DECUS
PROGRAM LIBRARY

DECUS NO.	12-8
TITLE	TELETYPE CONVERSION ROUTINES
AUTHOR	Donald A. Overton
COMPANY	Eastern Pennsylvania Psychiatric Institute Philadelphia, Pennsylvania
DATE	Submitted: July 10, 1970
SOURCE LANGUAGE	LAP6-DIAL

TELETYPE CONVERSION ROUTINES

DECUS Program Library Write-up DECUS No. 12-8

Introduction

The teletype conversion routines BINDEC, DECBIN, BINOCT and OCTBIN provide a standardized set of subroutines for doing various types of conversion from binary integers to decimal teletype codes and vice versa. They were written by D. J. Nichols for the LINC keyboard and have been modified for the PDP-12 so that they accept or generate ASCII codes.

Similarly, the program SERINDEX by T. Gutterman has been modified so that it will search a LAP6-DIAL index for a given name.

The program TELETYPE is a new subroutine which reads or prints ASCII characters via the ASR33 teletype.

All of the above subroutines are in the LINC mode and are intended to be added to the user program source in the current instruction field.

The binaries for DAYCOM and GRENSLV are included in this tape for the amusement of PDP-12 users.

Credit for the above programs obviously belongs to the original authors. The current author has simply made the minor revisions necessary to allow these subroutines to operate on the PDP-12, as a convenience to PDP-12 users. Address all correspondence concerning these subroutines to the current author: Donald A. Overton, Eastern Pennsylvania Psychiatric Institute, 3300 Henry Avenue, Philadelphia, Pennsylvania 19129.

PROGRAM NAMES: BINDEC 1, BINDEC 2
BINDEC 3, BINDEC 4
BINDEC 5, BINDEC 7
BINDEC 8, BINDEC 9

Program Language: LAP6-DIAL

Computer: PDP-12

Programmer: Dennis J. Nichols
Laboratory Computer Facility*
University of Wisconsin

PURPOSE

The BINDEC collection of conversion routines provides a standardized set of subroutines for doing various types of conversion from binary integers to decimal teletype codes. The resulting 6-bit ASCII codes are stored in consecutive half-words of memory. All the BINDEC routines use the same index registers and similar calling sequences.

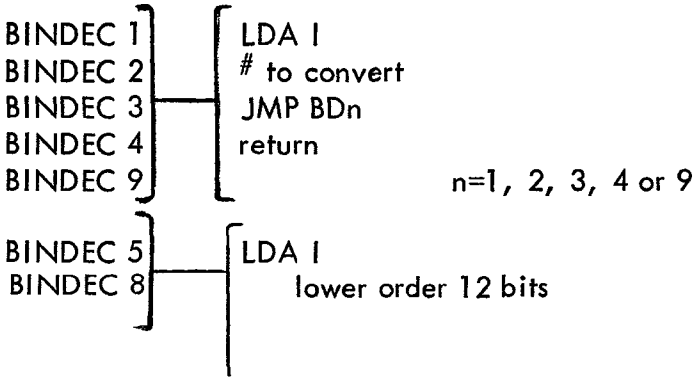
USAGE

Manuscript Implementation

The manuscript of the desired BINDEC routine must be added to the user's manuscript with the Add Program command.

Usage Procedure

The 6-bit ASCII characters produced by these subroutines are stored half-indexed via index 1. It is the calling program's job to have index 1 set one half-word in front of the position where the first character is to be stored. The calling sequences are as follows:



---

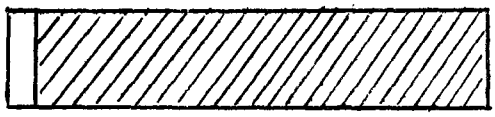
*The BINDEC collection of subroutines was developed by the Laboratory Computer Facility at the University of Wisconsin Medical Center under grant FR00249 from the Division of Research Facilities and Resources of the National Institutes of Health.

```
                    STC BDnLO
                    LDA I
                        high order 12 bits
                    STC BDnHI
                    JMP BDn
                    return                  n= 5 or 8

BINDEC 7            signed decimal
                    LDA I
                        # to convert
                    JMP BD7S
                    return

                    unsigned decimal
                    LDA I
                        # to convert
                    JMP BD7NS
                    return
```

