**CSCI 360 TR and TRT Instructions**

**Translate**

It may be necessary to replace each value in a field with a corresponding value that is determined by a fixed correspondence.

For example, suppose we want to change all of the lowercase letters in a character string to uppercase. Then:

Original: ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz

Replacement: ABCDEFGHIJKLMNOPQRSTUVWXYZ ABCDEFGHIJKLMNOPQRSTUVWXYZ

In order to accomplish the conversion in ASSEMBLER, a 256 byte table must be constructed. Each byte that should be replaced in the string has its new value put into the table.

TRANTAB DC X'000102030405060708090A0B0C0D0E0F'

DC X'101112131415161718191A1B1C1D1E1F'

...

...

DC X'80***C1C2C3C4C5C6C7C8C9***8A8B8C8D8E8F'

DC X'90***D1D2D3D4D5D6D7D8D9***9A9B9C9D9E9F'

DC X'A0A1***E2E3E4E5E6E7E8E9***AAABACADAEAF'

DC X'B0B1B2B3B4B5B6B7B8B9BABBBCBDBEBF'

...

...

DC X'F0F1F2F3F4F5F6F7F8F9FAFBFCFDFEFF'

If it's known that the string to be converted has only lowercase letters, then the table can be rewritten using the ORG statement:

TRANTAB2 DC 256X'00'

ORG TRANTAB2+C'a'

DC X'C1C2C3C4C5C6C7C8C9'

ORG TRANTAB2+C'j'

DC X'D1D2D3D4D5D6D7D8D9'

ORG TRANTAB2+C's'

DC X'E2E3E4E5E6E7E8E9'

ORG

The instruction to implement this process in ASSEMBLER is:

Format: label TR D1(L,B1),D2(B2)

The L byte field specified by D1(B1) is translated using the translate table specified by D2(B2).

TR STRING(5),TRANTAB2 changes hello to HELLO

\* changes 88 85 93 93 96 to C8 C5 D3 D3 D6

In storage:

STRING DC C'hello' in hex => 8885939396

TRANTAB2 DC 256X'00'

ORG TRANTAB2+C'a'

DC X'C1C2C3C4C5C6C7C8C9'

ORG TRANTAB2+C'j'

DC X'D1D2D3D4D5D6D7D8D9'

ORG TRANTAB2+C's'

DC X'E2E3E4E5E6E7E8E9'

ORG

**Translate and Test**

It may be necessary to locate the first occurrence of a character in a field.

Format: label TRT D1(L,B1),D2(B2)

The L byte field specified by D1(B1) is scanned for the character indicated in the translate table specified by D2(B2).

The TRT instruction does NOT alter the characters in the original field.

Each byte of the designated field is scanned until a character is found that corresponds to a value in the translate table that is not 00.

If no character is found, the condition code is set to 0.

If a character is found, the following occurs:

1. The scanning terminates.
2. The last 3 bytes of register 1 is changed to the address of the detected character.
3. The rightmost byte of register 2 is changed to the value from the translate table.
4. If the character that is found is the last character of the field, the condition code is set to 2; otherwise, it is set to 1.

To find the first non-blank in the field specified by BUFFER,

we could

XREAD BUFFER,80

TRT BUFFER(80),SCANTAB

BZ ALLBLANK

In storage:

BUFFER DS CL80

SCANTAB DC 256X'FF' All of the characters, except for

ORG SCANTAB+C' ' the space, are set to a non-zero

DC X'00' value.

ORG

Result:

If a non-blank is found, then:

Register 1: last three bytes set to address of first non-blank

Register 2: rightmost byte set to X'FF'

Condition code is set to either 1 or 2

Line after the branch instruction is executed

If no non-blank is found then:

Register 1: unaltered

Register 2: unaltered

Condition code is set to 0

Branch to ALLBLANK is taken

To find the first space in the field specified by BUFFER, we could

XREAD BUFFER,80

TRT BUFFER(80),SPACETAB

BZ NOSPACES

In storage:

BUFFER DS CL80

SPACETAB DC 256X'00' All characters, except for the space,

ORG SPACETAB+C' ' are set to zero.

DC X'FF'

ORG

To find the first numeric digit or alphabetic character in the

field specified by BUFFER, we could

SR R2,R2

XREAD BUFFER,80

TRT BUFFER(80),SCANTAB2

B BRTAB(R2)

BRTAB B NONE

B ALPHA

B DIGIT

In storage:

BUFFER DS CL80

SCANTAB2 DC 256X'00'

ORG SCANTAB2+C'A' If alphabetic, set value in

DC 9X'04' register 2 to 04

ORG SCANTAB2+C'J'

DC 9X'04'

ORG SCANTAB2+C'S'

DC 8X'04'

ORG SCANTAB2+C'0' If numeric, set value in

DC 10X'08' register 2 to 08

ORG

Result:

If no alphabetic or numeric character is found, the value in

register 2 remains 0 and the branch to NONE is taken.

If an alphabetic character is found, the rightmost byte of

register 2 is set to 04 and the branch to ALPHA is taken.

If a numeric character is found, the rightmost byte of register 2

is set to 08 and the branch to DIGIT is taken.