



# DECUS

## PROGRAM LIBRARY

DECUS NO.	8-285
TITLE	TELETYPE INPUT-OUTPUT PACKAGE
AUTHOR	Garth Peterson
COMPANY	Institute of Atmospheric Sciences South Dakota School of Mines and Technology Rapid City, South Dakota
DATE	December 9, 1969
SOURCE LANGUAGE	PALD



Title: Teletype Input-Output Package

Author: Garth Peterson

Date: 2 December 1969

Name: TTYP

Programming language: PALD

Core usage: One page, plus location 0.

Abstract:

This is a teletype control package containing subroutines for single-character input and output and for output of packed and open text. These subroutines are single-field oriented, but provide for interrupt-enabled and disabled operation.

Description:

There are nine subroutines:

<u>Name</u>	<u>Function</u>
KTREAD	Read one input character unconditionally.
KTGET	Get contents of keyboard buffer (conditional input).
KTCL	Clear keyboard buffer.
KTCSV	Service keyboard flag interrupt.
KTWRITE	Write one output character.
KTTSV	Service teletype output flag interrupt.
KTACPT	Accept keyboard input character and echo it.
TYPOP	Type open text.
TYPPAK	Type packed text.

KTREAD, KTGET, and KTACPT return to the calling program with data in the accumulator; all others return with the accumulator cleared to zero. KTWRITE takes data from the accumulator upon entry; all others clear the accumulator at the start.

The design of these subroutines is based on the following interrupt philosophy: Under interrupt-disabled operation the user's program consists of a single process which includes all calculations, decision branches, and peripheral device service. The operation of peripherals simultaneously with one another and with the calculating and branching parts of the program is thus limited to the inherent buffering capabilities of the peripheral hardware. Under interrupt-enabled operation the user's program contains first a main process, which makes calculations and decisions and which initiates peripheral operations, and second an interrupt service process, which clears flags and issues such commands as are required by the current states of the peripheral devices [8]. Whenever the interrupt service determines that no more devices currently require service, execution of the main process must resume at the point of interruption. The main process runs primarily with the interrupt facility enabled but with it disabled when instructions are executed which are common to both processes. Thus this teletype package is unsuitable for time-sharing systems, but it can run only one teletype anyway. Interrupt-enabled operation is made to resemble interrupt-disabled operation as closely as possible; therefore buffering of the teletype is actually or virtually limited to hardware buffering, and multiple entry points for the individual subroutines are avoided. Other peripherals can run simultaneously with the teletype as permitted by program logic and the interrupt facility.

If the calling program enables the interrupt facility, the interrupt flag identification routine should contain some coding sequence such as:

```
KSF
SKP
JMS KTKSV
TSF
SKP
JMS KTTSV
```

KTKSV and KTTSV should be called only while the interrupt facility is disabled and only in response to interrupts. No references in the calling program to these two subroutines are needed if the interrupt facility is always disabled. All the remaining subroutines may be called with the interrupt facility enabled or disabled, but only as part of the main process.

The text output subroutines have calling sequences as follows:

```
JMS TYPOPV
AREA

JMS TYPPAK
AREA
```

"AREA" is the address of the first word of the text. The separation of the character string from the subroutine call allows more efficient use of PDP-8 memory pages. TYPOPV outputs open text consisting of 8-bit ASCII with one right-justified character per memory word. The text string is terminated by a zero word; however, blank tape can still be punched by means of a non-zero word which is zero in the low-order 8 bits. When writing a program, certain characters, such as carriage return, line feed, rubout, and back arrow, must be written in octal to avoid problems with the Editor program. Most printing characters can be written explicitly, using the single-character assembly facility of PALD or MACRO-8 [2,5]. For example:

```
MESG, 215;212;"M;"E;"S;"S;"A;"G;"E;0
```

At least three characters in the source program are needed for each character in the text, so that open text is inefficient in both the source program and object program. However, open text is more convenient as program-generated text, such as numerical output.

