

IDENTIFICATION

READ DATA CARDS - MURCD2 - MURCD3
704 FORTRAN SUBROUTINES, coded in FAP

Terry W. Edwards et al. - April 23, 1965

Midwestern Universities Research Association, Stoughton, Wisconsin

PURPOSE

To read variable format data cards consisting of successive pairs of array addresses and numeric values and to store the numbers in the corresponding array address in either floating point or fixed point form. Data cards may be read either on-line via the card reader or off-line from tape.

METHOD

MURCD2-MURCD3 reads data cards until encountering an END-DATA card, a blank card, or an end-of-file condition. Program control is returned to the calling program if a non-zero number of data cards have been encountered, or under certain conditions (see below) if successive END-DATA cards are encountered.

Normally two successive END-DATA cards result in program termination. If reading is off-line, control is restored to the operating system via subroutine EXIT. If reading is on-line, a program stop occurs, after which, if more data is available, the START button causes a resumption of data reading.

By means of a preliminary entry, however, it is possible to force control back to the calling program whenever a second (or higher) successive END-DATA card is encountered. An associated counter is also incremental thus permitting program control to be subsequently varied.

A blank card or an end-of-file condition is usually ignored. An exception arises whenever such a condition is encountered immediately after an END-DATA card in which case the condition is treated as if another END-DATA card has been encountered.

If an illegally punched data card is encountered, MURCD2-MURCD3 ignores the offending address and/or numeric values together with the remainder of the card, prints on-line or writes off-line the offending card (followed by a program stop in the on-line case to permit rectification) and proceeds to read the next data card.

USAGE

An optional preliminary routine to force return of program control to the calling program whenever an END-DATA card is encountered may be entered via the FORTRAN call statement:

DISCLAIMER

Although this program has been carefully tested by its contributor, no guarantee is made of its correct functioning under all conditions, and no responsibility is taken by him in case of possible failure.

CALL ENDSET (N)

where N is a FORTRAN integer counter set thereafter by MURCD2-MURCD3 to the number of successive END-DATA cards encountered.

The two principal routines MURCD2-MURCD3 differ only in the manner in which they treat apparent integers (numbers without decimal points or exponents). MURCD2 always converts such data numbers to floating point form, whereas MURCD3 does so only if a decimal point and/or an exponent is found. If no decimal point or exponent is found, MURCD3 yields FORTRAN type integer conversion.

MURCD2-MURCD3 also permits numbers to be entered into multi-dimensional arrays, via the FORTRAN call statements:

```
CALL MURCD2(A)
CALL MURCD2(B, I1)
CALL MURCD2(C, I1, I2)
```

```
CALL MURCD3(A)           or   CALL MURCD3(K)
CALL MURCD3(B, I1)       or   CALL MURCD3(L, I1)
CALL MURCD3(C, I1, I2)   or   CALL MURCD3(M, I1, I2)
```

where A, B, C, K, L, and M are arrays of the forms:

```
A(I1)   B(I1, I2)   C(I1, I2, I3)
K(I1)   L(I1, I2)   M(I1, I2, I3)
```

In order to compute the proper address in which to store numbers in a multi-dimensional array MURCD2-MURCD3 requires the first dimension(s) of the multidimensional array as illustrated above.

DATA CARD FORMAT

At least one loading address and one numeric value must be punched per card. If more than one value is desired on a card, each value must be preceded by its own loading address. Sequential loading addresses are no exception to this rule. The appearance of two successive blank columns terminates interpretation of a data card. Each loading address must be separated from its intended value by a single blank column. Successive pairs of addresses and values must be separated from each other by a comma or a comma and a blank column. A maximum of 72 columns may be used on any one data card.

Each loading address may consist of up to three integer type array index values, up to four digits each in length, separated by commas and with no intervening blank columns. Second and third indices are taken as unity if not punched.

Each numeric value consists of an initial optional algebraic sign, a numeric factor with or without a decimal point, and an optional exponent. These subfields must not contain any blank columns nor be separated in any way from one another. If no algebraic sign is present, the number is assumed to be positive. If no decimal point is present, it is assumed to follow the rightmost numerical digit. The exponent is assumed to be zero if not present; if present, it must contain either an E or be signed, and may not be more than two digits in length.

Leading zeros of a numeric factor less than unity will be ignored, except for effective exponent, and only the first ten significant digits will be converted by MURCD2-MURCD3.

