 7041 CIA
1456 1371 TAD (377)
1457 7440 SZA
1460 5274 JMP INX2 /JUMP IF NOT RUBOUT
1461 1370 TAD (334) /ECHO 0
1462 4200 JMS PRINT
1463 1302 TAD INCNT /CHARACTER COUNT TO ACC
1464 7450 SNA /IGNORE RUBOUT IF ZERO COUNT
1465 5242 JMP INREAD
1466 1367 TAD (-1) /SUBTRACT 1 FROM WDCNT AND INPTR
1467 3302 DCA INCNT
1470 1303 TAD INPTR
1471 1367 TAD (-1)
1472 3303 DCA INPTR
1473 5242 JMP INREAD /NEXT CHARACTER
1474 7200 INX2, CLA /STORE CHARACTER IN CALLER'S AREA
1475 1304 TAD INHOLD
1476 3703 DCA I INPTR
1477 2303 ISZ INPTR /BUMP INPTR
1500 2302 ISZ INCNT /BUMP CHARACTER COUNT
1501 5242 JMP INREAD /NEXT CHARACTER
1502 0000 INCNT, 0 /CHARACTER COUNT
1503 0000 INPTR, 0 /POINTER TO NEXT STORAGE ADDR
1504 0000 INHOLD, 0 /HOLD ARE FOR CURRENT CHAR

```

/THIS SUBROUTINE CONVERTS AN ASCII STRING TO BINARY,
 /AUTO INDEX 10 SHOULD = STRING-1, ACC SHOULD = THE
 /LENGTH OF THE STRING. ON RETURN, ACC = BINARY EQUIVALENT.

```

1505 0000 ASCBIN, 0
1506 7041 CIA
1507 3327 DCA ASCCNT /STORE COUNT
1510 3326 DCA ASCWRK /CLEAR WORKAREA
1511 1326 ASCLP, TAD ASCWRK /WORKAREA TO ACC
1512 7106 CLL RTL /MULTIPLY BY 10
1513 1326 TAD ASCWRK
1514 7004 RAL
1515 3326 DCA ASCWRK
1516 1410 TAD I Z 10 /NEXT DIGIT TO ACC
1517 0366 AND (17) /CONVERT TO BINARY
1520 1326 TAD ASCWRK /ADD TO WORKAREA
1521 3326 DCA ASCWRK
1522 2327 ISZ ASCCNT /BUMP COUNTER
1523 5311 JMP ASCLP /LOOP UNLESS ALL PROCESSED
1524 1326 TAD ASCWRK /RESULTS TO ACC
1525 5705 JMP I ASCBIN /BYE
1526 0000 ASCWRK, 0
1527 0000 ASCCNT, 0

```

/THIS SUBROUTINE TYPES MESSAGES. IT DOES NOT ALTER THE
 /CALLING MESSAGE BLOCK. ON CALL, ACC=ADDR OF TWO WORD
 /BLOCK CONTAINING
 / -CHAR COUNT
 / ADDR-1 OF 1ST WORD OF TEXT

-----0020

```

/ACC=0 ON RETURN
1530 0000 MSGOUT, 0
1531 3343 DCA MSGWRK /STORE BLOCK ADDR
1532 1743 TAD I MSGWRK /COPY CHARACTER COUNT
1533 3344 DCA MSGCNT
1534 2343 ISZ MSGWRK /COPY ADDR
1535 1743 TAD I MSGWRK
1536 3345 DCA MSGADR
1537 1365 TAD (MSGCNT)
1540 6040 SAS /SEND THE MSG
1541 5337 JMP , -2
1542 5730 JMP I MSGOUT /BYE
1543 0000 MSGWRK, 0
1544 0000 MSGCNT, 0
1545 0000 MSGADR, 0
1565 1544
1566 0017
1567 7777
1570 0334
1571 0377
1572 0204
1573 0260
1574 0007
1575 7774
1576 0212
1577 0215

```

```

PAGE
/GET OBTAINS THE NEXT WORD FROM THE INPUT FILE AND
/HANDLES ALL INTERNAL BUFFERING, THE WORD IS RETURN IN
/THE ACC.

```