TYPPAK outputs 6-bit stripped ASCII, packed two characters per word, as assembled by PALD or MACRO-8 when the pseudo-op "TEXT" is used [3,6]. Permissible output characters are space, carriage return, line feed, and all the graphics except the at (@), per cent (%), and pound (#) signs. The text is terminated by 00 (octal), which is generated by the assembler, and which is what "@" in the text will compress to. The at sign is therefore a logical choice for a text delimiter. The

per cent and pound signs are compressed in the usual way by the assembler but then converted by TYPPAK to carriage return and line feed respectively [1]. The back arrow is excluded by the Editor, rather than by TYPPAK or the assembler. In addition, if the Editor converts successive spaces in the text to a single tabulate character, the final result in the output will be "I".

Both text subroutines call KTWRTTE, which the calling program may also access directly. KTWRTTE loops until the teletype flag rises, or until enough time has elapsed to assure that no flag is pending, and then issues the output command [7]. The entry point KTWRTTE is immediately followed by a skip and return:

```

KTWRTTE, 00
  SKP
  JMP I KTWRTTE
  . . . . .

```

This allows the output of TYPPAK to be diverted to another device when the skip is replaced by a call to the subroutine for that device. For example, suppose "PTWRTTE" is the name of a fast punch subroutine. Then output can be diverted and restored thus:

```

TAD (JMS I [PTWRTTE])
DCA KTWRTTE+1

TAD (SKP)
DCA KTWRTTE+1

```

Note that "PTWRTTE" must return to KTWRTTE+2 with the accumulator cleared. KTWRTTE must be in its normal condition whenever KTACPT is called. For the PDP-8/S the supplied version of KTWRTTE should be replaced by:

```

00
00
00
KTWRTTE, 00
  SKP
  JMP I KTWRTTE
  DCA KTWRTTE-2
  DCA KTWRTTE-3
  ISZ KTWRTTE-3
  TAD KTWRTTE-1
  SNA CLA
  JMP .+3
  TSF
  JMP .-5
  TAD KTWRTTE-2
  TLS
  STA
  DCA KTWRTTE-1
  JMP I KTWRTTE

```

The input subroutines KTKCL, KTGET, and KTKCL provide access to the "keyboard buffer," which is an abstraction but which has these properties from the viewpoint of the calling program: The buffer is cleared, or voided, by a program command; keyboard input data loads the buffer, which then remains loaded with this input character until cleared by the program or until reloaded by new input data. KTKCL clears the buffer and should be called before data is read from the teletype to avoid inputting spurious characters. KTKCL provides unconditional input; after entry KTKCL loops until the keyboard buffer contains data, then clears the buffer, and returns with the input data in the accumulator and also in core location KTNEXT. KTGET is a conditional input subroutine which shows the current contents of the keyboard buffer in the accumulator but which does not clear the buffer. If the buffer is void, it sets the accumulator to -1. KTGET is called by KTKCL and may be called directly by the user's program to determine whether or not a teletype key has been pressed. In order to determine the status of the interrupt facility, KTGET accesses location 0; the calling program must therefore avoid this location even if the interrupt facility is never used. The interrelation of KTKCL, KTGET, and KTKCL may be made clearer by the following examples of how these subroutines would be written if the interrupt facility were always disabled. Here the "keyboard buffer" is in fact the hardware buffer, provided that a keyboard flag must be present for the buffer to be considered loaded.

```

KTKCL, 00
  KCC
  JMP I KTKCL
/
KTGET, 00
  CLA
  KRS
  KSF
  STA
  JMP I KTGET
/
KTKCL, 00
  JMS KTGET
  SPA
  JMP .-2
  DCA KTNEXT
  KCC
  TAD KTNEXT
  JMP I KTKCL
KTNEXT, 00

```

Interrupt-enabled operation and teletype paper tape input are not fully compatible. This is a hardware characteristic and cannot readily be offset in programming; a fairly large input buffer in memory is usually required. If KTKCL is used for paper tape input with the interrupt facility enabled, successive calls must occur within 100 milliseconds

to avoid loss of data; if the teletype reader is ready, the tape will advance continuously even when input is not requested. Under interrupt-disabled operation the tape advances by one character for each call to KTREAD.

