IBM Customer Engineering
Manual of Instruction

J. H. McCarthy

1401 Data Processing System
Optional Features

IBM®

Customer Engineering
Manual of Instruction

J.H. McCarthy

R.H. NELSON

Issued to: _____

Branch Office: _____

Department: _____

Address: _____

If this manual is mislaid, please return it to the above address.

1401 Data Processing System

Optional Features

This manual is written to cover IBM 1401 systems with serial numbers above 1401-20000. At this point, a major revision was made in the system logics and hardware placement. A major portion of the philosophy of operation remains unchanged. Except for logic and test-point references this manual can serve for earlier systems. Retain copies of previous IBM 1401 Manuals of Instruction for reference.

Contents

The IBM 1401 Optional Features Instruction Manual provides instructional material on the additional features that have been released for the 1401 basic card system. It is assumed that the reader is familiar with the data flow and controls of the basic system. With the addition of some of the features, the basic card system may lose its identity. The 1401 processing unit with its core storage, however, remains as the center of all systems. It becomes possible to have a system with no card input or output. Many of the features require the use of the second frame for the processing unit (Model B, C, etc.).

The major of these features may be considered the tape configuration for the IBM 1401. In addition to its controls, it adds up to six tape units, which may be individually operated. Its addition to the basic system offers several new operating possibilities. These include the updating of tape records, and the transfer of data between the card equipment and magnetic tape. Associated with the tape controls though not a part of the same feature is the serial input/output adapter. This feature allows the use of many different devices for serial data transfer (one device at a time). These range from slow key-operated devices to other data processing units. The addition of one or both of these features allows the 1401 to serve as an off-line processor, an auxiliary to a larger system, or as a data collection device.

The remainder of the features provide for improved program flexibility, increased processing speed, or specific operating arrangements. These features include expanded storage, address indexing, direct calculating and several new operation codes in the processing unit, and several options for the input/output areas. The function and operating features of each are discussed in their individual sections.

The features are grouped by type of operation. The next four sections contain groups dealing with, respectively, storage, programming, data handling, and arithmetic operation. The last three sections cover groups representing magnetic tape, serial input/output, and card input/output. In a few cases an odd operation is seemingly out of place. This is due to coupling of items for sales benefit.

The storage capacity of the IBM 1401 can be expanded by the addition of an IBM 1406 Core Storage Unit. The 1406 is supplied in three models to store four, eight, or twelve thousand characters. These provide the 1401 with a total of eight, twelve, or sixteen thousand storage positions, respectively. Each position is capable of storing the full eight-bit 1401 character. The 1406 core arrays and controls are mounted in a standard sized single cube module arranged with sliding-type gates. The unit stands free of the 1401 but is limited in its placement because of cable loading (10 foot cable). Any character position may be addressed with the slightly modified address system.The basic *inhibit drive* is used to drive all areas. The storage read out from all areas is mixed to set the 1401 B-register.

IBM 1406 Layout

The 1406 gates are arranged in pairs to accommodate the core arrays. Gates 1 and 2 contain the 8K and/or 12K arrays along with their drivers and sense amplifiers. Gate 4 contains the control logic associated with the 1406. Gate 3 area is blank. Gates 5 and 6 contain the 16K core array with its drivers and sense amplifiers. The Gate 7 area is taken up with the cable connectors. One group of ten paddle connectors, and two power connectors, are used to connect the 1406 to the 1401. The Gate 8 area is used for power control, a +12V supply, two 18V differential supplies, and a +30 volt power supply. Other power supply voltages are obtained from the 1401 (or 1402). No sliding gate is used in position 7 and 8 area. The connector panel and the power panel are fastened directly to the frame. The connectors are accessible through the rear gate cover, while the power supply access is through the side cover.

Core Arrays

Two core arrays are used in the 1406. One is a 4K array and the other is an 8K. In appearance, both arrays resemble that used in the 1401. They differ in the core planes and connections. The basic 4K array has eight planes for data storage and eight planes for the input/output buffer, I-O checking, and terminals. The 4K array used in the 1406 has the same eight planes for data storage but the remaining eight frames are omitted. The 8K array has two sets of data planes stacked one above the other.

The 4K array is used both for the 4K Model 1 and the 12K Model 3. In the former, it is mounted in the Gate 1 and 2 position for the 4000 to 7999 storage addresses. In the latter, it is mounted in the Gate 5 and 6 position for the 12000 to 15999 storage addresses. The drive arrangements are identical to that of the basic 4K array except for the lines used in the I-O buffer area. The 4000 cores on a plane are arranged in a 50 by 80 block. The 50 side is fed by a 5 by 10 drive matrix and the 80 side by an 8 by 10 drive matrix. The decode to the matrix drivers is the same for each of the 4K groups because the address only changes in the units zone for the expanded areas. The 5 by 10 matrix is driven by the decode of the even-hundreds and the units positions. The 8 by 10 matrix is driven by the hundreds zone and the odd-hundreds against the ten digit. The inhibit drive is split at two thousand as is the basic array, to reduce noise. The sense amplifiers are mixed with the other arrays to set the 1401 B-register.

The 8K array is used to supply the 4000 to 11999 storage addresses for both Models 2 and 3. It is mounted in the Gate 1 and 2 position in either case. The drive arrangements are similar to the 4K but expanded at one point to double the capacity. The two 4K blocks, while mounted one above the other, are connected as if they lay side-by-side. Electrically, the array appears as an 80 by 100 block. The 8 by 10 drive matrix feeds both sections in series. As a result of this expansion, the 5 by 10 matrix is expanded to a 10 by 10 arrangement. Actually the 10 by 10 can be considered as two 5 by 10 matrices side-by-side with a single units-decode driving the two in series. Two separate even-hundreds decode networks are used to drive the remaining side. The proper decode network is selected by switching with the 8K or 12K activate storage signal. Since one half drives the 4 to 8K area, while the other drives the 8 to 12K area, the proper address is selected.

The two 4K areas have separate sense amplifiers to obtain a better signal to switching noise ratio. The output of these, and the sense amplifiers of other arrays, are mixed to set the 1401 B-register. The inhibit drivers are duplicated for over two thousand to reduce sense noise. Because the two halves of the array have separate sense amplifiers, the two sections can be driven in series without a noise problem. The combination of the two areas allows a considerable saving in drivers. The selection of the 4K area is made by gating the activate storage signal with the units zone bits of the address.

Controls (Figure 1)

The 1406 operates as a slave unit to the 1401 system. It has no controls for independent operation. The control area results from the need to select the proper 4K

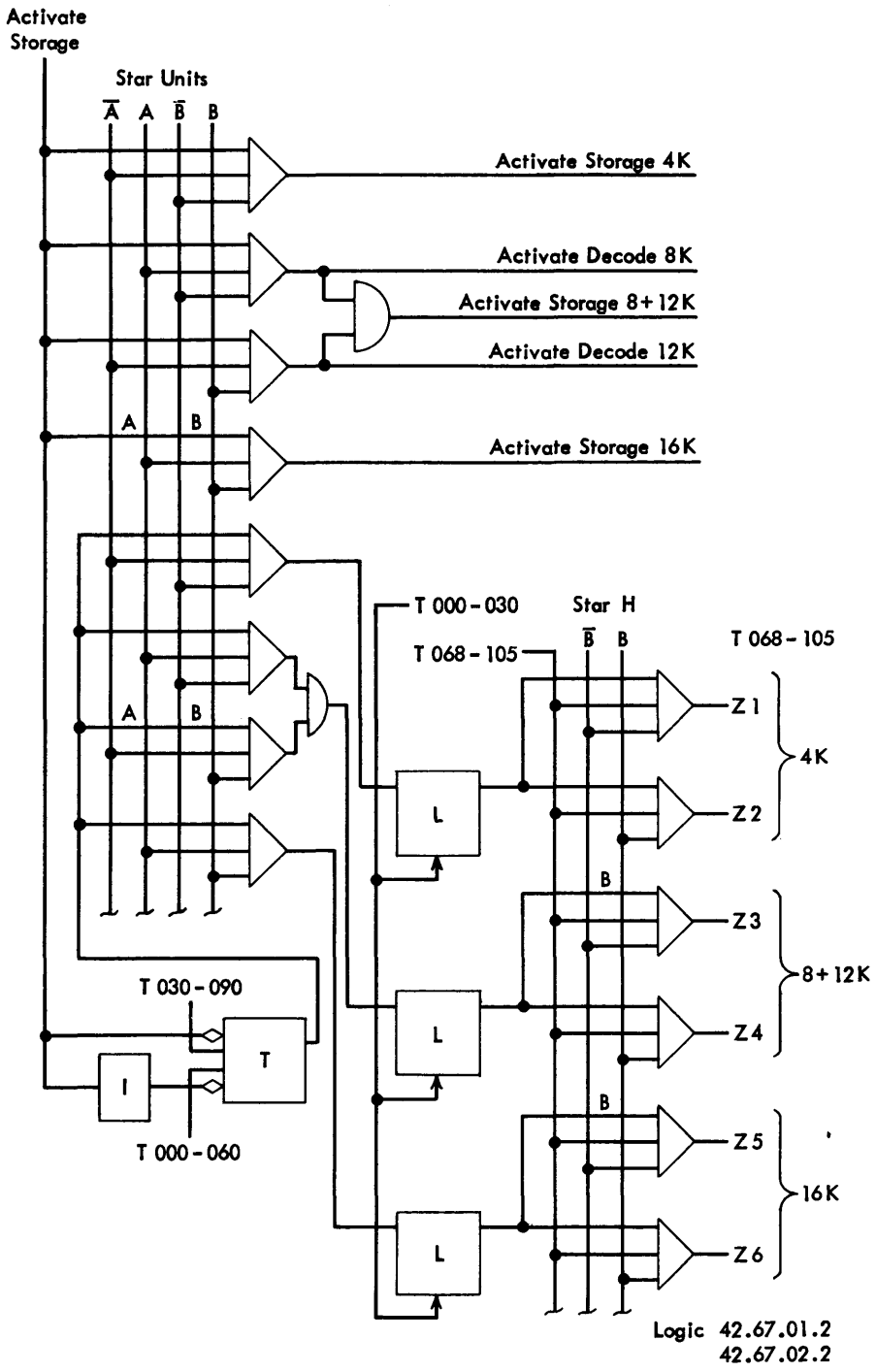


Figure 1. 1406 Storage Controls

area with the expanded address requirements. The activate storage signal is used in the basic system to control storage operation. As a result of this, it makes the most convenient point to select the 4K area. The activate storage for the basic 4K storage in the 1401 is first gated with the units no-zone before it can be used to select that area. Separate activate storage signals are developed for each array. The larger 8K array uses the separate signals to gate the even-hundreds decode, while the two signals are mixed to control the remainder of the drivers. Without the 1406, the activate storage line connects directly to the basic 4K controls. When the 1406 is added, the activate storage signal is sent to the 1406 controls. The gated return line drives the 1401 controls. The change requires the removal of one card and the addition of another.

On the expanded storage system, the inhibit drivers are gated for each core array. This switching is performed in the 1406 and a gate is returned to the 1401 to control the basic array. One card is added to the basic switching, which serves to block the basic inhibit drivers when an expanded area is used. In each case, the developed gate is further switched with the address register hundreds B-bit (2 thousand) and a clock pulse. The 2-thousand gate selects between two inhibit drivers for either the first two thousand characters or the second two thousand characters. Duplication of inhibit drivers is required for noise reduction the same as in the 1401 4K array.

Two new operation features are added for use with the expanded storage attachment. The special controls for these features are located on Gate 4 of the 1406. The first feature is an *address modify operation code* provided to facilitate adjusting a three-digit address by addition. The basic add operation does not properly handle the carry problem with units position zone being high order. The second feautre is a variation of the manual storage print from the setting of the console mode switch. The setting of an additional control switch on the auxiliary console provides for *full storage print.* Instead of printing a single line from a specified address block, it causes the full available storage to print out. Further details of these features are covered under *Op Modify Address* and *Full Storage Print.*

Addressing

Addressing for the full sixteen thousand storage positions can be designated by the three character address used in the basic system. The thousands position for up to four thousand made use of the A- and B-zone bits over the hundreds position. The added twelve thousand addresses make use of the A- and B-bits over the units position to expand the thousands position. By assigning binary weight to the four zone bits, any address can be directly decoded:

HA = 1	UA = 4
HB = 2	UB = 8

An address including all four zone bits has a numerical value of 15 (thousand). This added to the possible 999 digit value provides a total space of addresses from 000 to 15,999. The combination of digit and zone bits appear as alpha characters in the address.

Examples:

$$\text{Address M3X} = \frac{\text{B} - \text{A}}{4\ 3\ 7} = \frac{2 - 4}{4\ 3\ 7} = 6{,}437$$

$$\text{Address W6D} = \frac{\overset{\text{A}}{\text{A} - \text{B}}}{6\ 6\ 4} = \frac{\overset{4}{1 - 8}}{6\ 6\ 4} = 13{,}664$$

Expanded Address Controls

The expanded addressing requires changes in several areas to process the units position zone bits. The major of these areas is the address register system. Each of the registers requires latches for the A, B, and CZ bits (B-bit is not required for 8K). Circuitwise the added bit latches are identical to those of the basic register. Functionally, though located in the units position, they represent the high-order thousands (4 through 15). For this reason separate controls for the read-in and read-out are needed when serially processing.