```

1600 0000 GET, 0
1601 7300 CLA CLL
1602 1252 TAD INBCNT
1603 7450 SNA
1604 5213 JMP GETIO /JMP IF BUFFER EMPTY
1605 2253 GETGO, ISZ INBPTR /BUMP POINTER
1606 7200 CLA /OUTPUT WORD TO ACC
1607 1653 TAD I INBPTR
161 2252 ISZ INBCNT /BUMP WORD COUNT
1611 7000 NOP
1612 5600 JMP I GET /BYE
1613 4233 GETIO, JMS GETINT /INITIALIZE POINTERS
1614 1377 TAD (INBLK)
1615 6603 RFILE /READ A PAGE
1616 7440 SZA
1617 5241 JMP IOKILL /JUMP ON IO ERROR
1620 1255 TAD INERR
1621 7440 SZA
1622 5241 JMP IOKILL /JUMP ON IO ERROR
1623 4233 JMS GETINT /RESET POINTERS
1624 1254 TAD INADDR /ADD 128 TO DISK ADDR
1625 1376 TAD (200)
1626 3254 DCA INADDR
1627 7004 RAL /LINK TO ACC
1630 1250 TAD INBLK /ADD TO HIGH ORDER DISK ADDR
1631 3250 DCA INBLK

```

```

-----0021

```

```

1632 5205      JMP GETGO
1633 0000  GETINT, 0      /INITIALIZE SUBROUTINE
1634 1375      TAD (-200)      /WDCNT = -128
1635 3252      DCA INBCNT
1636 1374      TAD (INBUF-1)  /SET BUFFER ADDR
1637 3253      DCA INBPTR
1640 5633      JMP I GETINT
1641 3255  IOKILL, DCA INERR      /IO ERROR--PUT MSG AND HALT
1642 1373      TAD (MSG6)
1643 4772      JMS MSGOUT
1644 1255      TAD INERR      /PRINT OCTAL ERROR CODE
1645 4771      JMS OPRINT
1646 4770      JMS CRLF
1647 7402      HLT

```

```

1650 0000  INBLK, 0      /RFILE IOT BLOCK
1651 0000      0      /HIGH ORDER DISK ADDR
1652 0000      0      /FILE # (0)
1653 0000  INBCNT, 0    /WORD COUNT
1654 0000  INBPTR, 0    /BUFFER ADDR
1655 0000  INADDR, 0    /HIGH ORDER ADDR
1655 0000  INERR, 0    /ERROR INDICATOR

```

/PUT TRANSFERS WORDS TO THE OUTPUT FILE AND HANDLES ALL /INTERNAL BUFFERING. THE WORD IN THE ACC IS SENT TO THE /DISK AND THE ACC IS CLEARED.

```

1656 0000  PUT, 0
1657 2327      ISZ PTPTR      /BUMP POINTER FOR DEPOSIT
1660 3727      DCA I PTPTR    /DEPOSIT WORD IN DISK BUFFER
1661 2326      ISZ PTCNT      /BUMP COUNT, SKIP ON BUFFER FULL
1662 5656      JMP I PUT      /BYE
1663 4304      JMS PUTINT    /INITIALIZE IOT BLOCK
1664 1367  PUTGO, TAD (PTBLK)  /WRITE THE BUFFER PAGE
1665 6605      WFILE
1666 7440      SZA
1667 5241      JMP IOKILL    /JUMP ON IO ERROR
1670 1331      TAD PTERR
1671 7440      SZA
1672 5312      JMP PUTERR    /JUMP TO FIND FAIL CAUSE
1673 4304      JMS PUTINT    /INITIALIZE POINTERS AGAIN
1674 7100      CLL
1675 1330      TAD PTADDR    /ADD 128 TO DISK ADDR
1676 1376      TAD (200)
1677 3330      DCA PTADDR
1700 7004      RAL      /LINK TO ACC
1701 1324      TAD PTBLK    /ADD TO HIGH ORDER DISK ADDR
1702 3324      DCA PTBLK
1703 5656      JMP I PUT      /BYE
1704 0000  PUTINT, 0    /INITIALIZE SUBROUTINE
1705 1375      TAD (-200)    /WORD COUNT = -128
1706 3326      DCA PTCNT
1707 1366      TAD (OUTBUF-1) /SET BUFFER ADDR
1710 3327      DCA PTPTR
1711 5704      JMP I PUTINT    /END SUBR
1712 1365  PUTERR, TAD (-2)  /IO ERROR -- CK FILE SHORT
1713 7450      SNA

```

-----0022