KTACPT inputs characters from the keyboard and echoes them on the teleprinter. It first clears the keyboard buffer, then reads one character, echoes it, and returns with the character in the accumulator and in KTNEXT. All characters are typed as read except carriage return, which is echoed as carriage return plus line feed. KTACPT is not suitable for paper tape input because of this double echo and because KTKCL should be called only once at the beginning of a paper tape, not repeatedly while the tape is read. Paper tape input with teletype echo is performed by this coding sequence:

```
JMS KTREAD
JMS KTWRITE
TAD KTNEXT
```

This teletype control package is intended for use by programs which run entirely within 4096 words of memory or which use extended memory only as fast-access auxiliary data storage. Its use on a multiple-field PDP-8 is subject to these restrictions: Each subroutine must be called with the data field matching the instruction field, and it will return only to locations in the field where it resides. Location 0 must be avoided in any field where KTGET is called. Interrupt-enabled operation is limited to field 0. Text for TYPOP or TYPPAK must not "wrap around" from location 7777 to location 0.

It can be useful to patch the input and output functions of the PDP-8 floating-point interpreter [4] into the teletype package, allowing floating-point input and output with the interrupt facility enabled. The following patches will do this:

```
*7344+1 /OUTPUT PATCH
JMS I .+2
JMP I 7344
KTWRITE

*7142+1 /INPUT PATCH
JMS I [KTKCL] /OMIT IF PT INPUT
*7142+2
SKP
KTREAD
JMS I .-1
```

These patches should be assembled along with the main program, which must be loaded into core after the floating-point interpreter.

This subroutine package is supplied in ASCII, both with and without comments, and can be assembled as one page of core in the user's program. The first line has the pseudo-op "PAGE", which is the only exception to PAL III, and the last line is a dollar sign. There are 11 labels and no references to auto-index registers.



References:

1. Digital Equipment Corporation, Alphanumeric Message Typeout, Digital-8-18-Sym, February 16, 1967.
2. idem, PAL-D Disk Assembler Programmer's Reference Manual, DEC-D8-ASAA-D, April 1968, p. 1-12.
3. ibidem, p. 2-2.
4. idem, PDP-8 Floating-Point System Programming Manual, DEC-08-YQYA-D, 1968.
5. idem, MACRO-8 Programming Manual, DEC-08-CMAA-D, 1965, p. 5-12.
6. ibidem, pp. 6-3, 6-4.
7. Fichtenbaum, Matthew L., to Mrs. Angela J. Cossette (DECUS Executive Secretary), June 7, 1967, distributed to members of Digital Equipment Computer Users Society.
8. Wirth, Niklaus, "On Multiprogramming, Machine Coding, and Computer Organization," Communications of the ACM, vol. 12, no. 9 (Sept. 1969), pp. 489-498.

PAGE 1 13670

// JOB T 0002 0002 0002 13670

LOG DRIVE	CART SPEC	CART AVAIL	PHY DRIVE
0000	0001	0001	0000
0001	0002	0002	0001

V2M6 16K SDSMT COMP CNTR 05DEC69

// \* GARTH PETERSON 50120

// XEQ PTTCD

KTACPT 0274  
KTGET 0213  
KTKCL 0225  
KTKSV 0232  
KTNEXT 0211  
KTPA 0351  
KTREAD 0200  
KTTSV 0265  
KTWRTE 0242  
TYPOP 0311  
TYPPAK 0325