Components used for address modification require additional circuits to handle the units zone. The address modifier, used in routine address up-dating, is changed to proces the units zone simultaneously with the hundreds position. This requires analyzation of tens carry, hundreds digit and zone, and the units zone. These circuits vary with the storage size to effect the wrap-around or return-to-zero at the end of the available area. The modification requires no increase in time.

In effect, the zone-adder is changed to sense zone-carry. These circuits are physically located in the 1406 instead of in the adder area. Controls for the use of these circuits are part of address modify operations. A normal address increment is made in three cycles. If a hundreds zone carry occurs, an extra cycle is taken to increase the units zone value.

The addition of the 1406 to a 1401 system, or the change of one 1406 model to another, involves the addition and removal of circuit cards. The MFI coding on the individual logic blocks identifies which circuits are in use. Notes at the bottom of the drawing are used to augment the information. Usually a block that is added for one size stays for the larger sizes, but not in all cases.

OP Modify Address

An additional operation code is provided with the expanded storage feature. The new *modify address code* allows correct handling of the zone bits which designate the thousands and ten-thousands position. The basic add operation is unsuitable for the modify operation because of the extra zone bits and control requirements. The modify operation uses a special sequence control without reference to word-marks in either field. The three digit field starting at the specified A-address is added to the three digit field starting at the specified B-address. The result is stored in the B-field. The operation may be performed with only one address (A-address reads into both A-Star and B-Star). In this case, the A-field is doubled and the resut is stored in the A-field.

Data Flow

The modify operation is always performed as true-add. To decrease the address, it is necessary to use the complement value to 16,000. The extra B-cycle used in the add operation for the analysis of signs is not required. Both the digit-bits and the zone-bits of the units position are added. The carry from the digit bits is not added to the zone but to the tens digit. A carry from the zone bits serves no purpose. It indicates an advance beyond 16,000 into the low-order area as a wrap-around. The tens digit bits only are added in the normal manner. Both the digit and zone bits are added for the hundreds position. A digit carry adds to the zone result. The zone carry indicates an advance to the next 4K storage area. The units position result is brought back in the case of a carry. An A-bit forced into the A-register adds to the zone bits to serve as a carry.

Cycle control for the operation is provided by a three-position binary-connected trigger ocunter. The full operation requires a total of eight cycles. If no zone carry occurs in the hundreds position, the operation is stopped after six cycles. These six cycles are alternating A- and B-cycles for the add operation. The last two cycles, when taken, are B-cycles. The first of these adjusts the address back to the units position, and the second adds the zone. Word-marks serve no purpose in the control. If word-marks are present, they are retained but not transferred (A-field to B-field).

Addressing for the three digit add uses the normal −1 modifier. At the end of three B-cycles, the B-Star stands at a value one less than the high-order position. If no zone carry occurs, the operation ends at this point. The return of the address to the units position requires a dummy B-cycle. The B-Star is read out to the storage address register and modified by +3 to reach the units position. During the cycle, the addressed position reads out to the B-register, and is regenerated. The A-register is reset and an A-bit is forced in to serve as the zone carry. On the sceond B-cycle, the units position is read out to the B-register, is added to the A-bit in the A-register, and is returned to storage.

Control Circuits (Timing Chart — Figure 2)

The address modify operation makes use of the basic cycle control of the 1401. These controls, in addition to cycling between instruction and execute operations, provide for the transfer of the A- and B-fields to their respective registers. The difference, as in many operations, lies in the control of the inhibit drive to return a character to storage on the B-cycle. The address modify operation does not stop on word-marks, but makes use of its own cycle ring to control transfer and operating conditions of the eight cycles. The first six cycles appear as the alternate A- and B-cycles of a three digit add operation. The remaining two cycles are used for entering carry into the units zone. A-cycle eliminate is forced to make the last two cycles both B-cycles. The operation ends with a forced I-E Change after six calls, if no zone carry is developed for the hundreds position. Continuation of the cycle effects the zone carry entry and forces the I-E Change at the end of the eighth cycle.

Objectives (Op Modify Address)

1. Setup
   - A. Op Decode
   - B. I-E Change
   - C. Set A-Cycle
2. Units Position Add
   - A. RI A-field Units to A-reg
   - B. Set Delta B-cycle
   - C. Set Cycle Trigger 1
   - D. RI B-field Units to B-reg
   - E. Mod Op Block NU Carry
   - F. Mod Op Not Adder Carry
   - G. Arithmetic Zone Inhibit
   - H. Arithmetic Digit Inhibit
   - I. Set A-cycle
3. Tens Position Add
   - A. Reset Cycle Trigger 1
   - B. Set Cycle Trigger 2
   - C. RI A-field Tens to A-reg
   - D. Set Delta B-cycle
   - E. Set Cycle Trigger 1
   - F. RI B-field Tens to B-reg
   - G. B-reg Zone Inhibit
   - H. Arithmetic Digit Inhibit
   - I. Set A-cycle

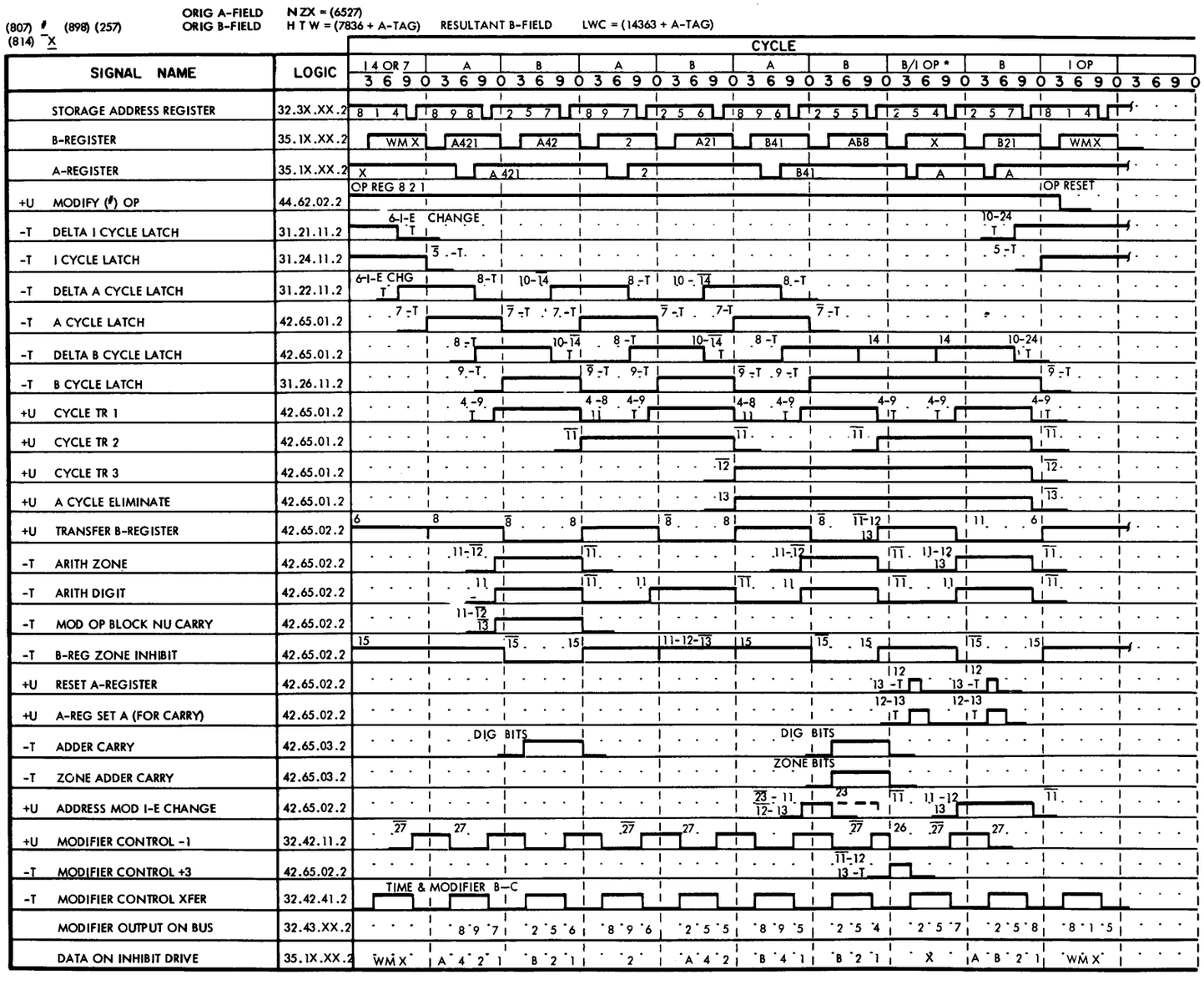


Figure 2. Modify Address #(AAA)(BBB)

4. Hundreds Position Add
   A. Reset Cycle Trigger 1
   B. Reset Cycle Trigger 2
   C. Set Cycle Trigger 3
   D. A-cycle Eliminate
   E. RI A-field Hundreds to A-reg
   F. Set Delta B-cycle
   G. Set Cycle Trigger 1
   H. RI B-field Hundreds to B-reg
   I. Arithmetic Zone Inhibit
   J. Arithmetic Digit Inhibit
5. Zone Carry Controls
   A. Zone Adder Carry
      (If no adder zone carry occurs, the operation skips from here to 7B Force I-E Change.)
   B. Reset Cycle Trigger 1
   C. Set Cycle Trigger 2
   D. B-cycle
   E. Modifier Plus 3 (Units)
   F. Modifier Plus 1 (T & H)
   G. RI B-field Hundreds −1 to B-reg
   H. Transfer B-register
   I. Reset A-reg
   J. A-reg Set A (bit)
6. Units Position Zone Carry Entry
   A. B-cycle
   B. Set cycle Tr 1
   C. RI B-field Units to B-reg
   D. Arithmetic Zone Inhibit
   E. Arithmetic Digit Inhibit
7. Force I-E Change
   A. Addr Mod I-E Change (carry)
   B. Addr Mod I-E Change (no carry)

Logic Detail (Op Modify Address)

1. SETUP

1A. Op Decode: The operation gate serves in two major areas: to prevent sensing word-marks and to gate the modify ring.

44.62.02.2	Modify Op	Op Decode Not AB
4B		Op Decode 821 Not 4

1B. I-E Change: The normal I-E change is used to end the instruction cycle and to start the execution.

1C. A-Cycle: An A-cycle is performed with the normal cycle gating. The Op modify ring does not start until near the end of the cycle.

2. UNITS POSITION ADD

2A. RI A-field Units to A-reg: A normal A-cycle reads the A-field from core storage to the B-reg and regenerates. The character transfers from the B-reg to the A-reg, where it remains for the next cycle.

2B. Set Delta B-Cycle: At the normal 075 time, the Delta A-cycle is reset and the Delta B-cycle set by the basic circuits.

2C. Set Cycle Trigger 1:

42.65.02.2	Cycle Tr 1	Delta B-Cycle
2B		Modify Op
		Time 105-000

2D. RI B-field Units to B-reg: A normal B-cycle control reads out the storage location to the B-reg without regeneration.

2E. Modify Op Block NU Carry: This signal blocks the use of digit adder carry in the zone adder. The carry trigger is still set for the tens cycle.

42.65.02.2	Modify Op Block	Cycle Tr 1
2A	NU Carry	Not Cycle Tr 2
		Not Cycle Tr 3

2F. Modifier Op Not Adder Carry: Without the adder carry signal available, the zone adder requires a Not Adder Carry to gate.

34.31.15.2	Not Adder Carry	Mod Op Not Adder Carry
(4B)		

2G. Arithmetic Zone Inhibit: Gates zone adder inhibit drive to return the modified zone to storage.

42.65.02.2	Arith Zone	Cycle Tr 1
3G		Not Cycle Tr 2

2H. Arithmetic Digit Inhibit: Gates digit adder inhibit drive to return the modified digit to storage.

42.65.02.2	Arith Digit	Cycle Tr 1
3H		

2I. Set A-Cycle: The set of the Delta A-cycle and reset of Delta B-cycle occurs in the normal manner to read the next character.

3. TENS POSITION ADD

3A. Reset Cycle Trigger 1:

42.65.01.2	Not Cycle Tr 1	Cycle Tr 1
2A		A Cycle
		Modify Op

3B. Set Cycle Trigger 2:

42.65.01.2	Cycle Tr 2	Not Cycle Tr 1
3D		Not Cycle Tr 2

3C. RI A-field Tens to A-reg: Operation is a normal A-cycle storage read-in.

3D. Set Delta B-Cycle: Same as step 2 B.

3E. Set Cycle Trigger 1:

42.65.02.2	Cycle Tr 1	Delta B-cycle
2B		Modify Op
		Time 105-000

3F. RI B-field Tens to B-reg: Same as step 2D.

