



TIPS
UP-7500.1
UP-7500.2

RELEASE

UNIVAC 1005 System Library Memo 6 announces the release and availability of "UNIVAC 1005 Systems TIP 1, Instructions 80/90-Column," UP-7500.1, 3 pages, and "UNIVAC 1005 Systems TIP 2, Assembler Functions 80/90-Column," UP-7500.2, 1 page. These are Standard Library Items (SLI). These TIPS pertain to double address 1005 Systems only.

TIP 1, "Instructions 80/90-Column," UP-7500.1 - This Tip covers a variety of instructions concerning:

Count Circular	Transfer Constant
Edit Mask	Transfers to Bank 1, Row 32
Indirect Addressing	Translate
Unconditional Jump	

Some of this information is new, some corrects existing information, while other items attempt to correct misunderstanding of a subject. It is most important that you review this Tip very thoroughly.

TIP 2, "Assembler Functions 80/90-Column," UP-7500.2 - Supplemental information concerning Assembler Functions is contained in this Tip. The proper functioning of the Assembler can depend upon the information supplied by this Tip.

Tips can be stored in the standard 1005 library binder.

Automatic distribution is being made as indicated below. Distribution is being made to all Customer 1005 mailing lists. Additional copies for Customers should be requisitioned from Holyoke, Massachusetts, via Sales Help Requisition through your local UNIVAC Manager.

Manager
Systems Programming Library Services

TO LISTS: 211 (less 217),
650, 692, and 153,
Library Memo 6 only.

ATTACHMENTS: UP-7500.1, UP-7500.2, and
Library Memo 6 to 10U, 217, 630, and
S.P.L.S. Lists 45, 46, and 47.

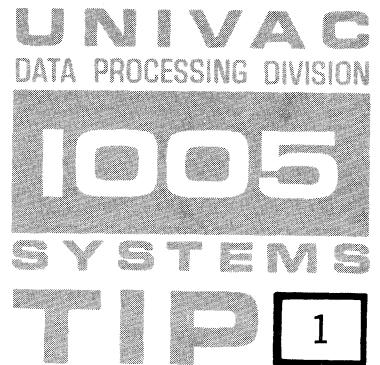
THIS SHEET IS:
UNIVAC 1005 System
Library Memo 6

DATE:
December 23, 1966

C

C

C



Instructions 80/90-Column

This tip contains information about the following UNIVAC 1005 Instructions and subjects related to Instructions:

Count Circular
Edit Mask
Indirect Addressing
Unconditional Jump

Transfer Constant
Transfers to Bank 1, Row 32
Translate

COUNT CIRCULAR (CC)

- A. The CC instruction adjusts the bank bits over the two character address being modified whenever the counting causes a change from one bank to another. This adjustment can have the following effects on the instruction being modified:

Field A

Since the bank bits in Field A correspond to the address designated by Field A, the bank bit modification will produce consistent results.

Field B

In a descending instruction, the bank bits correspond to the address designated by Field B. In an ascending instruction, the bank bits in Field B correspond to the address in Field C. Thus, a CC operation on Field B can cause an unintended change to the starting bank number, even though the starting address of an ascending instruction is located in Field C.

Field C

In an instruction which permits indirect addressing, the X/O bits of Field C are reserved for indirect addressing signals. A CC instruction applied to this field will cause the insertion of X/O bits, if a change in bank occurs. When this instruction is executed, the extraneous bit(s) will be interpreted as indirect addressing signals, producing incorrect results.

These bits can be removed from Field C by following the CC instruction with an Edit Erase instruction.

- B. In Field A (DD), space is not equal to zero. When the value of DD is less than ten (10), a zero must be entered into column 12.
Example: 07 not ♂7.

EDIT MASK

A transfer of the Edit Mask to the X Register must precede each Edit (Mask) instruction because the mask in the X Register is replaced by the data characters whose MSL address is specified in Field A of the ED instruction. The conventions for Editing should be; (1) Load the mask to X Register, (2) Edit data via the mask to the output field.