```

1714 5317 JMP PUTEXT /JUMP IF FILE TOO SHORT
1715 1364 TAD (2) /RESTORE ERROR CODE
116 5241 JMP IOKILL /GO PUT MSG & ABORT
1717 1363 PUTEXT, TAD (EXTND) /EXTEND THE FILE BY 1 SGMT
1720 6611 EXT
1721 7440 SZA
1722 5241 JMP IOKILL /JUMP ON IO ERROR
1723 5264 JMP PUTGO

```

```

1724 0000 PTBLK, 0 /WFILE IOT BLOCK
1725 0001 1 /HIGH ORDER DISK ADDR
1726 7600 PTCNT, -200 /FILE #
1727 2377 PTPTR, OUTBUF-1 /WORD COUNT
1730 0000 PTADDR, 0 /BUFFER ADDR
1731 0000 PTERR, 0 /LOW ORDER DISK ADDR
1732 0001 EXTND, 1 /ERROR INDICATOR
1733 0001 1 /2 WORD EXTEND BLOCK

```

```

1763 1732
1764 0002
1765 7776
1766 2377
1767 1724
1770 1404
1771 1413
1772 1530
1773 0141
1774 2177
1775 7600
1776 0200
1777 1650

```

PAGE

/NEWLN COMPUTES THE NEW LINE NUMBER GIVEN THE OLD ONE.
/ON CALL, ACC=OLD#; ON RETURN, ACC=NEW#. IF THE OLD
/NUMBER IS UNDEFINED, IT IS NOT CHANGED.

```

2000 0000 NEWLN, 0
2001 7041 CIA /SAVE -OLD#
2002 3234 DCA OLD
2003 1777 TAD START /SET NEW = START
2004 3233 DCA NEW
2005 1776 TAD LINES /COPY # OF LINES
2006 3235 DCA NEWCNT
2007 1375 TAD (LNTBL-1) /SET X10 TO POINT IN LNTBL
2010 3010 DCA Z 10
2011 1234 NEWLP, TAD OLD /COMPARE OLD AGAINST TBL ENTRY
2012 1410 TAD I Z 10
2013 7440 SZA
2014 5217 JMP NEWNP /JUMP IF NO MATCH
2015 1233 TAD NEW /NEW TO ACC
2016 5600 JMP I NEWLN /BYE
2017 7500 NEWNP, SMA
2020 5227 JMP NEWNOT /JUMP IF OLD<TBL (NOT IN TBL)
2021 7200 CLA /OLD>TBL (KEEP LOOKING)
2022 1233 TAD NEW /ADD INCR TO NEW
2023 1774 TAD INCR
2024 3233 DCA NEW

```

-----0023

2025	2235		ISZ NEWCNT	/BUMP COUNTER
2026	5211		JMP NEWLP	/CHECK AGAINST NEXT UNLESS END TBL
2027	7200	NEWNOT, CLA		/NOT IN TBL--RETURN OLD
2030	1234		TAD OLD	
2031	7041		CIA	
2032	5600		JMP I NEWLN	/BYE
2033	0000	NEW, 0		/NEW # DEVELOPED HERE
2034	0000	OLD, 0		/-OLD # STORED HERE
2035	0000	NEWCNT, 0		/SEARCH LOOP CONTROL WORD
2174	0505			
2175	2577			
2176	0507			
2177	0504			

			PAGE	
			/THIS PAGE IS THE INPUT DISK BUFFER	
2200	0000		INBUF, 0	
			PAGE	
			/THIS PAGE IS THE OUTPUT DISK BUFFER	
2400	0000		OUTBUF, 0	
			PAGE	
			/THIS PAGE (AND THE NEXT 3) WILL HOLD THE LNTBL	
2600	0000		LNTBL, 0	
			PAGE	
			PAGE	
			PAGE	
			/THIS PAGE IS THE TTY AND UTILITY BUFFER	
3400	0000		BUFFER, 0	
			PAGE	
			/THAT IS ALL,	