3G. B-reg Zone Inhibit: The zone adder is not used for the tens position. The B-field zone is regenerated to save a possible index tag.

42.65.02.2 5C	B-reg Zone Inhibit	Cycle Tr 1 Cycle Tr 2 Not Cycle Tr 3

3H. Arithmetic Digit Inhibit: Same as step 2B.

3I. Set A-cycle: Same as step 2I.

4. HUNDREDS POSITION ADD

4A. Reset cycle Trigger 1: Same as step 3A.

4B. Reset Cycle Trigger 2:

42.65.01.2 3D	Not Cycle Tr 2	Not Cycle Tr 1 Cycle Tr 2

4C. Set Cycle Trigger 3:

42.65.01.2 3F	Cycle Tr 3	Not Cycle Tr 2 Not Cycle Tr 3

4D. A-cycle Eliminate: The A-cycle Eliminate does not affect the cycle in progress. The signal remains active to the end of the operation to prevent normal A-cycle controls.

42.65.01.2 4H	Address Mod A-cycle Elim	Cycle Tr 3

4E. RI A-field Hundreds to A-reg: Same as step 2A.

4F. Set Delta B-cycle: Same as for step 2B.

4G. Set Cycle Trigger 1: Same as for step 3E.

4H. RI B-field Hundreds to B-reg: Same as for step 2D.

4I. Arithmetic Zone Inhibit: Same as for step 2G.

4J. Arithmetic Digit Inhibit: Same as for step 2H.

5. ZONE CARRY CONTROLS

5A. Zone Adder carry: Several combinations of zone bits can produce an overflow or carry condition.

42.65.03.2	Zone Adder Carry	A-reg Zone bit B-reg Zone bit Digit Adder carry

5B. Reset Cycle Trigger 1: Same as for step 3A.

5C. Set Cycle Trigger 2: Same as for stop 3B.

5D. B-Cycle: With an A-cycle Eliminate gate, the Delta B-cycle latch is not reset, which causes the successive cycles to be B-cycles. The normal controls are used for this cycling.

5E. Modifier Plus 3:

42.65.02.2 4F-5F	Modifier Plus 3	Not Cycle Tr 1 Cycle Tr 2 Cycle Tr 3 Time 000-030

5F. Modifier Plus 1: A plus three modifier is used for the units digit only. If a carry is developed, the plus one modifier is used to effect the carry circuits.

42.65.02.2 4F	Address Mod Ctrl	Not Cycle Tr 1 Cycle Tr 2 Cycle Tr 3
42.62.01.2	Address Mod Plus 1	Address Mod Ctrl Time 030-090

5G. RI B-field Hundreds − 1 to B-reg: This is a dummy cycle to allow the plus three modification of the B-address. The character read-out has no bearing on the operation.

5H. Transfer B-register: This signal forces a regeneration of the character just read out of core storage.

42.65.02.2 4F	Transfer B-reg	Not Cycle Tr 1 Cycle Tr 2 Cycle Tr 3

5I. Reset A-register:

42.65.02.2 3D-4D	Reset A-reg	Cycle Tr 2 Cycle Tr 3 Time 030-045

5J. A-reg Set A (bit): The A-register is cleared and set both in this cycle and the next for the zone carry entry. It is not used in this cycle.

42.65.02.2 3D-4E	A-reg Set A	Cycle Tr 2 Cycle Tr 3 Time 030-060

6. UNITS POSITION ZONE CARRY ENTRY

6A. Set Cycle trigger 1: Same as for step 3E.

6B. B-Cycle: Same as for step 5D.

6C. RI B-field Units to B-reg: Same as for step 2D.

6D. Arithmetic Zone Inhibit: Same as for step 2G.

6E. Arithmetic Digit Inhibit: Same as for step 2B.

7. FORCE I-E CHANGE:

The I-E Change indicating the end of the operation can occur at the end of the three digit add if no zone carry is detected. Under this condition, steps 5B through 6E are omitted. The instruction cycle controls reset the operation and block the remainder of the sequence. If a zone carry is detected, the complete sequence is followed.

7A. Address Modify I-E Change (Carry):

42.65.02.2	Address Mod	Cycle Tr 1
3B-4B	I-E Change	Not Cycle Tr 2
		Cycle Tr 3
		Not Zone Adder Carry

7B. Address Modify I-E Change (No Carry):

42.65.02.2	Address Mod	Cycle Tr 1
3D-4C	I-E Change	Cycle Tr 2
		Cycle Tr 3
		Not Block Op B-Cycle

Full Storage Print

A *full storage print feature* is added with expanded storage. This feature is a modification of the manual storage print on the basic 4K system. The storage print operation is unchanged. One line may be printed starting with any hundreds address block, including the expanded storage area. A second line is immediately printed to show digit ones in the location of all word-marks in the storage area. With the larger storage, it becomes inconvenient to print the full storage for examination. The full 16K storage requires 160 manual switch settings.

With the mode switch set for storage print, an added switch on the auxiliary console converts the operation to full storage print. The full storage print control causes an automatic print-out, starting with 001, of the complete storage with a single switch setting. The settings of the console address switches have no effect on the operation. Each print line starts on a hundreds address block. If the printer has 132 print positions, the first portion of the next block is printed at the end of the line. These positions are printed again at the start of the next line. Between each data line printed, a word-mark print operation indicates the word-mark positions with the digit one.

Operation and Data Flow

The special controls for the full storage print operation are located in the 1406 frame. These controls provide the cycle gates to the printer control circuits. Operation differs with the installation of the print buffer feature. The control for the non-buffered system is described first, with differences for the buffer detailed later.

Data flow differs from a normal print operation in the use of different storage addresses. A normal operation automatically makes use of a specified storage area for printing (201-332). The scanning circuits are designed to cover this area. The basic storage print operation substitutes the hundreds and thousands address switches for the starting hundreds digit. For full storage print, special controls adjust the hundreds digit of the B-Star to provide a reset to the starting address.

The operation is started after setting the console switches by pressing the start key. The full storage print operation trigger is set ON and an address 001 is forced into the storage address register. The operation trigger remains set until the address 000 is detected. As long as the trigger is set, use of the print scan complete gate to end the operation is blocked. Another print operation is started for the next line. The control of the B-Star modification and the storage address register entry causes block-by-block advance through the storage addresses.

At the start of the operation the storage address register is set to 001 instead of 201 for the first block. The +3 modifier control is used as with a normal print operation. When the tens drive T13 is reached, the −1 modifier control is forced for the hundreds position, causing the 130 to modify to 033. On the next cycle the normal print circuits develop address 202. For full storage print, the hundreds digit is taken from the B-Star, while the units and tens digit read in from the 202 set. At the end of the second sub-scan, the same procedure causes the third sub-scan to start at 003. This cycling repeats for each of the 49 print scans, when the print scan complete causes the WM print trigger to set.

A second print cycle is started immediately to print the word-marks from the same storage area. The last character read out was from address 132, which modified to 035. The hundreds digit combined with the 201 set signal produces an address of 001 in the storage address register to start the new printing line. The address modification follows the same pattern as for the data printing until the last sub-scan of scan 49. The print scan complete gate on a word-mark cycle blocks the development of the forced −1 modifier control used to back-up the hundreds digit. The address 132 modifies to 135 in the normal manner. The print scan complete signal resets the word-mark print trigger to return the printing control to normal.

The operation trigger remains set to immediately start the next data print cycle. The advanced B-Star hundreds digit combined with the units and tens digit of the 201 set result in a starting address of 101. The operation cycles through 49 scans for the data printing and 49 scans for the word-mark printing in the same manner as the first block. On the last sub-scan the hundreds digit back-up is blocked, leaving the digit one point higher than previously. This advance of the hundreds digit allows successive printing of the hundreds address blocks. The process continues until, after printing the full storage, the 000 address is detected on the word-mark cycle. This signals the last

line of the operation by resetting the operation trigger. When the print scan complete gate sets, a print scan end signal is developed. Being a manual operation, the print scan end causes the reset of the DELTA PROCESS and D-CYCLE latches to end the operation.

The full storage print operation with print buffer differs in the storage read-out controls. Instead of repetitively scanning the main storage directly for the line of printing, the hundreds address block (132 characters) is transferred serially to the print buffer. The print scanning occurs on the print buffer, while the processing unit is inactive. No time gain is made because the system is in a manual status. Both the data line and the word-mark line are printed from one data transfer. The print transfer trigger setting is blocked and made to appear as transfer-complete to start the second print cycle (WM print). In the non-buffered operation, the setting of the print ready trigger develops the storage set addresses (201 set, etc). The buffered system forces the 201 set signal to start the data transfer. When the complete block is transferred, the print transfer trigger is reset and the print ready trigger set for the actual printing.

The transfer of data starts at the forced address 001 for the first line. The print buffer feature uses the +1 modifier instead of the +3 modifier control of the unbuffered operation. When the 132 address is reached the transfer stops. The hundreds back-up (−1 modifier) is blocked with print buffer by forcing the signals normally used to advance at the end of each block. For the next line with print buffer, the hundreds digit from the B-Star combines with the units and tens of the 201 set for an address 101. Each hundreds address block follows in sequence with the data transfer, the data print, and the word-mark print. The operation ends after transferring the last hundreds address block by resetting the operation trigger and developing print scan end.

Control Circuits

The full storage print operation makes use of the normal print circuits with additional controls located in the 1406. The basic storage print circuits function within the hundreds block while the full storage controls step from block to block.

Objectives (Full Storage Print)

1. Setup Operation
   - A. Mode Switch Set — Storage Print
   - B. Auxiliary Switch Set — Full Storage Print
   - C. Set D-cycle latch
   - D. Force B-cycle
   - E. Set Operation Control Trigger
   - F. Set Print Ready Trigger
2. Address Control
   - A. Generate Address 001
   - B. Block Address Switches
   - C. Gate B-Star H-TH to Star
   - D. Block Hundreds of 201 Set
3. Data Line — Scan 1 — Sub-Scan 1
   - A. Set Address Control 201 for 001 (X01)
   - B. Set Print Scan Trigger
   - C. Set Plus Three Modifier Control
   - D. RO Address 001 (X01)
   - E. RO Addresses 004 to 130 (X04 to Y30)
   - F. Block Tens B-C Control
   - G. Hundreds Position −1 Modifier
   - H. Set Print Ready Trigger
4. Data Line — Scan 1 — Sub-Scan 2
   - A. Set Address Control 202 for 002 (X02)
   - B. Set Print Scan Trigger
   - C. Set Plus 3 Modifier Control
   - D. RO Addresses 002 to 131 (X02 to Y31)
   - E. Block Tens B-C Control
   - F. Hundreds Position −1 Modifier
   - G. Set Print Ready Trigger
5. Data Line — Scan 1 Sub-Scan 3
   - A. Set Address Control 203 for 003 (X03)
   - B. Set Print Scan Trigger
   - C. Set Plus Three Modifier Control
   - D. RO Addresses 003 to 132 (X03 to Y32)
   - E. Block Tens B-C Control
   - F. Hundreds Position −1 Modifier
   - G. Set Print Ready Trigger
6. Data Line—Scan 2 through 48
   - A. Repeat steps 3, 4, and 5.
7. Data Line — Scan 49
   - A. Repeat steps 3 through 5F
   - B. Set Print Ready Trigger
   - C. Set Print Scan Complete Gate
   - D. Set Print Scan Complete Trigger
   - E. Set Word-Mark Print Trigger
8. Word-Mark Print Line Scans 1 through 48
   - A. Repeat steps 3 through 6
9. Word-Mark Print Line Scan 49
   - A. Repeat steps 3 through 5D
   - B. Not Block Tens B-C Control
   - C. Set Print Ready Start
   - D. Set Print Scan Complete Gate
   - E. Set Print Scan Complete Trigger
   - F. Reset Word-Mark Print Trigger
10. Hundreds Blocks 2 through Last −1
    - A. Repeat steps 3 through 9
11. Last Hundreds Block
    - A. Repeat steps 3 through 9
    - B. Reset Operation Control Trigger
    - C. Develop Print Scan End
    - D. Reset D-cycle Latch

Logic Detail (Full Storage Print)

1. SETUP OPERATION

1A. Mode Switch Set — Storage Print: At this setting the run line is down and a mode-select line is up to effect a manual control.

37.30.11.2 2-	Storage Print-Out	Manual Set

1B. Auxiliary Switch Set — Full Storage Print: Setting this switch in combination changes the basic single hundreds block print to full storage.

37.31.11.2 4G	Full Storage Print-Out	Storage Print-Out Manual Set

1C. Set D-Cycle Latch: The D-cycle signal substitutes for the process controls in a manual operation. It allows using the B-cycle controls without a program advance.

31.03.31.2 3B	D-Cycle	Start Mode Select Time 090-000

1D. Force B-cycle: The output of the latch is gated with D-cycle for this operation.

31.26.11.2 3F	B-Cycle Latch	Storage Print-Out Not Star Key

1E. Set Operation Control Trigger: The output serves to gate the print ready start and block the print scan end to give repeated print cycles.