Examples of acceptable array addresses and numeric values are:

3 107 2,17 6.283185307+00 1,3,5 -0.0016E-6

Note that the first example will be converted to floating point form by MURCD2, whereas it will be left as an integer type number by MURCD3.

CREDITS

The origin of MURCD2-MURCD3 is that version of MURCD2, G. A. Westlund - December 9, 1959 (MURA-536). The author is indebted to Dr. R. S. Christian, Mr. M. Storm, Mr. J. Miller, and Mr. H. Meier, for their contributions and/or suggestions.

CODING INFORMATION

MURCD2-MURCD3 requires 337 words of program space and up to 40 words of temporary storage.

The on-line version of MURCD2-MURCD3 is available in relocatable binary form from the system tapes at this installation. The off-line version of MURCD2-MURCD3 is automatically supplied to a MOPS (MURA OPERATING SYSTEM, MURA-695) job by the appearance of a * DATA card and thereby replaces the on-line version of MURCD2-MURCD3 in such a job.

AGENDUM SHEETS

A standard MURA/FORTRAN agendum sheet has been designed and is available at this installation for FORTRAN users of MURCD2-MURCD3. See attachment.

MURA/FORTRAN AGENDUM

PROGRAM NO. _____ SUBMITTER _____

Instructions:

1. Enter program number and submitter's name above.
2. Enter address and value below for parameters desired.
3. Enter sense switch settings:

Notes:

1. Decimal points may be omitted only if understood to follow the rightmost digit.
2. Addresses may not contain more than 4 digits.
3. Only 10 digits of the factor are interpreted. Leading zeros are ignored.
4. Exponents may not contain more than 2 digits.
5. Exponents may be omitted if zero. If not, they must be signed or contain an E.

[illegible]

MURCD2

MURCD3 - MURA FORTRAN ONLINE CARD READ SUBROUTINE - EDWARDS

CONTRIBUTIONS BY WESTLUND, CHRISTIAN, STORM, EDWARDS,
MILLER, AND MEIER.

LINKAGES FROM FORTRAN PROGRAMS

```
CALL MURCD2 (A)
CALL MURCD2 (B,I1)
CALL MURCD2 (C,I1,I2)
CALL MURCD3 (A)           (OR)   MURCD3 (K)
CALL MURCD3 (B,I1)       (OR)   MURCD3 (L,I1)
CALL MURCD3 (C,I1,I2)    (OR)   MURCD3 (M,I1,I2)
```

WHERE A, B, C, K, L, AND M ARE ARRAYS OF THE FORMS
A(I1), B(I1,I2), C(I1,I2,I3),
K(I1), L(I1,I2), M(I1,I2,I3).

THE DATA CARD FORMAT IS UNCHANGED FROM THAT OF MURCD2
(DEC. 9, 1959) EXCEPT THAT FOR TWO AND THREE DIMENSIONAL
ARRAYS, THE ARRAY ADDRESS INDICES MUST BE PUNCHED WITH
COMMAS (NO BLANKS) BETWEEN INDICES, I.E.

```
FOR 1 DIMENSION  PUNCH  I1 A(I1)
FOR 2 DIMENSIONS PUNCH  I1,I2 B(I1,I2)
FOR 3 DIMENSIONS PUNCH  I1,I2,I3 C(I1,I2,I3).
```

2ND AND 3RD ARRAY INDICES ARE TAKEN AS 1 IF NOT PUNCHED.

MURCD2 YIELDS NORMALIZED FLOATING POINT CONVERSION.

MURCD3 YIELDS (FORTRAN) INTEGER CONVERSION IF THE PARTICULAR NUMBER BEING CONVERTED HAS NO EXPONENT AND DOES NOT CONTAIN A DECIMAL (12-8-4) PUNCH. IF OTHERWISE, THE CONVERSION IS TO NORMALIZED FLOATING POINT FORM.

THE OPERATION OF MURCD2, MURCD3 UPON ENCOUNTERING END-DATA CARDS MAY BE CONTROLLED BY A (OR NO) PRIOR ENTRY INTO ENDSET, VIA THE FORTRAN STATEMENT

```
CALL ENDSET (N)           WITH N = 0 .
```

UPON FINDING A FIRST SUCCESSIVE END-DATA CARD, CONTROL IS RETURNED TO THE CALLING PROGRAM IN EITHER CASE, BUT WITH N SET TO UNITY IF THERE HAS BEEN A PRIOR ENTRY INTO ENDSET.