\*

```

                PAGE
                /TTY I/O PACKAGE
                /11MR68, 30NV69
                /
0200 0000      KTREAD, 00          /SUBR, READ KEYBOARD UNCONDITIONALLY
0201 4213      JMS KTGET          /GET INPUT CHARACTER, IF ANY
0202 7510      SPA
0203 5201      JMP .-2            /IF NONE, TRY AGAIN
0204 3211      DCA KTNEXT        /DEPOSIT INPUT CHARACTER
0205 7240      STA
0206 3212      DCA KTNEXT+1      /MARK BUFFER IN CORE EMPTY
0207 1211      TAD KTNEXT
0210 5600      JMP I KTREAD      /RETURN WITH INPUT IN AC
0211 0000      KTNEXT, 00       /+0 TELETYPE INPUT WORD
0212 0000      00                /+1 TELETYPE INPUT BUFFER
                /
0213 0000      KTGET, 00        /SUBR, GET TELETYPE INPUT CHARACTER IF
0214 7200      CLA                / ANY
0215 3000      DCA 0             /LOCATION 0 HOLDS RETURN ADDRESSES OF
0216 6031      KSF                / INTERRUPTS
0217 5223      JMP .+4           /BRANCH IF NO FLAG, LOOK FOR CHARACTER
0220 1000      TAD 0             / IN CORE
0221 7650      SNA CLA           /SKIP IF FLAG CAUSED INTERRUPT
0222 4232      JMS KTKSV        /READ HARDWARE BUFFER INTO CORE
0223 1212      TAD KTNEXT+1     /GET BUFFER CHARACTER, AC= -1 IF NONE
0224 5613      JMP I KTGET
                /
0225 0000      KTKCL, 00        /SUBR, CLEAR OUT PREVIOUS KEYBOARD INPUT
0226 6032      KCC                /CLEAR HARDWARE FLAG WHEN INTERRUPT
0227 7240      STA                / DISABLED
0230 3212      DCA KTNEXT+1     /MARK BUFFER IN CORE EMPTY
0231 5625      JMP I KTKCL
                /
0232 0000      KTKSV, 00        /KEYBOARD INPUT INTERRUPT & FLAG SER' CE
0233 6036      KRB                /CLEAR FLAG, READ BUFFER, ALLOW NEW CHAR
0234 3212      DCA KTNEXT+1     /SAVE CHARACTER
0235 5632      JMP I KTKSV
                /
0236 0000      00                /-4 HIGH-ORDER TIME COUNT
0237 0000      00                /-3 LOW-ORDER TIME COUNT
0240 0000      00                /-2 OUTPUT CHARACTER
0241 0000      00                /-1 BUSY LATCH, -1 = BUSY, 0 = NOT BUSY
0242 0000      KTW RTE, 00      /SUBR, TELETYPE OUTPUT
0243 7410      SKP                /PATCH HERE TO DIVERT TO ANOTHER DEVICE
0244 5642      JMP I KTW RTE
0245 3240      DCA KTW RTE-2     /SAVE CHARACTER
0246 7346      CLL STA RTL       /AC = -3
0247 3236      DCA KTW RTE-4     /INITIALIZE TIME COUNT (140 MS MIN)
0250 2237      ISZ KTW RTE-3     /BEGIN LOOP
0251 7410      SKP
0252 2236      ISZ KTW RTE-4     /SKIP IF TIME COUNT DONE, EXPECT NO FLAG
0253 1241      TAD KTW RTE-1     /CHECK BUSY LATCH
0254 7650      SNA CLA           /SKIP IF LATCH ON AND COUNT NOT DONE
0255 5260      JMP .+3           /BRANCH IF NOT BUSY
0256 6041      TSF                /SKIP IF FLAG, INTERRUPT MAY BE DISABLED
0257 5250      JMP .-7           /PREVIOUS CHARACTER NOT DONE, TRY AGAIN
0260 1240      TAD KTW RTE-2     /GET CHARACTER
0261 6046      TLS                /START TELETYPE
0262 7240      STA
0263 3241      DCA KTW RTE-1     /MARK BUSY