42.66.03.2 3E-4A	Print Ready Start Expanded Storage	Full Storage Print Start

1F. Set Print Ready Trigger: The print ready trigger sets up for the first character of the scan. The trigger turns OFF when a printer drum pulse indicates the chain is in position. Its turnoff sets the scan latch.

36.31.31.2 3E-4E	Print Ready	Storage Print-Out Start

2. ADDRESS CONTROL

2A. Generate Address 001: This address is needed only for the first sub-scan because the previous value in the B-Star may not be zero.

42.66.03.2 3E	Generate 001	Full Storage Print Start

2B. Block Address Switches:

32.30.11.2 3D	Not H-TH D-Cycle Set	Not Full Storage Print

2C. Gate B-Star H-TH to Star: This gate is not used for the first cycle when the generate 001 is active.

42.66.03.2 3H	B-Star Gate Out Storage Print	Not Start Full Storage Print Auto Set Time 090-000

2D. Block Hundreds of 201 Set: The 202 set and 203 set are blocked in the same manner.

32.21.11.2 4F	Not Auto Set Hundreds 2	201 Set Storage Print-Out

3. DATA LINE — SCAN 1 — SUB-SCAN 1

3A. Set Address Control 201 for 001 (X01): Only the 01 is used for this operation, the hundreds digit coming from the B-Star.

36.31.31.2 6A	Set 201	Print Ready SS Ring 3

3B. Set Print Scan Trigger:

36.31.41.2 4A	Print Scan	Not Print Ready Not Tr-5

3C. Set Plus Three Modifier Control:

36.31.71.2 4J	Plus 3 Ctrl Mod	Print Scan

3D. RO Address 001 (X01): The B-cycle has been forced and activate storage is forced; the character at address 001 reads out and regenerates in the normal manner. The character is compared to the printer chain position to determine if the hammer should fire.

3E. RO Addresses 004 to 130 (X04 to Y30): The remaining 43 character positions for the sub-scan are read out in succession using the plus 3 modifier. Characters which compare with the chain position are printed.

3F. Block Tens B-C Control: This signal blocks the normal modifier transfer control by forcing the carry control. The blocking allows use of the minus 1 modifier to back up the hundreds digit.

42.66.03.2 3F-4G	Block Tens B-C Control	Full Storage Print Tens Drive T-13 Not WM Print and Print Scan Comp Gate

3G. Hundreds Position −1 Modifier for 033 (X33): The use of the −1 Modifier changes the B-Star entry from 133 to 033.

32.42.21.2 4H	Modifier Minus 1	Block Tens B-C Ctrl Time 060-090

3H. Set Print Ready Trigger:

36.31.71.2 6A	Print Ready Start	Not Units Drive U2 Tens Drive T13 Time 060-090

4. DATA LINE — SCAN 1 — SUB-SCAN 2

4A. Set Address Control 202 for 002 (X02): Only the 02 is used for this operation, the hundreds digit coming from the B-Star.

36.31.31.2 6B	Set 202	SS Ring 1 Print Ready Not First Scan

4B. Set Print Scan Trigger:

36.31.41.2 4A	Print Scan	Not Print Ready Not TR 5

4C. Set Plus Three Modifier Control:

36.31.71.2 4J	Plus 3 Ctrl Mod	Print Scan

4D. RO Addresses 002 to 131 (X02 to Y31): The 44 positions are read out in the same manner as for sub-scan 1, or similar to a normal print.

4E. Block Tens B-C Control:

42.66.03.2 3F-4G	Block Tens B-C Control	Full Storage Print Tens Drive T-13 Not WM Print and Print Scan Complete

4F. Hundreds Position −1 Modifier for 034 (X34): B-Star entry becomes 034 instead of 134.

32.42.21.2 4H	Modifier Minus 1	Block Tens B-C Ctrl Time 060-090

4G. Set Print Ready Trigger:

36.31.71.2 6A	Print Ready Start	Not Units Drive U2 Tens Drive T13 Time 060-090

5. DATA LINE — SCAN 1 — SUB-SCAN 3

5A. Set Address Control 203 for 003 (X03): Only the 03 is used for this operation, the hundreds digit coming from the B-Star.

36.31.31.2 6C	Set 203	SS Ring 2 Print Ready Not First Scan

5B. Set Print Scan Trigger:

36.31.41.2 4A	Print Scan	Not Print Ready Not Tr 5

5C. Set Plus Three Modifier Control:

36.31.71.2 4J	Plus 3 Ctrl Mod	Print Scan

5D. RO Addresses 003 to 132 (X03) to Y32): The 44 positions are read out in the same manner as for sub-scan 1 and 2.

5E. Block Tens B-C Control:

42.66.03.2 3F-4G	Block Tens B-C Control	Full Storage Print Tens Drive T-13 Not WM Print and Print Scan Complete

5F. Hundreds Position −1 Modifier for 035 (X35): B-Star entry becomes 035 instead of 135.

32.42.21.2 4H	Modifier Minus 1	Block Tens B-C Ctrl Time 060-090

5G. Set Print Ready Trigger:

36.31.71.2 6B	Print Ready Start	Not Scan 49 Last Address

6. DATA LINE — SCANS 2 THROUGH 48

6A. Repeat steps 3, 4, and 5 for each scan.

7. DATA LINE — SCAN 49

7A. Repeat steps 3, 4 and 5 through 5F.

7B. Set Print Ready Trigger: When repeating this step for the last hundreds block, not-word-mark print gates instead of print ready start expanded storage.

36.31.71.2 6C	Print Ready Start	Print Ready Start Ex Stor Storage Print-Out Last Address Time 060-090

7C. Set Print Scan Complete Gate:

36.31.41.2 3D-3E	Pr Scan Comp Gate	Scan 49 Last Address Time 030-060

7D. Set Print Scan Complete Trigger:

36.31.41.2 4D	Pr Scan Comp	Pr Scan Comp Gate Not Tr 5

7E. Set Word-Mark Print Trigger: Setting the word-mark trigger causes the next line to print ones in the positions of the word-marks in storage. The data does not print. The carriage spaces the forms for the next line.

36.31.41.2 3G-5G	WM Print	Storage Print-Out Not WM Print Print Scan Comp

8. WORD-MARK PRINT LINE — SCANS 1 THROUGH 48

8A. Repeat steps 3 through 6.

9. WORD-MARK PRINT LINE — SCAN 49

9A. Repeat steps 3, 4, and 5 through 5D.

9B. Not Block Tens B-C Control: Preventing the block control on the end of the word-mark print cycle allows the B-Star address to stand at 135. Using the

hundreds position for the setup of the next line starts the scan at 101 instead of 001, thus an advance into the next hundreds block.

42.66.03.2 2G-4G	Not Block Tens B-C Controls	WM Print

9C. Set Print Ready Trigger:

36.31.71.2 6C	Print Ready Start	Print Ready Start Ex Stor Storage Print-Out Last Address Time 060-090

9D. Set Print Scan Complete Gate:

36.31.41.2 3D-3E	Print Scan Comp Gate	Scan 49 Last Address Time 030-060

9E. Set Print Scan Complete Trigger:

36.31.41.2	Pr Scan Comp	Pr Scan Comp Gate Not Tr 5

9F. Reset Word-Mark Print Trigger:

36.31.41.2	Not WM Print	Print Scan Complete WM Print

10. HUNDREDS BLOCKS 2 THROUGH LAST −1

10A. Repeat steps 3 through 9 for each block.

11. LAST HUNDREDS BLOCK

11A. Repeat steps 3 through 9.

11B. Reset Operation Control Trigger: The print ready start expanded storage gate supplied by this trigger is not needed after the start of the last block.

42.66.03.2 2A-3A-4A	Pr Scan End Exp Storage	Address 000 WM Print Time 030-060

11C. Develop Print Scan End:

36.31.01.2 3D	Print Scan End	Print Scan Comp Gate Print Scan End Exp Stor WM Print

11D. Reset D-Cycle Latch: The D-cycle was serving as the process control. The system returns to the full stopped status until the next console command.

31.03.31.2 2G	Not D-Cycle	Print Scan End Time 075-105

The advanced programming option for the 1401 adds several features which permit greater program flexibility. These features fall into three groups:

1. *Address Indexing,* which allows adding an increment to either or both the A and B addresses specified by the instruction. Three index registers located in storage are selected by zone tags in the tens position of the address. The absence of a tag makes the feature inoperative.
2. *Address Store* allows storing one of the address registers in a specified address. Two new operation codes are used for this purpose. Operation H stores the B-address register, and operation Q stores the A-address register. The I-address register can only be stored following a branch, when it is transferred to the B-address register.
3. *Move Record* allows moving a random size block of words as defined by an A-field record or groupmark. A P-operation code is provided for this transfer. The fields are reverse-scanned to read the record mark at the end of the transfer. The operation cannot be chained.

The basic system permits only a direct approach to the locations of instructions and data. Complex problems require individual instructions for each level. In branch, move, and chain operations, the address of the next available position is lost except by direct program retrieval. Movement of random length records requires the processing of a fixed record of maximum length. With the advanced programming concept, these operations become more automatic. The results of one operation are, in effect, written into the program for future steps.

Address Indexing

The index feature permits adding an increment to the A and B addresses specified by the instruction. This becomes useful in types of operations using the same program routine for each processing step. In practice, one or both factors are located in sequential blocks or problem groupings. Without indexing, either multiple groups of program steps or an arithmetic program to update is required. With indexing, a simple program routine looped on itself remains constant, while the addresses are effectively advanced by controlling the index increment. Use of the index feature is optional. The A-address, the B-address, or both addresses may be indexed as desired.

Index operations are always performed as true add. When it is desired to decrease the address value, the index value is made large enough to wrap-around the end of the storage area. With the basic storage only installed (4K or less), a value of 4,000 minus the decrement is used. For expanded storage, a value of 16,000 minus the decrement is used. Addresses developed in the unused areas produce an invalid address signal.

Three three-digit areas are assigned location in storage to serve as index registers. Designated as A, B, and AB, they are located as follows:

A	087 - 088 - 089
B	092 - 093 - 094
AB	097 - 098 - 099

Indexing is indicated by the zone bits in the tens position of either or both of the addreses. An A-bit designates the A-index, a B-bit the B-index and both bits the AB index. Any of the index registers may be used with either address. The index registers are loaded and adjusted in the same manner as any storage location.

The instruction cycle is started in the usual manner. When the tens digit of the A-address is read in at I-Ring-2, it is tested for a zone bit tag. If a tag latch is set, the index control is initiated. The reset of the I-Ring is inhibited at the end of the I-Ring-3 step. The I-Ring-3 output remains on throughout the index transfer. A four position trigger ring is used to cycle index and store operations. The first three ring outputs are timed similarly to delta cycle pulses. The ring outputs switching with the tag gate generate the index register address in the storage address register. The adder operation continues, placing the increased factor in the A-address register. In most operations the B-address register is also controlled simultaneously. At the end of the index operation, the I-Ring reset is permitted and the instruction continues to read in the B-address, if one is used. The tens position is tested for an index tag during I-Ring-5. If a tag is present, the index operation repeats on I-Ring-6 to increase the B-address register.

The store-index ring is started and the units position of the required address register is transferred to the

A-register. The storage address register is forced to the units position of the index, which reads out to the B-register. Both the digits and the zones are added. If expanded storage is not installed, a zone check bit is forced to validate. A carry from the digit adder is not entered in the zone adder but is retained for the tens position digit add. A zone adder carry is not used, since it only represents a wrap-around, which may be intentional. The result is returned to the units position of the originating address register.

The store-index ring advances to cycle 2 to process the tens digit. The read-in of the units position to the address register actually occurs after the ring advance. The address register tens position transfers to the A-register, and the tens position of the index factor enters the B-register, the same as for the previous cycle. The tens position has no zone, but the zone adder is left on with a zone check bit entered to validate. The digit carry is not blocked from the zone adder in this case, but the output of the zone adder is not used. The digit carry is retained for the hundreds digit add. The result is returned to the tens position of the originating address register.

The store-index ring advances to cycle 3 to process the hundreds position. The read-in of the tens position occurs after the advance. The address register hundreds position transfers to the A-register, and the hundreds position of the index factor enters the B-register. Both digit and zone bits are added, and the carry from the digit adder is entered in the zone adder. During the processing, the store-index cycle-four trigger is set to allow reading the result back into the address register hundreds position. In the basic 4K system and in the expanded storage without a zone carry, the operation ends at this point by causing the I-Ring to advance.

If a zone carry occurs, the zone carry trigger is set, and the cycle-1 trigger is set again to enter the carry in the units zone. The units position of the address register is transferred to the A-register. With the zone carry trigger on, the storage read-out is blocked and an A-bit is forced into the B-register. An A-bit represents a value of one unit and thus serves as a carry entry to the zone adder. The carry from the hundreds digit add is reset. The digit adder being fed with a blank from the B-register (A-bit only) sets a binary and quinary zero which causes no change in the digit portion. The resultant character is returned to the units position of the address rgeister under control of the zone carry trigger. The zone carry trigger also blocks further advance of the store-index ring and causes the I-Ring to advance. The remainder of the instruction then reads in, in the normal manner.