UPON FINDING A SECOND (OR HIGHER) SUCCESSIVE END-DATA CARD, A STOP (HPR 6060606) OCCURS IF NO PRIOR ENTRY INTO ENDSET, WHEREAS IF THERE HAS BEEN A PRIOR ENTRY INTO ENDSET, CONTROL IS AGAIN RETURNED TO THE CALLING PROGRAM WITH N INCREMENTED BY UNITY FOR EACH SUCCESSIVE END-DATA CARD ENCOUNTERED.

IN THE ABOVE CASE OF NO PRIOR ENTRY INTO ENDSET AND A SECOND OR HIGHER SUCCESSIVE END-DATA CARD OR UPON FINDING TWO SUCCESSIVE BLANK CARDS, OR AN EMPTY CARD READER, A STOP (HPR 6060606) OCCURS. IN ORDER TO CONTINUE, LOAD MORE DATA AND PUSH START. (A DATA CARD WITH AN ADDRESS OF 0 WILL RETURN CONTROL TO THE CALLING PROGRAM WITH NO FURTHER INPUT OF DATA.)

UPON FINDING AN INCORRECTLY PUNCHED DATA CARD, IT IS PRINTED ON LINE AND A STOP (HPR 7070707) OCCURS. IN ORDER TO CONTINUE, RECTIFY CARD AND PUSH START.

***** SUPER

00000	ENTRY	ENDSET
00010	ENTRY	MURCD2
00012	ENTRY	MURCD3

00000	0500	00	4	00001	ENDSET	CLA	1,4	END-DATA SETUP ENTRY POINT
00001	0621	00	0	00114		STA	NSIG1	
00002	0621	00	0	00116		STA	NSIG2	STORE END-COUNTER ADDRESSES
00003	0621	00	0	00125		STA	NSIG3	
00004	0560	00	0	00500		LDQ	NOP	
00005	-0620	00	0	00111		SLQ	NSWT1	SET END-COUNTER SWITCHES
00006	-0620	00	0	00124		SLQ	NSWT2	
00007	0020	00	4	00002		TRA	2,4	

00010	0560	00	0	00011	MURCD2	LDQ	**+1	MURCD2 ENTRY POINT
00011	0020	00	0	00013		TRA	**+2	
00012	0560	00	0	00101	MURCD3	LDQ	TMI	MURCD3 ENTRY POINT
00013	-0620	00	0	00317		SLQ	FXOFL	
00014	-0634	00	1	00072		SXD	IX1,1	SAVE INDEX REGISTERS
00015	-0634	00	2	00075		SXD	IX2,2	
00016	-0634	00	4	00103		SXD	IX4,4	
00017	-0500	00	4	00001	SEQUN	CAL	1,4	PLANT LOCATION OF ARRAY
00020	0400	00	0	00475		ADD	ONE	
00021	0621	00	0	00430		STA	STORE	
00022	0560	00	4	00001	A1	LDQ	1,4	RETAIN 1ST. ARG. INSTRUCTION
00023	0600	00	0	77772		STZ	CINST	
00024	-0620	00	0	77772		SLQ	CINST	
00025	0534	00	2	00022		LXA	A1,2	
00026	-0634	00	2	00426		SXD	TXI1,2	INITIALIZE DIMENSION COUNTERS
00027	-0634	00	2	00422		SXD	TXI2,2	
00030	0534	00	2	00031		LXA	A2,2	
00031	-0500	00	4	00002	A2	CAL	2,4	LOOK FOR POSSIBLE 2ND
00032	0621	00	0	00040		STA	CLAR2	ARG. IN CALLING SEQUENCE
00033	-0320	00	0	00504		ANA	LMASK	
00034	0602	00	0	77773		SLW	TEMP	
00035	0500	00	0	77773		CLA	TEMP	
00036	0402	00	0	77772		SUB	CINST	MATCH INSTRUCTIONS
00037	-0100	00	0	00060		TNZ	SEEX	
00040	0500	00	0	00000	CLAR2	CLA	**	IF MATCH, PROCESS ARGUMENT
00041	0622	00	0	00426		STD	TXI1	
00042	0601	00	0	77771		STO	NUMB	
00043	1 00001	2	00044			TXI	**+1,2,1	
00044	-0500	00	4	00003		CAL	3,4	LOOK FOR POSSIBLE 3RD
00045	0621	00	0	00053		STA	CLAR3	ARG. IN CALLING SEQUENCE
00046	-0320	00	0	00504		ANA	LMASK	
00047	0602	00	0	77773		SLW	TEMP	
00050	0500	00	0	77773		CLA	TEMP	
00051	0402	00	0	77772		SUB	CINST	MATCH INSTRUCTIONS
00052	-0100	00	0	00060		TNZ	SEEX	
00053	0560	00	0	00000	CLAR3	LDQ	**	IF MATCH, PROCESS ARGUMENT
00054	0200	00	0	77771		MPY	NUMB	
00055	0767	00	0	00021		ALS	17	