The ED instruction is a two phase instruction of; (1) Editing data to the X Register under control of the mask, and (2) transferring the results stored in the X Register to the output field as specified by Operand 2 of the ED instruction (Field B and Field C).

INDIRECT ADDRESSING

In most UNIVAC 1005 instructions, only one set of bank bits is stored for each operand. These bank bits, located in fields A and B, reflect the banks in which the execution of the instruction is to begin. The Assembler inserts these bank bits according to the mode of the instruction. For ascending transfers, LSL bank bits are inserted; for descending transfers, MSL bank bits are inserted.

However, when an instruction referring to a field that is to be addressed indirectly is assembled, the Assembler must insert MSL bank bits. Descending instructions, and instructions in which both the MSL and LSL of a working storage area are in the same bank, will be executed as desired. An ascending instruction in which a working storage area is both addressed indirectly and split across two memory banks will not operate correctly, because the bank bits will reflect the MSL bank instead of the LSL bank.

This problem is encountered when an arithmetic operation is performed on one of a series of accumulators if the accumulator is selected by indirect addressing and if one of the accumulators is split between two memory banks.

This problem can be circumvented by assigning memory in a way that will avoid the splitting of fields across memory banks.

A good rule to follow is to code all DA instructions first, so that they will be restricted to the last bank of memory.

UNCONDITIONAL JUMP

The mnemonic operation code for the Unconditional Jump instruction is J, not JU.

TRANSFER CONSTANT (TK)

As indicated in the UNIVAC 1005 Programmers Reference Manual, the TK instruction may be used to substitute addresses in another instruction. This transfer of addresses can be accomplished by entering the label of the address to be transferred in Field A of the TK instruction. The label in Field A must be preceded by a colon (:).

When this technique is used, caution must be exercised when the Field A or Field B address of one instruction is being transferred to Field C of another instruction. Since Fields A and B normally contain bank bits, the Field C will receive bank bits which may not be desired. These bits will be interpreted as Indirect Addressing signals on many instructions and will cause erroneous results.

These bits can be removed from Field C by following the TK instruction with an Edit Erase instruction.

TRANSFERS TO BANK 1, ROW 32

Transfers to Bank 1, Row 32 must be made in an ascending mode.

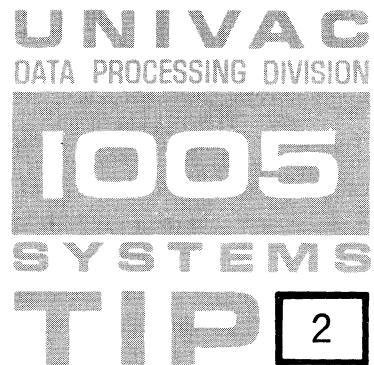
TRANSLATE (Optional Feature)

The data to be translated must be located in the same memory bank as the translate table.

O

O

C



Assembler Functions

80/90-Column

Supplemental information concerning the following Assembler Functions is contained in this UNIVAC 1005 Tip:

Definition of Work Storage Areas
Instruction Location Counter Addressing
Restarting an Assembly

DEFINITION OF WORK STORAGE AREAS

Work storage areas are assigned in an ascending mode. Thus, if ten accumulators are defined with accumulator 1 entered first and accumulator 10 last, accumulator 10 will be the most significant accumulator.

INSTRUCTION LOCATION COUNTER ADDRESSING

The coding to use the address of the next instruction as an operand should be: \$ + 7 or \$ + 5, not \$ + 1.

RESTARTING AN ASSEMBLY

An assembly may be restarted at either pass 2 or 3 by placing one of the cards described below in front of the Assembly Load Card and depressing the Clear, Start, Feed and Run.

Card Columns

	1	2	3	4	5	6	7	8
80-Restart at Pass 2	2	2	2	1	Ø	1	Ø	V
Restart at Pass 3	2	3	3	1	Ø	1	Ø	V
90-Restart at Pass 2	6	2	2	5	8	5	8	?
Restart at Pass 3	6	3	3	5	8	5	8	?

C

C

C