Address Store

Store B-Star Operation — H (AAA)

The store B-Star operation is provided to allow retaining the address left in the B-address register. In some cases of operation chaining, and in cases of move and load operations, the last storage address may be a variable. The address represents a point of subsequent program operations. With the I-B Star transfer feature, added with advanced programming, the next instruction address of the main routine has been moved to this register. This instruction must immediately follow that leaving the desired address.

The operation and A-address are specified. The B-address register reset and read-in are inhibited during the instruction cycle. The A-address may be indexed during read-in. During the execute cycle only A-cycles are used. The B-cycle controls are suppressed. Three positions of the store-index ring are used to control the transfer. The last digit position is sensed from the ring and no word-marks are used.

The B-Star is transferred, one digit at a time, to the A-register over an added data line. The storage position designated by the A-address reads out to the B-register and is lost. The value in the A-register is read into the storage location. The transfer starts with the low-order position. The A-Star is modified by minus one (forward scan) for the next position. The I-E change is developed from the third ring position, and a special delta I-cycle set is provided since the B-cycle is not used.

Store A-Star Operation — Q (AAA)

The store A-Star operation is provided to allow retaining the address left in the A-address register. This becomes desirable after some types of move and load operations. The address may be the starting point for a subsequent operation. The store operation is used to place the address directly into the later instruction. This instruction must immediately follow that leaving the desired address.

The operation and A-address are specified. The B-address register normal reset and read-in are inhibited during the instruction cycle. The A-address may be indexed during read-in. The Q-operation is unique in that part of its operation is performed during the instruction cycle. The A-address to be saved must be moved to the B-address register. Normal read-in of the new A-address is inhibited as soon as the Q-operation is detected. The index feature interlocks are used to hold the I-Ring in the Op position and prevent further read-in. A one digit delay provides time to transfer the

previous A-address. The address is transferred to the storage address register and scanned out to the address modifier. The modifier control transfer gate is on to provide a direct serial read-in to the B-address register. Following this transfer the I-ring is allowed to reset and advance to complete the instruction read-in.

The execute cycle for the Q-operation is identical to that for the H-operation for store B-Star. Only A-cycles are used, with the B-cycle controls inhibited. Three positions of the store-index ring are used to control the transfer. The last digit position is sensed from the ring and no word-marks are used.

The B-Star is transferred, one digit at a time, to the A-register over an added data line. The storage position designated by the A-address reads out to the B-register and is lost. The value in the A-register is read into the storage location. The transfer starts with the low order position. The A-Star is modified by minus one (forward scan) for the next position. The I-E change is developed from the third ring position, and a special delta I-cycle is provided since the B-cycle is not used.

I-B Star Branch Transfer

The I-B Star transfer feature is automatic. When a compare operation or other operation resulting in a branch routine was performed, the next instruction address was lost. With the transfer feature, the I-address register is transferred to the B-address register. This allows the address to be subsequently stored by using the store B-Star operation (Opr H). The original B-Star address is lost during the transfer, but with most branch operations it is of no consequence.

The branch condition is tested in the normal manner. If the branch condition is satisfied, the program skip latch is set to restart the I-Ring and to provide the I-A Star gate. The latter gate in turn sets an I-B Star latch to provide a one digit delay. The I-B Star gate inhibits the I-Ring start during the delayed digit, and provides the transfer circuit gates.

The I-address register is transferred to the storage address register, which in turn is scanned out to the address modifier. The modifier control transfer control is on to provide direct serial read-in to the B-address register. The instruction cycle then continues. If the next operation is a store B-Star (Opr H), the address is stored. If any other operation, it is lost.

Move Record

Move Record Operation — P (AAA) (BBB)

The move record operation is provided to permit moving random length records for input/output transfers. The operation is similar to an M-operation, except that the word-mark is not used to stop the transfer. The operation stops when either a record-mark (82A) or a group-mark (8421 AB WM) is encountered in the A-field. The fields are reverse-scanned (Modifier +1) so as to encounter the mark at the end of the record.

The instruction cycle is normal. The instruction contains the operation, an A-address, and a B-address. Both specify the high order of the fields. The transfer is from A-field to B-field. The execute cycle uses both A-cycle and B-cycle controls. During the A-cycle, the A-field character is read out of storage to the B-register. The character is regenerated in storage and read into the A-register. The A-address register is used and its value increased by one. During the B-cycle, the B-address register is used and increased by one. The character is read out of storage to the B-register and is lost. The A-register is read into storage to fill the vacancy. A-field word-marks, while not used in the operation, are retained in the A-cycle transfer.

When either a record-mark or a group-mark is sensed in the A-register, the transfer is stopped at the end of the B-cycle. The mark is the last character transferred. The I-E change signal is developed to start the next instruction.

Functional Component Circuits

Store-Index Ring (Figure 3)

The *store-index ring* is composed of four CTDL triggers connected for pulse advance. The first three triggers are used for both store and index operations. The fourth trigger serves as a trailer gate for index operations. The first three stages advance with periodic timing pulses under control of the operation gate. The timing differs between the two operations.

All triggers are DC-reset by I-Ring-1 or START RESET. Store cycle 1 trigger is driven both on and off by the periodic set pulse. It is gated to turn on if all three triggers are off. The store cycle 2 trigger sets with the turning off of store cycle 1. It is turned off with the periodic set pulse. Store cycle 3 trigger sets with the turning off of store cycle 2. It, like the other two stages, is reset by the set pulse. In the basic system the cycle stops with one sequence because the operation gate goes down. The fourth trigger serves as a follower gate for store cycle 3. It is set and reset from clock pulses with I-Ring -3 or -6 as gating control. It is only used for the index operation.

For expanded storage, when a zone carry must be accommodated in indexing, the operation gate remains and the store cycle 1 trigger is allowed to set again.

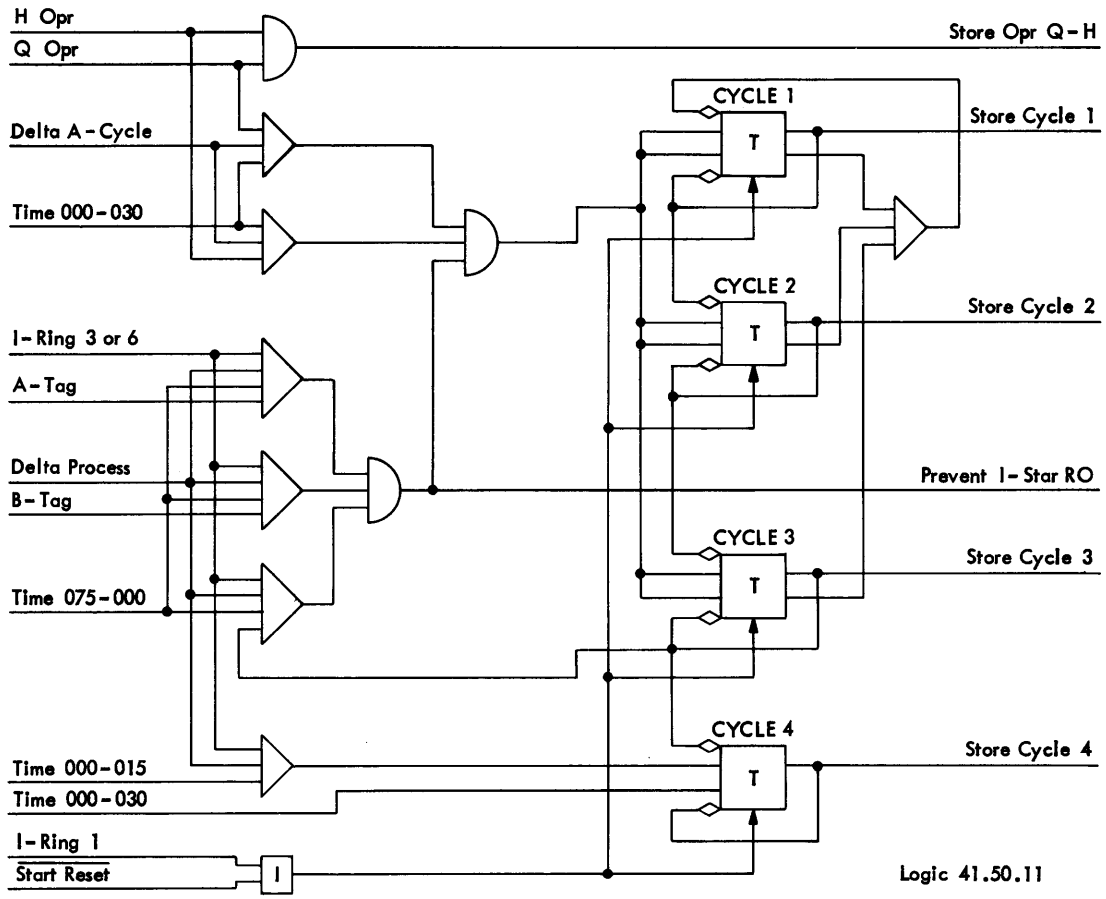


Figure 3. Store-Index Ring

The zone carry trigger becomes the follower gate during this cycle. The store cycle 2 trigger is blocked from setting a second time by the carry trigger.

The timing of the three ring position for a store operation corresponds to cycle control pulses. They are required to get the B-address register characters to storage during normal transfer time. One ring position is required for each character transferred.

The timing of the first three positions for the index operation corresponds to the delta cycle control pulses. The earlier timing is required to generate the index register address in storage. The earlier timing requires the use of two positions to process a character. The initial transfer of the first character is made with store cycle 1. The return of the adjusted character from the adder occurs at the start of store cycle 2. The third character is returned by the fourth trigger, which is timed as a cycle control pulse.

Address Register to A-Register Transfer (Figure 4)

The transfer of the address characters from the A- and B-address registers to the A-register requires a new data flow path (Figure 5). The individual positions of the address register are gated out by the store-index ring. The output is mixed with the normal A-register set from the B-register. The low order reads out on store cycle 1 with the high order on store cycle 3. All bits are read out directly except the check bits. The CD and CZ bits are combined to develop the C-bit for the A-register. During the latter portions of these cycles the characters are processed from the A-register.

Index Arithmetic Output

A special adder output is developed for indexing operations. The normal output is concerned with storage inhibit controls. The new output is gated with the index trigger to form the Index Star. The output is mixed with the output of the address modifier to feed the address register input line. Figure 5 of the modified data flow shows the relation of this path.

Index Address Generation (Figure 6)

The index operation requires the generation of addresses for each position of the index register. Combinations of index tag and ring position gate these