00056	0622 00 0 00422	STD	TXI2	
00057	1 00001 2 00060	TXI	++1,2,+1	
00060	-0754 00 2 00000	SEEX PXD	,2	
00061	0771 00 0 00022	ARS	18	
00062	0621 00 0 00461	STA	OUT+3	STORE EXIT ADDRESS
00063	0762 00 0 00321	READ RCD		READ A CARD
00064	0534 00 1 00070	LXA	A15,1	
00065	0600 00 1 77777	STZ	INDIC+15,1	CLEAR ALL INDICATORS
00066	2 00001 1 00065	TIX	*-1,1,1	
00067	-0534 00 7 00122	LXD	D24,7	
00070	-0754 00 0 00017	A15 PXD	15,0	
00071	-0700 00 1 77760	CPY CAD	ROBIN+24,1	COMPUTE LOGICAL SUM
00072	1 00000 0 00076	IX1 TXI	++4,,**	
00073	0502 00 0 00474	CLS	BBX	
00074	0120 00 0 00102	TPL	DSTOP	BRANCH TO STOP ON 2ND EOF
00075	1 00000 0 00456	IX2 TXI	OUT,0,**	
00076	2 00001 1 00071	TIX	CPY,1,1	
00077	-0100 00 0 00104	TNZ	NBLNK	BRANCH ON NON-BLANK CARD
00100	0502 00 0 00474	CLS	BBX	
00101	-0120 00 0 00063	TMI TMI	READ	READ AGAIN IF 1ST BLANK
00102	0420 06 0 60606	DSTOP HPR	24966,0,6	STOP ON 2ND BLANK OR END-DATA CARD
00103	1 00000 0 00063	IX4 TXI	READ,0,**	
00104	-0500 00 2 77760	NBLNK CAL	ROBIN+24,2	LOOK FOR AN END-DATA CARD
00105	0771 00 0 00042	ARS	34	
00106	-0100 00 0 00123	TNZ	NBBX	BRANCH IF COLS. 1 AND 2 NON-BLANK
00107	2 00002 2 00104	TIX	*-3,2,2	
00110	0502 00 0 00474	ENDAT CLS	BBX	END-DATA CARD FOUND
00111	0020 00 0 00120	NSWT1 TRA	DTEST	BRANCH IF NO SETUP
00112	0120 00 0 00114	TPL	++2	
00113	0601 00 0 00474	STO	BBX	BBX NEG. IF 1ST END-DATA
00114	-0534 00 1 00000	NSIG1 LXD	**,1	
00115	1 00001 1 00116	TXI	++1,1,+1	INCREMENT END COUNTER
00116	-0634 00 1 00000	NSIG2 SXD	**,1	
00117	0020 00 0 00456	TRA	OUT	
00120	0120 00 0 00102	DTEST TPL	DSTOP	BRANCH TO STOP ON 2ND END-DATA
00121	0601 00 0 00474	STO	BBX	BBX NEG. IF 1ST END-DATA
00122	1 00030 0 00456	D24 TXI	OUT,0,24	EXIT ON 1ST END-DATA

