

SEL PROGRAM LIBRARY

PROGRAM DESCRIPTION

Page 1 of 1

Catalog No. 300019A

IDENTIFICATION: 810A ASCII to EBCDIC conversion routine

AUTHOR: W. A. Speer

ACCEPTED: August 4, 1967

PURPOSE: To translate a buffer of data from ASCII to EBCDIC

COMPUTER
CONFIGURATION: SEL 810A

SUBROUTINES
REQUIRED: None

STORAGE: 76₈ locations

TIMING: 15 + 24·N cycles - approx. where N=number of words in
buffer

LOADING
PROCEDURE: The program which calls this subroutine and this subroutine
must be loaded with the 810A Standard Paper Tape Load/
Dump (Catalog number 300001B)

USE: Calling Sequence:

```
CALL    ATØE
DAC     BUF Address of buffer
DATA    N    Word count
```

The ASCII character to be translated must be in bits 8-15 of
the word. Bits 0-7 are ignored. THE EBCDIC character is
returned in bits 0-7 and bits 8-15 are set to zero.

METHOD: The 2 most significant bits of the ASCII character are dropped
and an indexed table look-up is performed.

1

```

0001 00000 00000000 *****
0002 00000 00000000 *
0003 00000 00000000 *
0004 00000 00000000 * 810A ASCII TO EBCDIC CONVERSION ROUTINE          CATALOG NO. 300019A *
0005 00000 00000000 *
0006 00000 00000000 * PROGRAMMER W. A. SPEER *
0007 00000 00000000 *
0008 00000 00000000 * AUGUST 4, 1967 *
0009 00000 00000000 *
0010 00000 00000000 * PURPOSE - TO TRANSLATE A BUFFER FROM FULL ASCII TO EBCDIC. *
0011 00000 00000000 * THE ASCII CHARACTER IS TO BE IN BITS 8-15, BITS 0-7 ARE *
0012 00000 00000000 * IGNORED UPON ENTRANCE TO THE SUBROUTINE. AN EBCDIC CHARACTER *
0013 00000 00000000 * IS RETURNED IN BITS 0-7 AND BITS 8-15 ARE SET TO ZERO. *
0014 00000 00000000 *
0015 00000 00000000 * CALLING SEQUENCE *
0016 00000 00000000 * CALL AT0E *
0017 00000 00000000 * DAC BUF ADDRESS OF FIRST WORD OF BUF *
0018 00000 00000000 * DATA N WORD COUNT *
0019 00000 00000000 *
0020 00000 00000000 *****
0021 00000 00000000 REL *
0022 00000 50000000 NAME AT0E,AT0E *
0022 00000 00000000 *
0022 00000 00212017 *
0022 00000 01220040 *
0023 00000 00000000 AT0E *** ** *
0024 00001 01300000 LAA* AT0E PICK UP ADDRESS OF BUF *
0025 00002 14100000 IMS AT0E INCREMENT TO WD COUNT *
0026 00003 03100033 STA ADDR STORE ADDRESS OF FIRST WD OF BUF *
0027 00004 01300000 LAA* AT0E PICK UP WORD COUNT *
0028 00005 14100000 IMS AT0E INCREMENT FOR RETURN *
0029 00006 00000002 NEG NEGATE WORD COUNT *
0030 00007 03100032 STA WDCT STORE NEGATIVE WORD COUNT *
0031 00010 01300033 LOOP LAA* ADDR PICK UP DATA WD *
0032 00011 02100034 LBA MSK1 PICK UP MASK- *
0033 00012 00000027 ABA 'AND' TO GIVE TRUNCATED ASCII *
0034 00013 00000112 FRA 1 PUSH OFF LAST BIT INTO B ACC *
0035 00014 00000006 IAB INTERCHANGE TO TEST LAST BIT *
0036 00015 00000052 SN0 WAS THE LAST BIT ON *