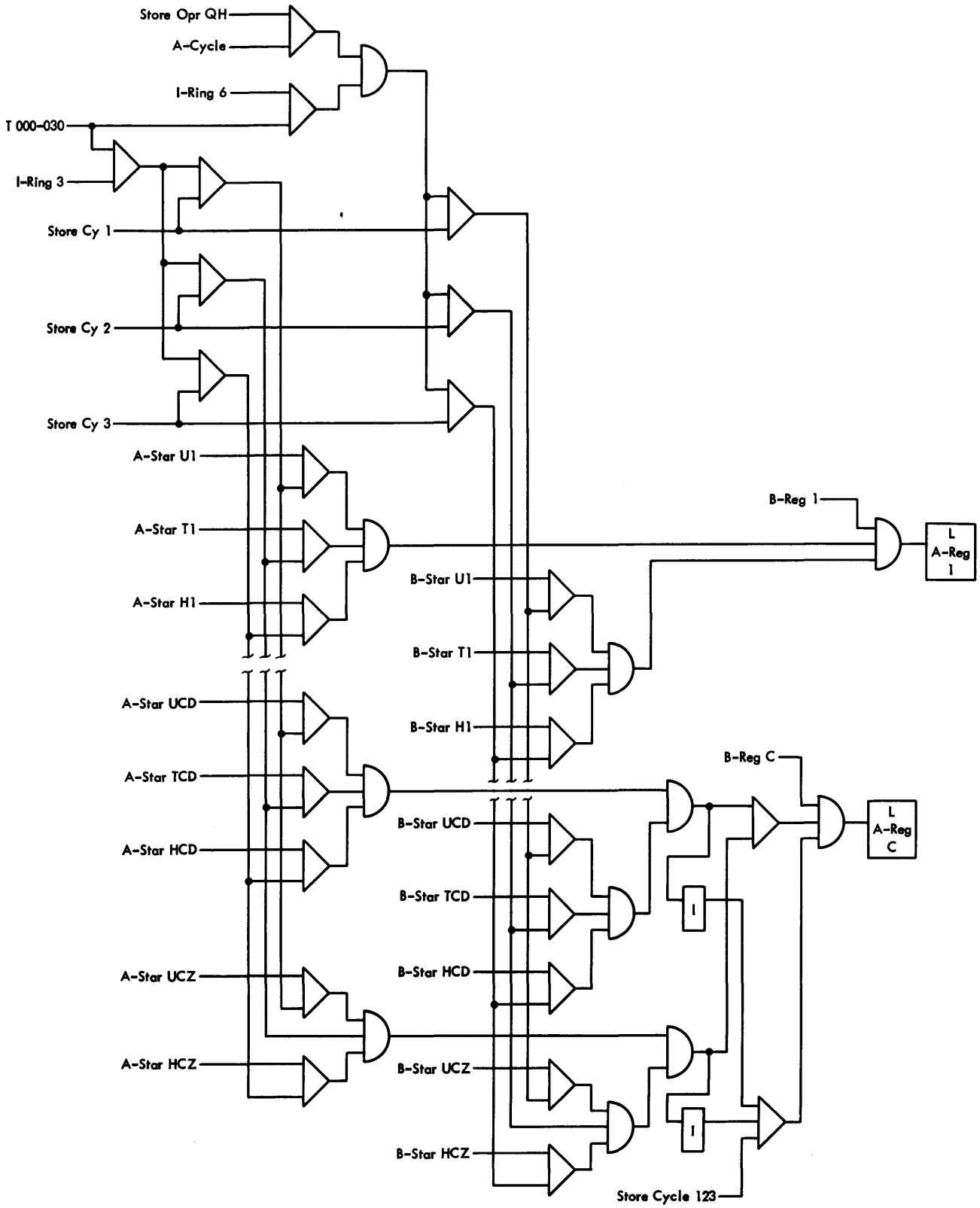


Figure 4. Address Register to A-Register Address Path

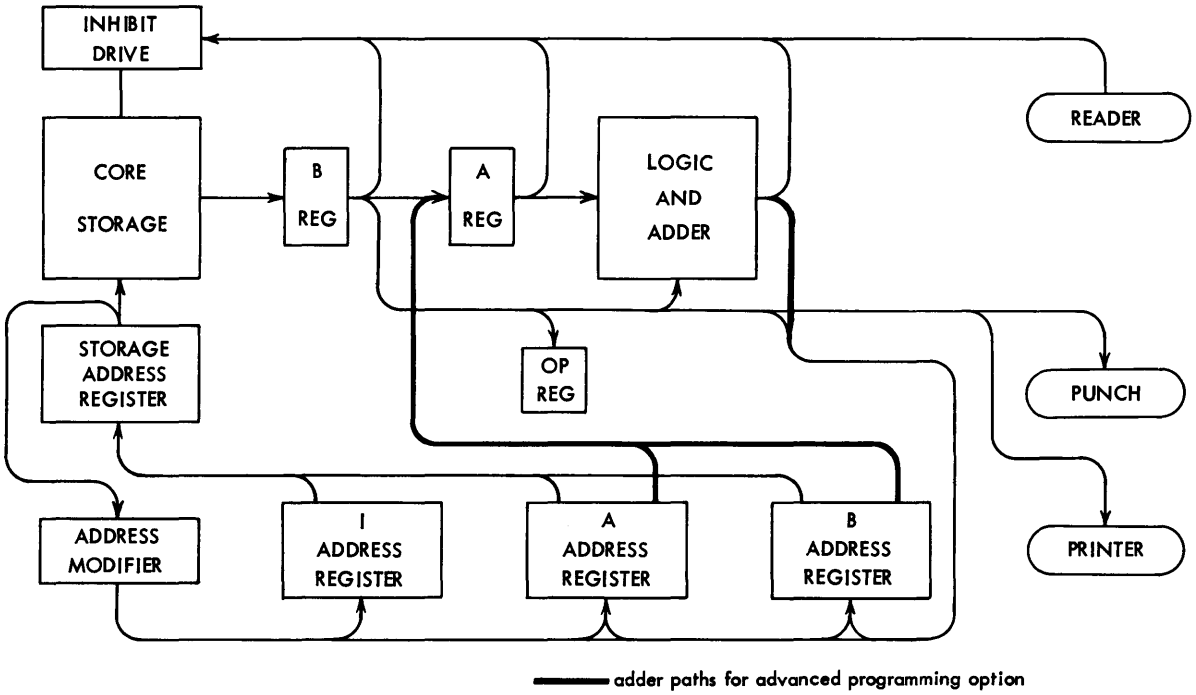


Figure 5. IBM 1401 Data Flow

addresses to the storage address register. Normal address register and modifier operation are prevented during the index cycle.

Indexing is initiated and the register selected by the A- and B-bits over the tens position of the address. The following address combinations are used:

TAG BIT	CYCLE 1	CYCLE 2	CYCLE 3
A	089	088	087
B	094	093	092
AB	099	098	097

These addresses have been selected for ease in logic switching. The units position is the same when the A-bit is present. The addresses for the B-bit are prevented if the A-bit is also present. The tens position of the addresses is the same when a B-bit is present. The digits are such that is only necessary to add a *one* bit and a *check* bit to change the number.

Logic Circuits

Indexing Operation (Timing Chart — Figure 7)

The A-address specified in the instruction is increased by the A-index register to arrive at an address to be used in the operation. The same conditions may occur for the B-address. Any of the three index registers could be used in either or both cases. The complete sequence assumes a zone carry entry into the units position.

Objectives (Indexing Operation)

1. Sense Index Tag
   A. Set A-tag Latch
2. Prevent I-Address Register Gate-out
   A. Prevent I-Star RO
   B. Not Index
   C. Prevent I-Star Gate-out
3. Start Store-Index Ring
   A. Gate Store Cycle 1
   B. Set Store Cycle 1
4. Prevent I-Ring Advance
   A. Block I-Ring
   B. Prevent I-Ring Advance
5. Set Storage Address Register
   A. Generate UP Index Address
   B. Generate TP Index Address
   C. Generate HP Index Address
6. Transfer A-Star UP to a A-Reg
   A. A-Star UP Gate-out
   B. Store-Index A-register Reset
   C. Set A-reg Latches
7. RI Storage Address to B-Register
   A. B-register Reset
   B. Set B-register Latches
8. Advance Store Index Ring
   A. Reset Store Cycle 1
   B. Set Store Cycle 2
9. Arithmetic Output to Modifier Bus
   A. Set Index Arithmetic RO Gate
   B. Set Zone Adder RO Gate
   C. Index Star Bus
   D. Modifier Bus Mix

10. RI A-Star and/or B-Star
    A. A-Star RI
    B. B-Star RI Control
    C. B-Star RI
11. Repeat steps 4 through 10 for Tens Position
12. Repeat steps 4 through 7 for Hundreds Position
13. Advance Store-Index Ring for Cycle 4
    A. Set Store Cycle 4
14. Repeat steps 9 and 10 for Hundreds Position
15. Zone Adder Carry Test
    A. Set Zone Carry Trigger
    B. Set Store Cycle 1
    C. Reset Store Cycle 3
16. Set B-Register for Zone Carry
    A. Prevent Activate Storage
    B. Reset B-register for Carry
    C. Insert Zone Bit for Carry
17. Repeat steps 6, 9, and 10 for Units Position Zone
18. Reset Store Index Ring Controls
    A. Reset Store Cycle 1
    B. Reset Tag Latches
    C. Reset Zone Carry Trigger
19. Continue I-Ring Advance
    A. Allow I-Star Gate-Out
    B. Advance I-Ring
    C. Not Index Interlocks

Logic Detail (Indexing Operation)

1. SENSE INDEX TAG

1A. Set A-Tag Latch: The B-tag is set in the same manner. Either one or both latches may be set to cause indexing. If neither is set, no indexing is performed. The tag designates the index register to be used (one of three).

ILD 41 B2	A-Tag	B-Reg A I-Ring 2 (or 5) Time 060-090

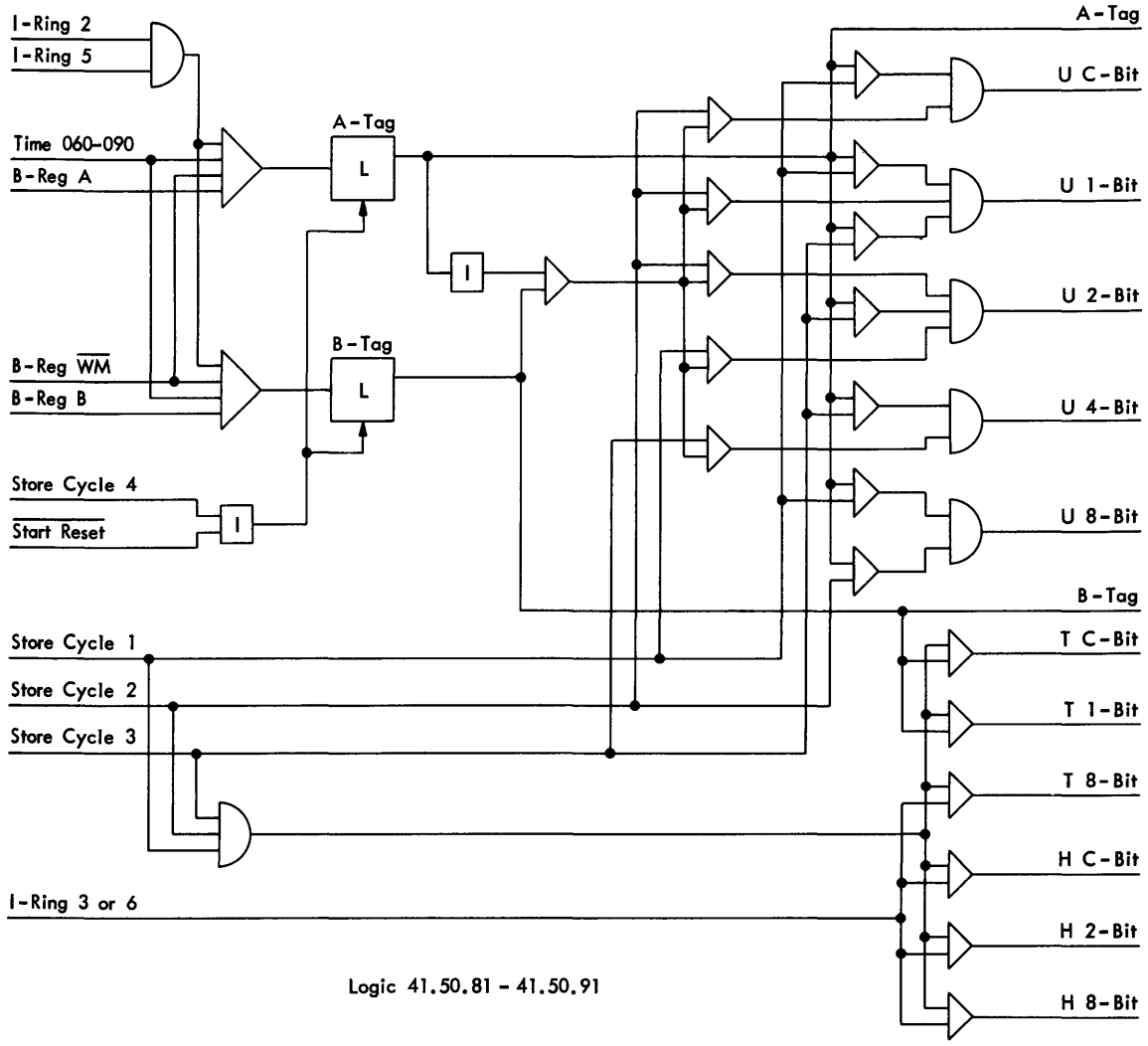
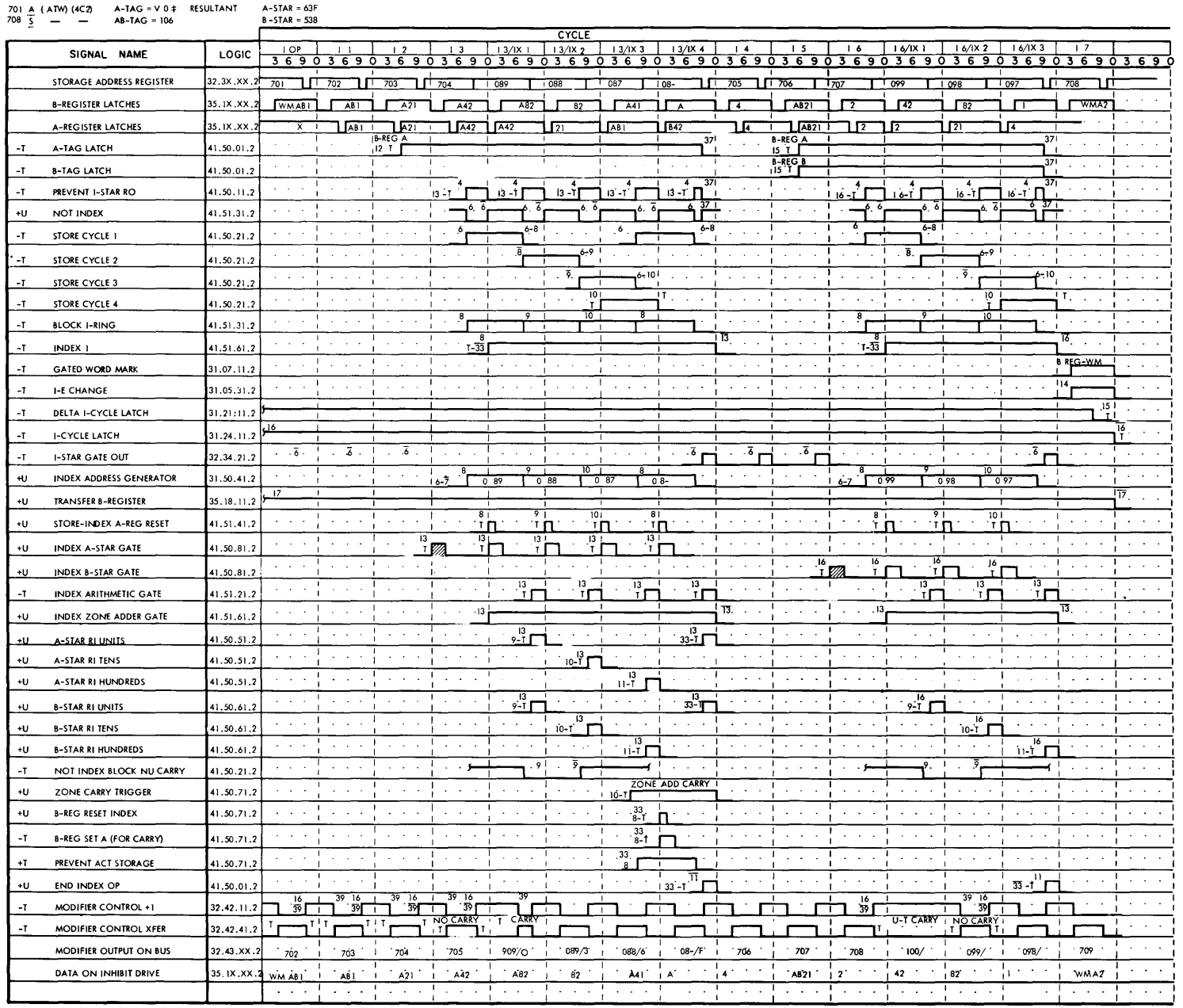