00123	0600 00 0 00474	NBBX	STZ	BBX	RESET BBX TO ZERO
00124	0020 00 0 00126	NSWT2	TRA	**+2	BRANCH IF NO SETUP
00125	0600 00 0 00000	NSIG3	STZ	**	RESET END-DATA COUNTER
00126	-0534 00 1 00141		LXD	D00,1	X1 = FIELD TYPE COUNTER
00127	-0634 00 4 00133	SET	SXD	LOR,4	SET UP IMAGE LEFT OR RIGHT
00130	-0534 00 4 00143		LXD	D37,4	X4 = COLUMN COUNTER
00131	-2 00001 4 00127	LOOK	TNX	SET,4,1	
00132	-0534 00 2 00122		LXD	D24,2	X2 = WORD SELECTOR
00133	2 00000 2 00134	LOR	TIX	**+1,2,**	
00134	0560 00 2 77760	AGAN	LDQ	ROBIN+24,2	LOOK AT A COLUMN BIT
00135	-0773 00 4 00044		RQL	36,4	
00136	0162 00 0 00144		TQP	TIX2	
00137	0600 00 0 77760		STZ	BLNK2	NOT BLANK COLUMN
00140	3 00004 2 00146		TXH	TROW8,2,4	BRANCH IF 0-9 PUNCH
00141	-3 00000 1 00435	D00	TXL	BADCD,1,0	BRANCH IF ADDRESS
00142	-3 00002 2 00177		TXL	YPLUS,2,2	BRANCH IF PLUS
00143	1 00045 0 00202	D37	TXI	YMIN,0,37	BRANCH IF MINUS
00144	2 00002 2 00134	TIX2	TIX	AGAN,2,2	
00145	1 00016 0 00303	D14	TXI	YBLNK,0,14	BRANCH IF BLANK COLUMN
00146	3 00026 2 00230	TROW8	TXH	DIGIT,2,22	TEST FOR 8 ROW PUNCH
00147	-3 00024 2 00165		TXL	TROW5,2,20	
00150	0560 00 2 77772		LDQ	ROBIN+34,2	CHECK FOR COMMA OR PERIOD
00151	-0773 00 4 00044		RQL	36,4	
00152	0162 00 0 00230		TQP	DIGIT	
00153	0560 00 2 00000		LDQ	ROBIN+40,2	CHECK FOR COMMA
00154	-0773 00 4 00044		RQL	36,4	
00155	0162 00 0 00157		TQP	C	
00156	1 00000 0 00221		TXI	YCOMA,0,0	BRANCH IF COMMA
00157	-3 00000 1 00230	C	TXL	DIGIT,1,0	BRANCH IF ADDRESS
00160	3 00001 1 00264		TXH	F,1,1	BRANCH IF EXP
00161	0560 00 2 00004		LDQ	ROBIN+44,2	CHECK FOR PERIOD
00162	-0773 00 4 00044		RQL	36,4	
00163	0162 00 0 00234		TQP	E	
00164	1 00000 0 00216		TXI	YPERD,0,0	BRANCH IF PERIOD
00165	3 00020 2 00230	TROW5	TXH	DIGIT,2,16	TEST FOR 5 ROW PUNCH
00166	-3 00016 2 00230		TXL	DIGIT,2,14	
00167	0560 00 2 77776		LDQ	ROBIN+38,2	CHECK FOR E-TYPE FORMAT
00170	-0773 00 4 00044		RQL	36,4	
00171	0162 00 0 00230		TQP	DIGIT	BRANCH IF NOT AN E

00172	-3	00000	1	00230	YEMAT	TXL	DIGIT,1,0	BRANCH IF ADDRESS
00173	3	00001	1	00230		TXH	DIGIT,1,1	BRANCH IF EXP
00174	0502	00	0	00174		CLS	*	
00175	0601	00	0	77761		STO	EIND	SET EXPONENT INDICATOR
00176	1	00001	1	00131		TXI	LOOK,1,1	
00177	0500	00	0	77762	YPLUS	CLA	SIND	YES PLUS
00200	0120	00	0	00131		TPL	LOOK	
00201	0020	00	0	00211		TRA	YEXPO	
00202	0500	00	0	77762	YMINUS	CLA	SIND	YES MINUS
00203	-0120	00	0	00207		TMI	**+4	
00204	0502	00	0	77770		CLS	FRACT	SET FRACTION MINUS
00205	0601	00	0	77770		STO	FRACT	
00206	0020	00	0	00131		TRA	LOOK	
00207	0502	00	0	77767		CLS	EXP	SET EXPONENT MINUS
00210	0601	00	0	77767		STO	EXP	
00211	0500	00	0	77761	YEXPO	CLA	EIND	
00212	-0120	00	0	00131		TMI	LOOK	BRANCH IF ALREADY EXP
00213	0502	00	0	00213		CLS	*	
00214	0601	00	0	77761		STO	EIND	SET EXPONENT INDICATOR
00215	1	00001	1	00131		TXI	LOOK,1,1	
00216	0502	00	0	00216	YPERD	CLS	*	YES PERIOD
00217	0601	00	0	77764		STO	DECPT	
00220	0020	00	0	00131		TRA	LOOK	
00221	3	00000	1	00316	YCOMA	TXH	FPCON,1,0	BRANCH IF NOT ADDRESS
00222	0500	00	0	77772		CLA	NUMB2	
00223	0601	00	0	77773		STO	NUMB3	
00224	0500	00	0	77771		CLA	NUMB	
00225	0601	00	0	77772		STO	NUMB2	
00226	0600	00	0	77771		STZ	NUMB	
00227	0020	00	0	00131		TRA	LOOK	
00230	3	00000	1	00234	DIGIT	TXH	E,1,0	BRANCH IF NOT ADDRESS
00231	0502	00	0	00231		CLS	*	
00232	0601	00	0	77763		STO	BLNK1	
00233	0020	00	0	00264		TRA	F	
00234	3	00001	1	00264	E	TXH	F,1,1	BRANCH IF EXP
00235	0502	00	0	00235		CLS	*	
00236	0601	00	0	77762		STO	SIND	