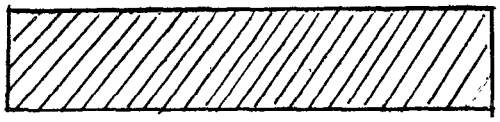
Usage Requirements

The requirements and features of the various routines are shown in Table I. The format for double-precision is understood to be the sign plus 23 bits as shown below.

	most significant	least significant
sign	11 bits	12 bits



upper 12 bits



lower 12 bits

TABLE I.

Name	Entry Point	Words Input	Conversion* Range	Characters Output	Time (msec)	Justification	Leading Zeroes ?	Routine Length (Octal)	Indices Used
BINDEC 1	BD1	1	[-2047, 2047]	5	0.28--0.7	right	blanks	65	1-3
BINDEC 2	BD2	1	[0, 4095]	4	0.28--.94	right	blanks	56	1-3
BINDEC 3	BD3	1	[-2047, 2047]	≤ 5	0.24--0.7	left	**	57	1-3
BINDEC 4	BD4	1	[0, 4095]	≤ 4	0.26--.94	left	**	52	1-3
BINDEC 5	BD5	2	[-8388607, 8388607]	8	1.14--3.9	right	blanks	141	1-3
BINDEC 7	BD7S	1	[-2047, 2047]	5	0.3--1.1	right	blanks	77	1-3
	BD7NS	1	[0, 4095]	4	0.3--1.1	"	"		
BINDEC 8	BD8	2	[0, 8388607]	7	1.1--3.8	right	blanks	122	1-3
BINDEC 9	BD9	1	[0, 99]	2	0.06--3.0	right	zeroes	32	1-2

*Negative Nos. treated as 1's complement Nos.

**In the left-justified routines, only the significant digits are stored. The field is not filled out with trailing spaces.

DECBIN

PROGRAM NAMES: DECBIN 1
DECBIN 2
DECBIN 3
DECBIN 4

Program Language: LAP6-DIAL

Computer: PDP-12

Programmer: Dennis J. Nichols
Laboratory Computer Facility*
University of Wisconsin

ABSTRACT

The DECBIN collection of conversion routines provides a standardized set of subroutines for conversion of decimal ASCII codes into binary integers.

PURPOSE

The DECBIN routines convert decimal ASCII codes stored in memory to binary integers. The routines are:

DECBIN 1: Conversion to single-precision binary. Five characters are converted to binary.

DECBIN 2: Conversion to single-precision binary. This subroutine first searches to a Qanda field delimiter. Then all characters before the next occurrence of a Qanda delimiter are converted to binary.

DECBIN 3: Conversion to double-precision binary. Ten (decimal) characters are converted to binary.

DECBIN 4: Conversion to double-precision binary. This subroutine first searches to a QANDA field delimiter. Then all characters before the next occurrence of a QANDA delimiter are converted to binary.

USAGE

Initialization

The keyboard characters converted by the DECBIN subroutines are loaded from consecutive

---

*The DECBIN collection of subroutines was developed by the Laboratory Computer Facility at the University of Wisconsin Medical Center under grant FR00249 from the Division of Research Facilities and Resources of the National Institutes of Health.

half-words of memory via index register 1. It is the calling program's job to have index register 1 set properly before entry into the subroutine.

DECBIN 1 and DECBIN 3 require index register 1 to be set one-half word before the first keyboard code to be converted. DECBIN 1 converts five characters; DECBIN 3 converts ten (decimal) characters.

DECBIN 2 and DECBIN 4 search for the delimiters which demarcate QANDA answer fields from which the characters are converted. The delimiters searched for are 74 and 34. Index register 1 may be set pointing anywhere before the first delimiter. It may not be set pointing to the first delimiter.