Figure 6. Index Address Generator

Figure 7. Instruction Cycle with Indexing



2. PREVENT I-ADDRESS REGISTER GATE-OUT

2A. Prevent I-Star RO: This gate is used for both gating the various index functions and starting and advancing the store ring. The pulse occurs each digit time.

ILD 41	Prevent I-Star	A-Tag
B2	Read-Out	I-Ring 3 (or 6)
		Time 075-000

2B. Not Index: The not-index signal is used for normal gating. Index prevents these gatings for its duration.

ILD 41	Not Index	(Not) Prev I-Star RO
A2		(Not) Q Modify

2C. Prevent I-Star Gate-out: This signal, when powered, normally gates the I-Star to the storage address register.

ILD 18	I-Star Gate-Out	Not Index
A1		

3. START STORE-INDEX RING

3A. Gate Store Cycle 1:

ILD 41	Store Cycle 1	Not Store Cycle 1
B4	Gate	Not Store Cycle 2
		Not Store Cycle 3

3B. Set Store Cycle 1:

ILD 41	Cycle Trigger	Prevent I-Star RO
B3	1, 2, 3 Sets	Process

4. PREVENT I-RING ADVANCE

4A. Block I-Ring:

ILD 41	Block I-Ring	Store Cycle 1, 2, 3
A3		I-Ring-3 (or 6)

4B. Prevent Reset I-Ring: The I-Ring advances from the reset of one position driving the set of the next. Preventing the reset of a position holds that position set.

ILD 14	Reset	Block I-Ring
A5	I-Ring	

5. SET STORAGE ADDRESS REGISTER

5A. Generate UP Index Address: A B-tag produces similar switching for the four bit only. With an AB tag, only the A-tag switching is used for 0 x 9 address. The second and third cycle switch with store cycle 2 and 3 for the bit combinations.

ILD 42	Adr Reg Set	A-tag
B2	U8 Index	Not Carry Tr and
		Store Cycle 1
A2	U1 Index	
B2	UC Index	

5B. Generate TP Index Address: A B-tag sets the 1-bit and C-bit in addition for a 09 with both B- and AB-tags.

ILD 41	Adr Reg Set	I-Ring-3 (or 6)
C1	T8 Index	Store Cycle 1, 2, 3

5C. Generate HP Index Address:

ILD 41	Adr Reg Set	I-Ring-3 (or 6)
C1	HCZ Index	Store Cycle 1, 2, 3
C1	H8 Index	
C1	H2 Index	
C1	HCD Index	

6. TRANSFER A-STR UP TO A-REG

6A. A-Star UP Gate-out: Digits are read out to an added data path between the address registers and the A-register. Successive cycles read out the tens and hundreds digits for serial adding. The B-Star read-out is similar during I-Ring-6.

ILD 42	Index A-Star	St Cycle 1
A4	Units Gate	I-Ring-3
		Time 000-030

6B. Storage-Index A-register Reset: The normal A-register reset on I-cyles is prevented by the fall of the not-index signal.

ILD 41	St-Index A	St Cycle 1, 2, 3
D5	Reg Reset	I-Ring-3 (or 6)
		Time 000-015

6C. *Set A-reg Latches:* Simultaneously, the remaining bits, including zone, are gated. The set signal drives the latches directly, indexing with other set conditions. The B-Star RO during I-Ring-6 is similar. The tens and hundreds gates cause their read-out in successive store ring cycles. The normal A-register set is prevented by the fall of the not-index signal.

ILD 42	A -reg Set 1	Index A-Star Units Gate
A6		A-Star RO Units 1

7. RI STORAGE ADDRESS TO B-REGISTER

7A. B-register Reset:

ILD 10	B-reg Reset	Activate Storage
C1	Zone and Digit	Time 000-015

7B. Set B-register Latches: The controls for the B-register are part of the normal storage read-out.

ILD 10	B-reg 1	Sense 1
A1		

8. ADVANCE STORE-INDEX RING

8A. Reset Store Cycle 1:

ILD 41	Not St Cycle 1	Cycle Tr 1, 2, 3 Sets
B3		St Cycle 1

8B. Set Store Cycle 2:

ILD 41	St Cycle 2	Not Zone Carry Tr
B3		Not St Cycle 1

9. ARITHMETIC OUTPUT TO MODIFIER BUS

9A. Set Index Arithmetic RO Gate: The control latch remains set until the fall of I-Ring-3 or -6 to allow the transfer of the three digits and carry.

ILD 41 B5	Index 1	St Cycle 1 Not Carry Tr I-Ring-3 (or 6) Time 000-030 Delta Process

9B. Set Zone Adder RO Gate: The zone adder is set for all digits and the carry. The tens digit does not have a zone, but the adder causes no problem.

ILD 41 B5	Index Zone Adder Gate	Index 1

9C. Index Star Bus: All bits are gated through in a similar manner.

ILD 41 A6	Index Star 1	Arith 1 Index 1 Time 090-000

9D. Modifier Bus Mix: Similar mixes occur for each bit to feed the modifier output line. This in turn feeds the A- and B-Star.

ILD 20 A4	1 Bus	Index Star 1

10. RI A-STAR AND/OR B-STAR

10A. A-Star RI: Tens and hundreds positions are gated on successive store cycles.

ILD 41 C5	A-Star RI Units	St Cycle 2 Trigger I-Ring-3 Time 090-000

10B. B-Star RI Control: The B-Star does not read simultaneously on *Move* and *Load* operations. This gating is part of the normal system operation.

ILD 41 B6	B-Star RI Gate	Index 1 I-Ring 6 Time 090-000

10C. B-Star RI: Tens and hundreds positions are gated on successive store cycles.

ILD 41 C6	B-Star RI Units	B-Star RI Gate St Cycle 2 Trigger

11. REPEAT STEPS 4 THROUGH 10 FOR TENS POSITION

Operation is gated with store cycle 2 and 3.

12. REPEAT STEPS 4 THROUGH 7 FOR HUNDREDS POSITION

Operation is gated with store cycle 3.

13. ADVANCE STORE-INDEX RING FOR CYCLE 4

13A. Set Store Cycle 4:

ILD 41 C3	St Cycle 4	I-Ring-3 (or 6) Time 000-015 St Cycle 3 Delta Process

14. REPEAT STEPS 9 AND 10 FOR HUNDREDS POSITION

Operation is gated with store cycle 4.

15. ZONE ADDER CARRY TEST

15A. Set Zone Carry Trigger: With a zone carry, the complete sequence is followed. Without a zone carry, store cycle 1 is not set and steps 15, 16, and 17 are omitted.

ILD 41 A3	Zone Carry Trigger	Store Cycle 3 Store Cycle 4 Tr Zone Adder Carry Time 060-090

15B. Set Store Cycle 1:

ILD 41 B3-B4	Store Cycle 1 Trigger	Not Store Cycle 1 Tr Not Store Cycle 2 Tr Zone Carry Trigger Cycle Tr 1, 2, 3 Sets

15C. Reset Store Cycle 3:

ILD 41 C3	Not Store Cycle 3 Trigger	Store Cycle 3 Tr Cycle Tr 1, 2, 3 Sets

16. SET B-REGISTER FOR ZONE CARRY

16A. Prevent Activate Storage: The B-register is needed for zone carry entry. This gate blocks the storage RO cycle.

ILD 41 A4	Prevent Act Storage	Store Cycle 1 Zone Carry Trigger

16B. Reset B-register for Carry:

ILD 41 A5	B-reg Reset Index	Prevent Act Storage Time 000-015

16C. Insert Zone Bit for Carry:

ILD 41 A4	B-reg Set A	Store Cycle 1 Zone Carry Trigger Time 000-030

17. REPEAT STEPS 6, 9, AND 10 FOR UNITS POSITION ZONE CARRY ENTRY

The zone carry trigger is used to gate the operation.

18. RESET STORE INDEX CONTROLS

18A. Reset Store Cycle 1: If no carry cycle is being taken, the store cycle 3 trigger is reset in the same manner. The store cycle 4 trigger resets automatically in either case at the end of the third cycle.

ILD 41 B3	Not Store Cycle 1 Trigger	Store Cycle 1 Cycle Tr 1, 2, 3 Sets

18B. Reset Tag Latches: The tag latches are reset by an END INDEX OP. This signal is developed by one of two switches depending on the use of the carry cycle.

ILD 41 C4	End Index Op (No Carry)	Store Cycle 4 Tr Not Zone Carry Tr Time 090-000

18C. Reset Zone Carry Trigger:

ILD 41 B3	Not Zone Carry Trigger	Zone Carry Tr Not Store Cycle 1, 2, 3 Time 000-030

19. CONTINUE I-RING ADVANCE

19A. Allow I-Star Gate-Out: This signal is forced until the Tag latches fail to prevent I-Star RO.

ILD 41 A2	Not Index	End Index Op

19B. Advance I-Ring: The advance of the ring releases the remainder of the interlocks.

ILD 41 A3	Not Block I-Ring	Not Store Cycle 1, 2, 3 Not Q Op 2

19C. Not Index Interlocks:

ILD 41 B5	Not Index 1	Not I-Ring-3 (or 6)

Store B-Star Operation (Timing Chart — Figure 8)

The address left in the B-address register from the previous operation is stored in the specified A-address. Following a branch operation with branch specified, the I-address appears in the B-address register. The address must be stored on the next operation or it is lost.

Objectives (Store B-Star Operation)

1. Operation Code Development
   - A. H Operation
   - B. Prevent B-Star RI
2. Complete Instruction RI
3. Initiate A-Cycles
   - A. Set Delta A-Cycle
   - B. (Not) Store
   - C. Prevent Delta A-Cycle Reset
4. Start Store-Index Ring
   - A. Gate Store Cycle 1
   - B. Set Store Cycle 1
5. RO and Update A-Star
   - A. A-Star RO to Star
   - B. Modifier Control Minus 1
   - C. A-Star RI
6. B-Register RI Storage Character
   - A. Normal B-register RI
   - B. Prevent Regen Inhibit Control
7. B-Star Digit RO to A-Register
   - A. B-Star UP Gate-Out
   - B. Store-Index A-register Reset
   - C. Set A-register Latches
8. Transfer A-Register to Storage
   - A. A-register Store Inhibit
   - B. A-register Inhibit Entry
9. Advance Store-Index Ring
   - A. Reset Store Cycle 1
   - B. Set Store Cycle 2
10. Repeat steps 5 through 9 for Tens Position
11. Repeat steps 5 through 8 for Hundreds Position
12. Force I-E Change
    - A. Store I-E Change
    - B. Delta-I set Q-H
    - C. Reset Delta-A Latch

Logic Detail (Store B-Star Operation)

1. OPERATION CODE DEVELOPMENT

1A. H Operation: This is a normal operation code development. The signal is required immediately to stop B-Star reading of the A-address. This operation mixes with the Q-operation for a Store Opr Q-H gate ILD 41.

