MEMORANDUM

TO: TX-0 Computer Users

FROM: J. B. Dennis

SUBJECT: Proposed Revision of the TX-0 Operate Command

DATE: 6 August 1959

In the two charts attached to this note is a proposal which would considerably increase the usefulness of the TX-0 operate class commands with a relatively small investment of time and material required to put it into effect. The proposal would affect currently operating programs for the TX-0 only in that five commands would be assigned new operation codes. These commands are TAC, TBR, PEN, "AND" and "OR". In all other respects currently operating programs would be unaffected.

Eight new micro commands have been added; see the chart for their mnemonic symbols and what they do.

The command MAG forms the magnitude of the contents of AC and is very useful in testing for zero. The operate combination amb, com, cry is equivalent to adding minus zero to the AC. Therefore, the program sequence

$$mag,amb,com,cry*$$

$$trn \longrightarrow$$

will transfer if and only if AC originally contained plus zero or minus zero.

If the commands mlr and lmb are given together the contents of LR and MBR would be interchanged (this is presently true only if MBR contains zero). Therefore, the combination

$$aba,mlr,lmb,pad$$

will exchange the contents of AC and LR. A number of variations of this are possible:

$$abl,mlr,lmb,pad,cry$$

puts (AC) in LR and (LR)+1 in AC;

$$aba,mlr,lmb,cry$$

puts (AC) in LR and (LR) in AC cycled left;

$$abl,mlr,lmb,com,pad,cry$$

puts (AC) in LR and (LR)-1 in AC;

---

* Operate commands are written in this form here for ease of understanding the examples. Of course, (unfortunately), this form cannot be converted by MACRO.

aba,mlr,lmb,com,pad

puts (AC) in LR and -(LR) in AC;

etc.

The micro operation ANB and ORB allow the user to form a logical sum or product in AC without affecting LR.

Long right shift or cycle, LRS and LRC would be necessary features when and if an automatic multiply is built into the TX-0, and would be useful by themselves for unpacking words and for scaling arithmetic quantities.

The change sign command is frequently convenient when synthesizing complex arithmetic commands.

In the following program examples the features of the index register proposal (M-5001-9) will also be used.

<u>Example 1</u>   Sum number of ones in 18-bit word.

    define

       sumones

       tsx son

       terminate

son, stx so4

     adx (-22+1

sol, trn so3

     aba,mlr,lmb,pad

     tra so2

so3, abl,mlr,lmb,pad,cry

so2, aba,mlr,lmb,cry

     tix sol

     lac

so4, tra

<u>Example 2</u>   Integer multiply subroutine.*

    define

       multiply X

       tsx mul

       add X

       terminate

---

*   This routine will multiply positive or negative integers as long as the magnitude of the product is $\leq 2^{17}-1$.

```
        mul, llx 0
              tix mul
        mul, stx mu5
              slr mn2
              lro
              adx (-22+1
        mb0, trn mu4
              aba,mbl,lmb,cry
              tra mu3
        mu4, aba,mbl,lmb,cry
        mu2, XX
        mu3, aba,mbl,lmb,cry
              tix mb0
              cla,lmb,pab
        mu5, tra
```

<u>Example 3</u>   Multiply subroutine for positive numbers retaining
             34 bits. The minor half of product is left in AC,
             the major half in LR.

```
             define
                  multiply X
                  llr X
                  tsx mul
                  terminate



        mul, stx mu5
              slr t

              adx (-21+1
              amb,mbl,pad
              lrc


        mul, trn mu2
              lrc
              tix mul
```

```
        tra mu3
mu2,  chs,lrc
        add t
        tix mul
mu3,  lrc
mu4,  chs
        trn mu4
        aha,mbl,lmb,pad
        cyr
mu5,  tra
t,    0
```

Signed *J. Dennis*

J. B. Dennis

JBD/dbh

att. (2)

MEMORY ADDRESS REGISTER BITS

TP	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
0.8	‡ 0	1 CLR														
	‡ 1	0 CLL														
	‡ 1	1 CLA														
	₱ 1	0 * CHS														
	₱ 1	1 * MAG														
IN-OUT STOP			2	IN-OUT ORDERS SEE BELOW.											

MAR bits 6,7,8

MAR bits 4,5		0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
	00	2	2	EXTERNAL USERS' EQUIPMENT FRC 2	2	2	2	2	2
	01	‡ 2	₱ 2	● PEN 2 1.1	● TAC 2 1.1	● THR 2 1.2	□ MUL 2	2	2	2
	10	I-O SEL 2	RIL 2	DIS 2	R3L 2	PRT 2	2	P6H 2	P7H 2
	11	HLT 2	2	EXTERNAL USERS' EQUIPMENT 2	2	2	2	2	2

Assignment of input-output orders.