00237	3	00006	2	00251		TXH	EE,2,6	BRANCH IF NOT A 0 PUNCH
00240	-3	00004	2	00251		TXL	EE,2,4	
00241	0500	00	1	77771		CLA	NUMB,1	
00242	-0100	00	0	00251		TNZ	EE	BRANCH IF NUMB NOT ZERO
00243	0500	00	0	77764		CLA	DECPT	
00244	0120	00	0	00131		TPL	LOOK	BRANCH IF NO DECIMAL POINT YET
00245	0500	00	0	77766		CLA	EXPA	
00246	0402	00	0	00475		SUB	ONE	IGNORE LEADING ZEROS
00247	0601	00	0	77766		STO	EXPA	
00250	0020	00	0	00131		TRA	LOOK	
00251	0500	00	0	77765	EE	CLA	DIV	
00252	0402	00	0	00477		SUB	TEN	
00253	0100	00	0	00131		TZE	LOOK	BRANCH IF 10 DIGITS ALREADY READ
00254	0400	00	0	00477		ADD	TEN	
00255	0400	00	0	00475		ADD	ONE	ADD ONE TO DIV
00256	0621	00	0	77765		STA	DIV	
00257	0500	00	0	77764		CLA	DECPT	
00260	-0120	00	0	00264		TMI	F	BRANCH IF PERIOD BEFORE
00261	0500	00	0	77766		CLA	EXPA	ADD ONE TO EXPA
00262	0400	00	0	00475		ADD	ONE	
00263	0601	00	0	77766		STO	EXPA	
00264	0560	00	1	77771	F	LDQ	NUMB,1	MULTIPLY NUMBER BY TEN
00265	0200	00	0	00477		MPY	TEN	
00266	-0600	00	1	77771		STQ	NUMB,1	
00267	1	77773	2	00270		TXI	*+1,2,-5	ADD THIS TO NUMBER
00270	-0754	00	2	00000		PXD	0,2	
00271	0771	00	0	00023		ARS	19	
00272	0361	00	1	77771		ACL	NUMB,1	
00273	0602	00	1	77771		SLW	NUMB,1	
00274	1	00003	2	00275		TXI	*+1,2,3	CHECK FOR ANYTHING ELSE
00275	0560	00	2	77760		LDQ	ROBIN+24,2	
00276	-0773	00	4	00044		RQL	36,4	
00277	0162	00	0	00301		TQP	*+2	
00300	0020	00	0	00435		TRA	BADCD	
00301	2	00002	2	00275		TIX	*-4,2,2	
00302	0020	00	0	00131		TRA	LOOK	
00303	3	00000	1	00314	YBLNK	TXH	SBLNK,1,0	MUST BE BLANK COLUMN
00304	0502	00	0	77760		CLS	BLNK2	TEST FOR SECOND BLANK
00305	0120	00	0	00063		TPL	READ	