```

0264	5642	JMP I KTWRT	
		/	
0265	0000	KTTSV, 00	/TELETYPE OUTPUT INTERRUPT SERVICE
0266	6042	TCF	/CLEAR FLAG
0267	7200	CLA	
0270	3241	DCA KTWRT-1	/MARK NOT BUSY
0271	5665	JMP I KTTSV	
		/	
0272	0212	212	
0273	7563	-215	
0274	0000	KTACPT, 00	/ACCEPT (AND ECHO) KEYBOARD INPUT
0275	4225	JMS KTKCL	/CLEAR KEYBOARD BUFFER
0276	4200	JMS KTREAD	/READ KEYBOARD
0277	4242	JMS KTWRT	/ECHO ON TELEPRINTER
0300	1211	TAD KTNEXT	
0301	1273	TAD KTACPT-1	/CHECK FOR CARRIAGE RETURN
0302	7640	SZA CLA	
0303	5306	JMP .+3	
0304	1272	TAD KTACPT-2	/IF CARRIAGE RETURN, SUPPLY LINE FEED
0305	4242	JMS KTWRT	
0306	1211	TAD KTNEXT	/GET CHARACTER
0307	5674	JMP I KTACPT	/RETURN
		/	
0310	0000	00	
0311	0000	TYPOPN, 00	/TYPE MESSAGE OF OPEN TEXT
0312	7200	CLA	/ JMS TYPOPN (EFFECTIVE)
0313	1711	TAD I TYPOPN	/ ADDRESS-OF-MESSAGE
0314	3310	DCA TYPOPN-1	/ (NEXT INSTRUCTION)
0315	2311	ISZ TYPOPN	
0316	1710	TAD I TYPOPN-1	
0317	2310	ISZ TYPOPN-1	
0320	7450	SNA	
0321	5711	JMP I TYPOPN	/ZERO WORD TERMINATES MESSAGE
0322	4242	JMS KTWRT	/WRITE CHARACTER
0323	5316	JMP .-5	
		/	
0324	0000	00	/ TEXT POINTER
0325	0000	TYPPAK, 00	/TYPE PACKED TEXT
0326	7200	CLA	/ JMS TYPPAK (EFFECTIVE)
0327	1725	TAD I TYPPAK	/ AREA
0330	3324	DCA TYPPAK-1	/ (NEXT INSTRUCTION)
0331	2325	ISZ TYPPAK	
0332	1724	TAD I TYPPAK-1	/GET FIRST CHAR IN WORD
0333	4351	JMS KTPA	
0334	1724	TAD I TYPPAK-1	/GET SECOND CHAR IN WORD
0335	7006	RTL	
0336	7006	RTL	/LEFT-JUSTIFY
0337	7006	RTL	
0340	4351	JMS KTPA	
0341	2324	ISZ TYPPAK-1	/INCREMENT POINTER
0342	5332	JMP TYPPAK+5	
0343	0245	245	/PER CENT
0344	7750	215-245	/CR - PER CENT
0345	7776	243-245	/POUND - PER CENT
0346	7747	212-243	/LF - POUND
0347	7535	-243	/ -POUND
0350	7700	7700	
0351	0000	KTPA, 00	/SUBR, PRINT 6-BIT ASCII LEFT
0352	0350	AND KTPA-1	/CLEAR RIGHT 6 BITS
0353	7450	SNA	

0354	5725	JMP I TYPPAK	/RETURN IF 00 CHAR (NO AT SIGN)
0355	7101	CLL IAC	/AC BIT 11 WILL GO TO CHANNEL 8
0356	7500	SMA	/LINK WILL GO TO CH 7 (= COMPLEMENT OF
0357	7120	STL	/ CH 6)
0360	7012	RTR	/AC 0-5 GO TO CHS 6-1
0361	7012	RTR	
0362	7012	RTR	/AC HOLDS 8-BIT ASCII
0363	1347	TAD KTPA-2	
0364	7450	SNA	
0365	1346	TAD KTPA-3	/CHANGE POUND SIGN TO LINE FEED
0366	1345	TAD KTPA-4	
0367	7450	SNA	
0370	1344	TAD KTPA-5	/CHANGE PER CENT TO CARRIAGE RETURN
0371	1343	TAD KTPA-6	
0372	4242	JMS KTWRT	/WRITE CHARACTER
0373	5751	JMP I KTPA	
		/END, TTY PKG	*****