□ Possible operation code for a multiply order.
‡ These commands selected if MAR 45678 ≠ 01000.
₱ These commands selected if MAR 45678 = 01000.
* New commands – see second sheet for meaning.
● Altered commands.

NOTE: Commands with same index in lower left corner are decoded together and cannot be used simultaneously.

PROPOSED ALTERATIONS TO OPERATE CLASS COMMANDS - CYCLE ONE

MEMORY ADDRESS REGISTER BITS

TP	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1.2	0 0** / 1 ABA													0 0 1 / 4 ANB		
	0 1* / 1 AB1													0 1 1* / 4 IMB		
1.3								0 1 1* / 3 ANL			1 / COM			1 1 0* / 4 ANB		
								0 1 0 / 3 MIR						0 1 0 / 4 IMB		
								0 0 1° / 3 ORL						1 0 0* / 4 ORB		
1.4								1 0 1+ / 3 MRX				1 / PAD				
1.5								1 0 0° / 3 SHR								
								1 1 0° / 3 CYR								
1.6												1 / CRY				
1.7								1 1 1+ / 3 MPC								
								1 0 0* / 3,4 LRS						1 0 1		
								1 1 0* / 3,4 LRC						1 0 1		

+ Commands proposed in connection with the proposed index register.
  - XMB   transfer XR to MBR_{5-17}.
  - MRX   transfer MBR_{5-17} to XR.
  - MPC   transfer MBR_{5-17} into PC.

Explanation of new commands:
  - CHS   complement bit zero of AC.
  - MAG   complement AC if negative.
  - ABA   transfer AC to MBR and clear AC.
  - AB1   transfer AC to MBR and set AC to 1.
  - ANB   "And" contents of LR into MBR.
  - ORB   "OR" contents of LR into MBR (inclusive).
  - LRC   Long right cycle - cycle AC and LR right. $\begin{cases} AC_{17} \rightarrow LR_0 \\ LR_{17} \rightarrow AC_0 \end{cases}$
  - LRS   Long right shift - same as LRC but AC_0 is unchanged.
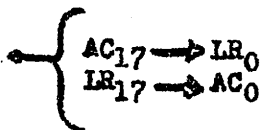
Altered commands:
  - AHL   "And" MBR into LR. ⎱ New codes as above.
  - ORL   "Or" MBR into LR. ⎰
  - TAC, TBR, PEN - recoded as variations of in-out commands.
  - SHR, CYR - occur on time pulse 1.5 and may be used with PAD.

MEMORANDUM

TO: TX-0 Computer Users and Other Interested Persons

FROM: J. B. Dennis

SUBJECT: Proposals for Additions to the TX-0 Order Code

DATE: 6 August 1959

In the past year the TX-0 has proved its value as a teaching aid and research tool. So that it might better serve its users a number of changes and additions have already been made to the logic of the machine. These include the addition of the three new instruction codes, LLR, SLR and TRA, "logical product" and "inclusive or" variations of the operate command, and the doubling of the core memory currently being installed.

During the coming year there will be further opportunities for increasing the flexibility of the TX-0 for the research and pedagogical needs of its users. To make the most of these opportunities, your interest and suggestions are earnestly solicited.

Enclosed are copies of two proposals for modifications of the TX-0 order code. The first is a scheme for adding an index register which would allow automatic address modification for the LLR and SLR instructions, would permit counting and testing to be performed without disturbing the AC or LR, and would allow a single instruction calling sequence for sub- routines. This proposal would involve the addition of a new 14-bit register to the arithmetic element of the machine.

The second proposal is designed to increase the usefulness of the operate class commands. The proposal would change the MAR bit patterns defining TAC, TBR, PEN, "AND" and "OR", but would not otherwise alter the present operation of the computer. It seems that the advantages of the proposal outweigh the possible confusion resulting from the change.

Other proposals which will be circulated in the near future concern a deferred addressing scheme, and a multiply command. A command which will test and/or change a specified single bit of the accumulator or live register is a possibility which would not involve a great amount of additional equipment.

Again I ask your consideration of these proposals since the users of the machine will be the ones to profit,-- or suffer if the most useful possibilities are not discovered. Any suggestions or criticisms will be most welcome.

Signed _J. Dennis_
J. B. Dennis

JBD/dbh