00306	0601	00	0	77760	STO	BLNK2	
00307	0502	00	0	77763	CLS	BLNK1	
00310	0120	00	0	00313	TPL	TXILK	
00311	0601	00	0	77763	STO	BLNK1	
00312	0020	00	0	00131	TRA	LOOK	
00313	1 00001	1	00131	TXILK TXI	LOOK,1,1		
00314	0502	00	0	00314	SBLNK CLS	*	SET BLNK2 ON
00315	0601	00	0	77760	STO	BLNK2	
00316	0500	00	0	77764	FPCON CLA	DECPT	
00317	-0120	00	0	00325	FXOFL TMI	FLOAT	
00320	0500	00	0	77761	CLA	EIND	
00321	-0120	00	0	00325	TMI	FLOAT	
00322	0534	00	1	77770	FIXED LXA	FRACT,1	
00323	-0754	00	1	00000	PXD	0,1	
00324	0020	00	0	00407	TRA	LDQF	
00325	0500	00	0	77770	FLOAT CLA	FRACT	
00326	0760	00	0	00003	SSP		
00327	0100	00	0	00411	TZE	LDQF+2	
00330	0500	00	0	77767	CLA	EXP	
00331	0400	00	0	77766	ADD	EXPA	
00332	0601	00	0	77767	STO	EXP	
00333	0534	00	1	77765	LXA	DIV,1	DIVIDE FRACT BY 10 TO THE DIV
00334	0560	00	0	77765	LDQ	DIV	
00335	0200	00	0	00476	MPY	THREE	
00336	0763	00	0	00043	LLS	35	
00337	0621	00	0	00342	STA	*+3	
00340	0560	00	0	77770	LDQ	FRACT	
00341	0200	00	1	00474	MPY	BBX,1	
00342	0765	00	0	00000	LRS	**	
00343	-0600	00	0	77770	STQ	FRACT	
00344	0560	00	0	77767	LDQ	EXP	FORM (EXP)(LN10/LN2)
00345	0200	00	0	00501	MPY	LNLN	
00346	0763	00	0	00002	LLS	2	
00347	0767	00	0	00033	ALS	27	
00350	0601	00	0	77767	STO	EXP	
00351	-0200	00	0	00502	MPR	LN	FORM R*LN2/4
00352	0765	00	0	00002	LRS	2	
00353	0601	00	0	77762	STO	EKS	
00354	0600	00	0	77764	STZ	JAY	

00355	0600	00	0	77765	STZ	SUM	
00356	0500	00	0	00503	CLA	SXNTH	1/16
00357	0601	00	0	77766	STEA	TEA	
00360	0400	00	0	77765	ADD	SUM	
00361	0601	00	0	77765	STO	SUM	(1/16)(E**X)
00362	0500	00	0	77764	CLA	JAY	
00363	0400	00	0	00475	ADD	ONE	
00364	0601	00	0	77764	STO	JAY	
00365	-0754	00	0	00016	A14	PXD	14,0
00366	0560	00	0	77766	LDQ	TEA	CLEAR AC
00367	0763	00	0	00002	LLS	2	4T
00370	0221	00	0	77764	DVP	JAY	4T/J
00371	-0200	00	0	77762	MPR	EKS	(X/4)(4T/J)
00372	-0100	00	0	00357	TNZ	STEA	BRANCH UNTIL CONVERGES
00373	0560	00	0	77765	LDQ	SUM	
00374	-0200	00	0	77770	MPR	FRACT	
00375	0534	00	1	00403	LXA	**6,1	FLOAT NUMBER
00376	0763	00	0	00001	LLS	1	
00377	-0760	00	0	00001	PBT		
00400	1	77777	1	00376	TXI	*-2,1,-1	
00401	0765	00	0	00011	LRS	9	
00402	0602	00	0	77762	SLW	EKS	
00403	-0754	00	1	00204	PXD	132,1	
00404	0767	00	0	00011	ALS	9	
00405	0400	00	0	77762	ADD	EKS	
00406	0400	00	0	77767	ADD	EXP	
00407	0560	00	0	77770	LDQF	FRACT	
00410	0763	00	0	00000	LLS	0	SIGN TO NUMBER
00411	0534	00	1	77773	LXA	NUMB3,1	
00412	0534	00	2	77771	LXA	NUMB,2	
00413	3	00000	1	00421	TXH	3D,1,0	
00414	0534	00	1	77772	LXA	NUMB2,1	
00415	3	00000	1	00425	TXH	2D,1,0	
00416	0534	00	1	77771	LXA	ADDR,1	
00417	3	00000	1	00430	TXH	STORE,1,0	
00420	0020	00	0	00431	TRA	CLEAR	
00421	-3	00001	2	00424	3D	TXL	**3,2,1
00422	1	00000	1	00423	TXI2	TXI	**1,1,**
00423	1	77777	2	00421	TXI	*-2,2,-1	
00424	0534	00	2	77772	LXA	NUMB2,2	

00425	-3	00001	2	00430	2D	TXL	++3,2,1	
00426	1	00000	1	00427	TXI1	TXI	++1,1,**	
00427	1	77777	2	00425		TXI	*-2,2,-1	
00430	0601	00	1	00000	STORE	STO	**,1	
00431	0534	00	1	00365	CLEAR	LXA	A14,1	
00432	0600	00	1	77777		STZ	INDIC+15,1	CLEAR MOST INDICATORS
00433	2	00001	1	00432		TIX	*-1,1,1	
00434	1	77777	1	00131		TXI	LOOK,1,-1	