Execution

The calling sequence is the same for all four routines and is shown below.

```
JMP DBn                    where n= 1, 2, 3 or 4
    error return point
    normal return point
```

A number too large to be expressed in the precision being used or an illegal character causes an error return. The legal characters are the digits 0 thru 9, plus (+), minus (-), space, and, in the case of the double-precision routines, comma (,). All plusses, spaces and commas are ignored. The minus sign may appear anywhere. Unfilled half-words (ØØ) are ignored.

If the normal return is executed, then the accumulator contains:

a) The binary conversion when using DECBIN 1 or DECBIN 2, or

b) The lower order 12_{10} bits of the conversion when using DECBIN 3 or DECBIN 4. In this case, the high order bits may be obtained by loading the contents of location DBnHI where n= 3 or 4.

The indices used by these subroutines are not restored. After a normal return, index register 1 is pointing to either the last character converted (DECBIN 1 and 3) or the second delimiter (DECBIN 2 and 4).

SPECIFICATIONS

	DECBIN 1	DECBIN 2	DECBIN 3	DECBIN 4
Entry Point	DB1	DB2	DB3	DB4
Characters Input	5	*	10	*
Commas Legal?	no	no	yes	yes
Conversion Range	±4095	±4095	±4,194,303	±4,194,303
Words Output	1	1	2	2
Time (msec)	.14-.42	.14-.42	.32-1.2	.32-1.2
Routine Length (octal)	70	100	113	122
Indices Used	1-4	1-3	1-5	1-4

*These routines convert a field defined by delimiters.

BINOCT

PROGRAM NAMES: BINOCT 1
BINOCT 2

Program Language: LAP6-DIAL

Computer: PDP-12

Programmer: Dennis J. Nichols
Laboratory Computer Facility*
University of Wisconsin

PURPOSE

The BINOCT collection of conversion routines provides a standardized set of subroutines for various types of conversion from single-precision binary integers to octal teletype codes. The resulting ASCII codes are stored in consecutive half-words of memory. All the BINOCT routines use the same index registers and calling sequences.

USAGE

Manuscript Implementation

The manuscript of the desired BINOCT routine must be added to the user's manuscript with the Add Program meta-command.

Usage Procedure

The keyboard characters produced by these subroutines are stored half-indexed via index register 1. It is the calling program's job to have index register 1 set one half-word before the position where the first character is to be stored. The calling sequence for both routine is as follows:

```
LDA I
  # to convert
JMP B0n                    where n= 1 or 2
return
```

Usage Requirements

The requirements and features of these routines are shown in the following table.

---

*The BINOCT collection of subroutines was developed by the Laboratory Computer Facility at the University of Wisconsin Medical Center under grant FR00249 from the Division of Research Facilities and Resources of the National Institutes of Health.

	BINOCT 1	BINOCT 2
Entry Point	B01	B02
Words Input	1	1
Conversion Range	[0000,7777]	[0,7777]
Characters Output	4	4
Time (msec)	.12	.14 to .16
Justification	right	right
Leading Zeroes?	zeroes	blanks
Routine Length	20	34
Indices Used	1.2	1

OCTBIN

PROGRAM NAMES: OCTBIN 1
OCTBIN 2

Program Language: LAP6-DIAL

Computer: PDP-12

Programmer: Dennis J. Nichols
Laboratory Computer Facility*
University of Wisconsin

ABSTRACT

The OCTBIN collection of conversion routines provides a standardized set of subroutines for conversion of octal ASCII codes into binary integers.

PURPOSE

The OCTBIN routines convert octal keyboard codes stored in memory to single-precision binary integers. The subroutines are:

OCTBIN 1: Conversion of four characters to binary.

OCTBIN 2: Conversion of the characters which lie within a delimited field.

USAGE

Initialization

The keyboard characters converted by the OCTBIN subroutines are loaded from consecutive half-words of memory via index register 1. It is the responsibility of the calling program to have index register 1 set properly before entry into the subroutine.