```

0037	00016	11100022	BRU	EVEN	N0 - THEN CONSIDER THE N0. EVEN	
0038	00017	01500036	0DD	LAA	TABL,1	YES - 0DD - PICK UP THE ENTRY
0039	00020	00000000	*			FROM THE TABLE USING THE
0040	00020	00000000	*			TRUNCATED ASCII WITHOUT ITS
0041	00020	00000000	*			LAST BIT AS AN INDEX
0042	00020	00001016	LSL	8		SINCE THE DATA WD IS 0DD
0043	00021	00000000	*			DROP THE LEFTMOST 8 BITS
0044	00021	00000000	*			LEAVING THE EBCDIC EQUIVALENT
0045	00021	00000000	*			IN THE FIRST 8 BITS OF ACCUM
0046	00021	11100025	BRU	ARND		GO TO THE STORE ROUTINE
0047	00022	01500036	EVEN	LAA	TABL,1	PICK UP THE ENTRY FROM THE
0048	00023	00000000	*			TABLE USING THE TRUNCATED ASCII
0049	00023	00000000	*			WITHOUT ITS LAST BIT AS AN INDEX
0050	00023	02100035	LBA	MSK2		PICK UP MASK, 'AND' TO DROP
0051	00024	00000027	ABA			OFF THE RIGHTMOST 8 BITS LEAVING
0052	00025	00000000	*			THE EBCDIC EQUIVALENT IN THE
0053	00025	00000000	*			LEFTMOST 8 BITS OF ACCUM
0054	00025	03300033	ARND	STA*	ADDR	STORE EBCDIC IN BUF
0055	00026	14100033		IMS	ADDR	INCREMENT TO NEXT WD IN BUF
0056	00027	14100032		IMS	WDCT	INCREMENT NEG WD CT - IS IT ZERO
0057	00030	11100010	BRU	L00P		N0 - GO BACK FOR ANOTHER WD
0058	00031	11300000	BRU*	AT0E		YES - FINISHED - RETURN
0059	00032	00000000	WDCT	***	**	
0060	00033	00000000	ADDR	***	**	
0061	00034	00000077	MSK1	DATA	'000077	
0062	00035	00177400	MSK2	DATA	'177400	
0063	00036	00076301	TABL	DATA	'076301	300-301
0064	00037	00141303		DATA	'141303	302 - 303
0065	00040	00142305		DATA	'142305	304 - 305
0066	00041	00143307		DATA	'143307	306 - 307
0067	00042	00144311		DATA	'144311	310 - 311
0068	00043	00150722		DATA	'150722	312 - 313
0069	00044	00151724		DATA	'151724	314 - 315
0070	00045	00152726		DATA	'152726	316 - 317
0071	00046	00153730		DATA	'153730	320 - 321
0072	00047	00154742		DATA	'154742	322 - 323
0073	00050	00161744		DATA	'161744	324 - 325
0074	00051	00162746		DATA	'162746	326 - 327
0075	00052	00163750		DATA	'163750	330 - 331
0076	00053	00164515		DATA	'164515	332 - 333

0077	00054	00060535	DATA	'060535	334 - 335
0078	00055	00003407	DATA	'003407	336 - 337
0079	00056	00040132	DATA	'040132	240 - 241
0080	00057	00077573	DATA	'077573	242 - 243
0081	00060	00055554	DATA	'055554	244 - 245
0082	00061	00050175	DATA	'050175	246 - 247
0083	00062	00046535	DATA	'046535	250 - 251
0084	00063	00056116	DATA	'056116	252 - 253
0085	00064	00065540	DATA	'065540	254 - 255
0086	00065	00045541	DATA	'045541	256 - 257
0087	00066	00170361	DATA	'170361	260 - 261
0088	00067	00171363	DATA	'171363	2262 - 263
0089	00070	00172365	DATA	'172365	264 - 265
0090	00071	00173367	DATA	'173367	266 - 267
0091	00072	00174371	DATA	'174371	270 - 271
0092	00073	00075136	DATA	'075136	272 - 273
0093	00074	00046176	DATA	'046176	274 - 275
0094	00075	00067157	DATA	'067157	276 - 277
0095	00076	70400000	END		