00435	0766	00	0	00361	BADCD	WPR		IF BAD CARD - SAY SO
00436	-0534	00	1	00455		LXD	D12,1	
00437	-0534	00	6	00122		LXD	D24,6	
00440	0760	00	0	00364		SPR	4	
00441	0700	00	1	00521		CPY	ERMSG+12,1	
00442	0700	00	2	77760		CPY	ROBIN+24,2	
00443	1	77777	1	00444		TXI	++1,1,-1	
00444	2	00002	2	00441		TIX	*-3,2,2	
00445	0766	00	0	00361		WPR		
00446	0760	00	0	00365		SPR	5	
00447	0760	00	0	00371		SPR	9	
00450	0700	00	4	77761		CPY	ROBIN+25,4	
00451	0700	00	0	00507		CPY	ZWORD	
00452	2	00002	4	00450		TIX	*-2,4,2	
00453	0760	00	0	00363		SPR	3	
00454	0420	07	0	70707	ESTOP	HPR	29127,,7	
00455	1	00014	0	00063	D12	TXI	READ,0,12	

00456	-0534	00	1	00072	OUT	LXD	IX1,1	RESTORE IRS
00457	-0534	00	2	00075		LXD	IX2,2	
00460	-0534	00	4	00103		LXD	IX4,4	
00461	0020	00	4	00000		TRA	**,4	

00462	+033371577317		OCT	033371577317	2**30/10,000,000,000
00463	+042270137202		OCT	042270137202	2**27/1,000,000,000
00464	+052746167043		OCT	052746167043	2**24/100,000,000
00465	+065537624654		OCT	065537624654	2**21/10,000,000
00466	+103067572027		OCT	103067572027	2**18/1,000,000
00467	+123705530434		OCT	123705530434	2**15/100,000
00470	+150667056544		OCT	150667056544	2**12/10,000
00471	+203044672274		OCT	203044672274	2**9/1,000

00472	+243656050754	OCT	243656050754	2**6/100
00473	+314631463146	OCT	314631463146	2**3/10
00474	0000 00 0 00000	BBX HTR	**	
00475	0000 00 0 00001	ONE HTR	1	
00476	0000 00 0 00003	THREE HTR	3	
00477	0000 00 0 00012	TEN HTR	10	
00500	0761 00 0 00000	NOP NOP	0,0	SWITCH INSTRUCTION
00501	+324464741140	LNLN OCT	324464741140	(LN10/LN2)/4
00502	+261344137677	LN OCT	261344137677	LN2
00503	+020000000000	SXNTH OCT	020000000000	1/16
00504	-377777000000	LMASK OCT	777777000000	
00505	+000000215000	ERMSG OCT	000000215000	
00506	+007600000370	OCT	007600000370	
00507	+000000000000	ZWORD OCT	000000000000	
00510	+000000002000	OCT	000000002000	
00511	+000000002000	OCT	000000002000	
00512	+007640100370	OCT	007640100370	
00513	+000011000000	OCT	000011000000	
00514	+000000000000	OCT	000000000000	
00515	+000024400000	OCT	000024400000	
00516	+000010000000	OCT	000010000000	
00517	+007600217370	OCT	007600217370	
00520	+000065520000	OCT	000065520000	

77760	INDIC SYN	-16	INDICATOR BANK
77760	BLNK2 SYN	INDIC	
77761	EIND SYN	INDIC+1	
77762	SIND SYN	INDIC+2	
77763	BLNK1 SYN	INDIC+3	
77764	DECPT SYN	INDIC+4	
77765	DIV SYN	INDIC+5	
77766	EXPA SYN	INDIC+6	
77767	EXP SYN	INDIC+7	
77770	FRACT SYN	INDIC+8	
77771	ADDR SYN	INDIC+9	
77772	NUMB2 SYN	INDIC+10	
77773	NUMB3 SYN	INDIC+11	

77771	NUMB	SYN	ADDR
77772	CINST	SYN	NUMB2
77773	TEMP	SYN	NUMB3
77762	EKS	SYN	SIND
77764	JAY	SYN	DECPT
77765	SUM	SYN	DIV
77766	TEA	SYN	EXPA
77730	ROBIN	SYN	INDIC-24
		END	

ROW BINARY CARD IMAGE BANK

00521 IS THE FIRST LOCATION NOT USED BY THIS PROGRAM

NO ERROR IN ABOVE ASSEMBLY

END OF JOB