OCTBIN 1 requires index register 1 to be set one-half word before the first keyboard code to be converted.

OCTBIN 2 uses delimiters to demarcate the field from which the characters are converted. The delimiting characters are those used in the QANDA answer string (i.e., 74 and 34).

Index Register 1 may be set pointing anywhere before the first delimiter. It may not be set pointing to the first delimiter.

---

*The OCTBIN collection of subroutines was developed by the Laboratory Computer Facility at the University of Wisconsin Medical Center under grant FR00249 from the Division of Research Facilities and Resources of the National Institutes of Health.

Execution

The calling sequence is the same for both routines and is shown below:

```
JMP OBn              where n= 1 or 2
    error return point
    normal return point
```

No check is made for entering a number too large to be expressed in 12 bits. Only an illegal character will cause an error return. The legal characters are the digits 0 through 7, space, and unfilled half words (ØØ). All leading, trailing and imbedded spaces and unfilled half words are ignored.

If the normal return is executed, the accumulator contains the binary conversion. After a normal return, index register 1 is pointing to either the last character converted (OCTBIN 1) or the second of a pair of delimiters (OCTBIN 2). The indices used by these subroutines are not restored.

SPECIFICATIONS

	OCTBIN 1	OCTBIN 2
Entry Point	0B1	0B2
Characters Input	4	4
Conversion Range	[0000,7777]	[0000,7777]
Words Output	1	1
Time (msec)	.12 to .16	0.1 to 0.2
Routine Length (Octal)	33	43
Indices Used	1-3	1,2

*Field determined by delimiters

TELETYPE

PROGRAM NAME: TELETYPE

Program Language: LAP6-DIAL

Computer: PDP-12

Author: D. A. Overton

TELETYPE is a LINC mode subroutine which will read the keyboard or print characters using the ASR33 teletype. It must reside in the current instruction field. The subroutine occupies 124_8 locations and has eight entry points. The functions of these entry points are defined in the listing.

A JMP to the entry point TYPNEW must be followed by a single argument which specifies the half-word address where the first six-bit character to be typed will be found. The TYPNEW subroutine will type all characters which can be generated by the DIAL TEXT pseudo-op with the exception of horizontal tabulate which is ignored and the back slash which is used to terminate typing. The DIAL TEXT code for carriage return is interpreted to give both a carriage return and a line feed. A JMP to the entry point TYPMOR requires no argument and continues typing characters starting at the location where typing was terminated by a back slash during the most recent prior use of TYPNEW or TYPMOR.

PROGRAM NAME: SERINDEX

Program Language: LAP6-DIAL

Computer: PDP-12

Programmer: Peter Gutterman
Laboratory Computer Facility*
University of Wisconsin

PURPOSE

SERINDEX is a subroutine which searches for a given name in a DIAL Index.

USAGE

SERINDEX is a completely self-contained subroutine which must be added to the calling program's manuscript with the Add Program command. The manuscript contains no origin although one may be added. Index registers 1 through 3 are used but not restored.

SERINDEX assumes that the DIAL Index has previously been read into quarters 6 and 7 (locations 3000-3777). The calling sequence is:

```
LDA I
FWA
JMP SINDEX
    returns here if not found
    returns here if found
```

FWA is the first word address of the name to be searched for in the Index. This name should be stored in eight half-words, beginning in the left half of FWA. Trailing unused half-words of the name must contain 77_8 to conform to DIAL Index convention.

If the name is found in the Index, Index register 1 on return will point to the fourth word of the matching name in the Index. The next four words in the Index contain information about the manuscript and binary. The routine assumes the standard Index structure of DIAL. Refer to the document NU INDEX (DECUS No. 12-2) for a description of the DIAL index structure.

---

*SERINDEX was developed by the Laboratory Computer Facility at the University of Wisconsin Medical Center under grant FR00249 from the Division of Research Facilities and Resources of the National Institutes of Health.