DECUS

PROGRAM LIBRARY

DECUS NO.	12-25
TITLE	THREE SUBROUTINES FOR QANDA FRACUS - SCRAMBL - QANDA-C
AUTHOR	R. E. Kingsley
COMPANY	Indiana University Bloomington, Indiana
DATE	February 9, 1971
SOURCE LANGUAGE	DIAL-MS

DECUS Program Library Write-up   DECUS No. 12-25

TITLE: FRACUS

USE

Extracts decimal integer or fraction input from QANDA answer field and places floating point equivalent in user defined output field.

DESCRIPTION

FRACUS extracts the trimmed ASCII coded answers from the QANDA answer fields and converts it to the floating point equivalent of the decimal integer or the decimal fraction input. The output field can be placed anywhere in the LINC memory fields at the discretion of the programmer. Individual entries from a single QANDA display are separated and placed in sequence in the output field. A "?" entry causes a special error return.

MEMORY REQUIREMENTS

The standard version of FRACUS uses LINC memory field 3 (1664_8 locations), and 22 locations on PDP page $\emptyset$.

CALLING SEQUENCE

The QANDA text fields and answer fields and the FRACUS output fields must reside in memory bank $\emptyset$. Before entering FRACUS, the current LINC instruction field must be in the left-most 2 bits of the accumulator. In the output field, the first word will be the floating point exponent; the following two words will be the 23 bit signed mantissa. The floating point output will be compatible with the Floating Point Package and with FOCAL-12.

For this QANDA text:

        GAIN ON CHANNEL 30 _ _ _ _ _

        GAIN ON CHANNEL 31 _ _ _ _ _

the calling sequence would be:

```
        JMP QAINIT          /ENTER QANDA
        TEXT1               /TEXT POINTER
        ANSW1               /ANSWER POINTER
        JMP QARFSH          /REFRESH QANDA
        /
        JMP DSET            /SET UP INST. FIELD IN AC.
        LIF 3
        JMP FRACUS
        ANSW1               /QANDA ANSWER FIELD POINTER
        OUT1                /FRACUS OUTPUT FIELD POINTER
        JMP HELP            /?RETURN POINT: JMP TO USERS SUBROUTINE
        . . . .             /NORMAL RETURN POINT


DSET,   CLR
        IOB
        RIF
        ROL 11
        BCL I
        1777
        JMP Ø
```

the output field would look like:

```
OUT1,   0000                /GAIN CHANNEL 30
        0000
        0000
        0000                /GAIN CHANNEL 31
        0000
        0000
```

Upon an error return, the accumulator will contain a $-\emptyset$ (7777_8) if the error was caused by a "?" entry in the answer field. The accumulator will contain +1 if the error was caused by format other than a "?", e. g. incorrect "E" format (see documentation on the floating point package).

TITLE: SCRMBL

USE

Extracts octal or decimal integer input from QANDA answer fields and places the octal equivalent in user defined output field.

DESCRIPTION

SCRMBL extracts the trimmed ASCII coded answers from the QANDA answer fields and converts it into the octal equivalent of the decimal or octal integer input. The output field can be placed anywhere in the LINC memory fields at the discretion of the programmer. Individual entries in a single QANDA display are separated and placed in sequence in the output field. A "?" entry causes a special error return.

MEMORY REQUIREMENTS

The subroutine uses 102_8 locations in the current instruction field. The supplied version contains no origin statement to facilitate insertion into the users manuscript.

CALLING SEQUENCE

The location tagged "OCTDEC" is used as a switch to indicate whether the input is to be interpretted as octal or decimal. Setting it to any positive value signifies octal input; any negative value, decimal input. For decimal input, four digits are required in the QANDA answer field. Leading zeros are required. For octal input, any number of digits may be used, and leading zeros are not required. The largest decimal value which is a legal input is 4095; the largest octal value 7777. Decimal input converted by SCRMBL is compatible with FOCAL-12.

For this QANDA text:

            BLOCK NUMBER_ _ _

            UNIT NUMBER   _

the calling sequence would be:

```
        JMP QAINIT      /ENTER QANDA
        TEXT1           /TEXT POINTER
        ANSW1           /ANSWER POINTER
        JMP QARFSH      /REFRESH QANDA
        CLR
        NOP             /INSERT "CMA" IF INPUT IS DECIMAL
        STC OCTDEC      /SET SWITCH + FOR OCTAL INPUT
        JMP SCRMBL
        ANSW1           /QANDA ANSWER FIELD POINTER
        OUT1            /SCRMBL OUTPUT FIELD POINTER
        JMP HELP        /?RETURN POINT: JMP TO USERS SUBROUTINE
        . . . .
```

3

the output field would look like:

```
OUT1,     0               /BLOCK NUMBER
          0               /UNIT NUMBER
```

TITLE: QANDA-C

USE

Allows the decoding of the "CONTROL" characters by QANDA.

DESCRIPTION

This subroutine is a modification of QANDA supplied on the standard DIAL systems tape. Its calling sequence is exactly the same as QANDA. Used as supplied, QANDA-C operates exactly as QANDA. However, the user may insert new instructions into the subroutine which will allow calling user subroutines when the "CONTROL" characters are struck.

MEMORY REQUIREMENTS

Two LINC quarters of memory are required, exactly as for QANDA. Locations $1663\text{-}1737_8$ are available to the user.

OPERATION

In the standard version, the look-up table for the "CONTROL" characters is located at *1740. To effect a jump to a user subroutine, insert the appropriate JMP instruction into the appropriate location in the look-up table. For example, to use CTRL/C, insert the JMP instruction at *1743. The only restrictions are as follows:

CTRL/I
CTRL/J
CTRL/M

since these "CONTROL" codes are identical to the regular codes for TAB, LINE FEED, and CARRIAGE RETURN respectively, and must be decoded as such.