ILD 41 A1	H Operation	Op Decode 8 Not 421 Op Decode AB

1B. Prevent B-Star RI: This signal is the same as that developed for the move and load operations. It serves to prevent the B-address register reading when the previous address is required.

ILD 17 C3	(Not) Prevent B-Address RI	St Opr (Q or) H

2. COMPLETE INSTRUCTION RI

The instruction cycle is completed in the normal manner to the I-E Change. No B-address or modifier character is used. The A-address is the store location, while the B-Star contains the address to be stored.

3. INITIATE A-CYCLES

3A. Set Delta A-Cycle:

ILD 13 B4	Delta A Latch	I-Cycle Latch I-E Change Not A-Cycle Elim

3B. (Not) Store: The not-store signal is an inversion of the operation gate, which serves to prevent normal operation. One of its major functions is to hold the operation in A-cycles.

ILD 41 D5	Not Store (inverted)	St Opr Q-H A-Cycle

3C. Prevent Delta A-Cycle Reset: Resetting the Delta A-cycle sets the Delta B-cycle, which is not desired for the store operations.

ILD 13 C4	(Not) Reset Delta A-Cycle Latch	(Not) Store

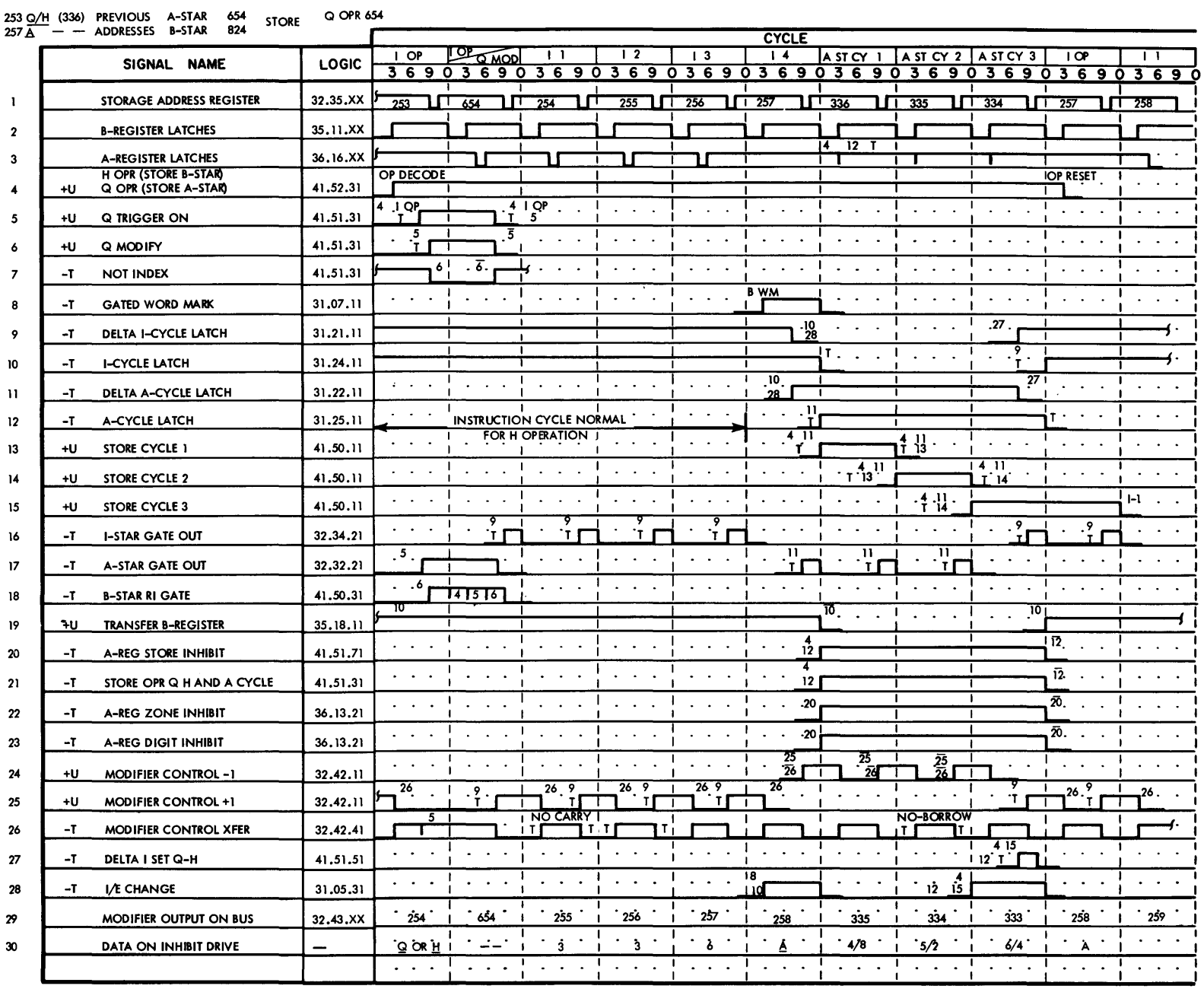


Figure 8. Store A-Star—Q(AAA) and Store B-Star—H(AAA)

4. START STORE-INDEX RING

4A. Gate Store Cycle 1:

ILD 41	Store Cycle 1	Not St Cy 1
B4	Gate	Not St Cy 2
		Not St Cy 3

5. RO AND UPDATE A-STAR

5A. A-Star RO to Star: The A-Star read-out is a normal function for each A-cycle.

ILD 17	A-Star Gate	Delta A-Cycle
B1	Out	A-Star Gate Out Calc
		Time 090-000

5B. Modifier Control −1: This is the normal switching for forward scan.

ILD 19	Modifier Control	Not Mod Ctrl 1
D3	Minus 1	Not Mod Ctrl TRF

5C. A-Star RI: Tens and Hundreds position follow in time succession. This is the normal A-cycle control.

ILD 17	A-Star RI	Delta A-Cycle
A6	Units	Time 000-030

6. B-REGISTER RI STORAGE CHARACTER

6A. Normal B-register RI: The latches are reset and set in the normal manner.

ILD 10

6B. Prevent Regen Inhibit Control: Regeneration is normally automatic on A-cycles. The store operation prevents the gating.

ILD 11	(Not) Transfer	(Not) Store
C1	B-register	

7. B-STAR DIGIT RO TO A-REGISTER

7A. B-Star UP Gate-Out: The digits are read out over the added data path to the A-register. Successive cycles read out the tens and hundreds digits for serial transfer.

ILD 42	Index B-Star	St Cycle 1
A4	Units Gate	St Opr Q-H
		A-cycle

7B. Store-Index A-register Reset:

ILD 41	St-Index A-Reg	St Opr Q-H
D5	Reset	A-cycle
		Time 030-060

7C. Set A-register latches: Simulteneously, the remaining bits are gated. The normal A-register set is blocked by the store signal on ILD 11. The tens and hundreds positions are similarly gated on store cycles 2 and 3.

ILD 42	A-Reg Set-1	Index B-Star Units Gate
A6		B-Star RO units I

8. TRANSFER A-REGISTER TO STORAGE

8A. A-register Store Inhibit: This signal mixes with those of other operations to develop the A-register digit inhibit and zone inhibit signals.

ILD 41	A-reg St Inhibit	St Opr Q-H
B2		A-cycle

8B. A-register Inhibit Entry: The remaining bits are similarly gated.

ILD 11	Inhibit Drive 1	A-reg-1
C3		A-reg Dig Inh

9. ADVANCE STORE-INDEX RING

9A. Reset Store Cycle 1:

ILD 41	Not St Cycle 1	Cycle Trig 1, 2, 3 Sets
B3	Trigger	St Cycle 1

9B. Set Store Cycle 2:

ILD 41	St Cycle 2	St Cycle 1
B3	Trigger	Not St Cycle 1

10. REPEAT STEPS 5 THROUGH 9 FOR TENS POSITION

The operation is gated with store cycle 2.

11. REPEAT STEPS 5 THROUGH 8 FOR HUNDREDS POSITION

The operation is gated with store cycle 3.

12. FORCE I-E CHANGE

12A. Store I-E Change: The I-E change must be forced because the operation takes no B-cycles.

ILD 41	I-E	St Cycle 3
D5	Change	St Opr Q-H

12B. Set Delta-I Latch: The normal circuit to set the Delta-I latch is gated with B-cycle. Since no B-cycles are used in this operation, the set must be forced.

ILD 41	Delta-I Set	St Opr Q-H
D5	Q-H	St Cycle 3
		Time 075-105

12C. Reset Delta-A Latch: This signal is mixed with the A-cycle reset but not in the B-cycle latch set.

ILD 41	Reset Delta A	Delta-I set Q-H
D5	Latch	

Store A-Star Operation (Timing Chart — Figure 8)

The address left in the A-address register from the previous operation is stored in the new specified A-address. The address must be stored on the next operation or it is lost. The A-address is transferred to the B-address register before the new read-in.

Objectives (Store A-Star Operation)

1. Operation Code Development
   - A. Q-Operation
   - B. Inhibit B-Star RI
   - C. Set Q Modify Trigger
   - D. Set Q Modify Latch
2. Prevent I-Address Register Gate-Out
   - A. Not Index
   - B. Prevent I-Star Gate-Out
3. Prevent Op Register Change
   - A. Prevent Op Reg Reset
   - B. Prevent Op Reg Read-in
4. Prevent I-Ring Advance
   - A. Not Index
   - B. Prevent Reset I-Ring
5. A-Star Transfer to B-Star
   - A. A-Star RO to Star
   - B. Modifier Control Transfer
   - C. Prevent I-Star RI
   - D. B-Star RI
6. B-Register RI and Regen Storage Position
   - A. Normal B-reg RI
   - B. Normal B-reg Inhibit Control
7. Complete Instruction RI
   - A. Not Index
8. Execute cycle same as Operation H from step 3 to end.

Logic Detail (Store A-Star Operation)

1. OPERATION CODE DEVELOPMENT

1A. Q-Operation: This is a normal operation code development. The signal is required immediately to stop further reading until the A-Star is transferred to the B-Star. It also prevents the B-Star from reading in the new A-address. The operation mixes with the H-operation for a store operation Q-H gate (Logic 41.51.31.2).

ILD 41 A1	Q Operation	Op Decode B Not A Op Decode 8 Not 421

1B. Inhibit B-Star RI: This signal is the same as that developed for the move and load operations. It serves to prevent the B-address register reading, when the previous address is required.

ILD 17 C3	(Not) Prevent B Address RI	St Opr Q (or H)

1C. Set Q Modify Trigger: The Q modify gate is used to gate the A-Star to B-Star transfer. It also forces the index interlock controls to prevent normal I-operations. The trigger resets at time 075-105 of the next cycle.

ILD 41 A2	Q Trigger On	Q-Opr I-Ring Op Delta Process Time 075-105

1D. Set Q Modify Latch: The Q Modify latch is used to provide a delay gate to control the address transfer. The trigger output is used for the early controls. The latch resets at 090-000 following the trigger reset.

ILD 41 A2	Q Modify	Q Trigger ON Time 090-000

2. PREVENT I-ADDRESS REGISTER GATE-OUT

2A. Not Index: The not-index signal is used for normal gating. Index prevents these gatings for its duration, or one cycle.

ILD 41 A2	(Not) Index	Not Prevent I-Star RO (Not) Q Modify

2B. Prevent I-Star Gate-Out: This signal, when powered, normally gates the I-Star to the storage address register.

ILD 18 A1	Not I-Star Gate-Out	(Not) Index

3. PREVENT OP REGISTER CHANGE

3A. Prevent Op Reg Reset:

ILD 15 A1	I-Ring Op (Not) Q Modify	(Not) Q Modify I-Ring Op

3B. Prevent Op Reg Read-in:

ILD 15 A2	(Not) Op Reg Set	I-Ring Op (Not) Q Mod

4. PREVENT I-RING ADVANCE

4A. Not Index: This signal for store operations is similar to that in 2A. Two signals are used to satisfy index operations. The index gates are used because the same conditions are required.

ILD 41 A3	Block I-Ring	Q Modify

4B. Prevent Reset I-Ring: The I-Ring advances from the reset of one position driving the set of the next. Preventing the reset holds the I-Ring in the position set.

ILD 14 A5	Not Reset I-Ring	Block I-Ring

5. A-STAR TRANSFER TO B-STAR

5A. A-Star RO to Star:

ILD 41 A2	A-Star RO	Q Trigger ON

5B. Modifier Control Transfer:

ILD 41 A2	Transfer Star	Q Modify

5C. Prevent I-Star RI:

ILD 18 A6	Not I-Star RI Units	(Not) Index

5D. B-Star RI:

ILD 41 A2	B-Star RI Transfer	Q Modify

6. B-REGISTER RI AND REGEN STORAGE POSITION

6A. Normal B-register RI: The latches are reset and set in the normal manner (ILD 10).

6B. Normal B-register Inhibit Control: Regeneration of the storage position uses the normal I-cycle control (ILD 11).

7. COMPLETE INSTRUCTION RI

7A. Not Index: The index interlocks end with the reset of the Q Modify trigger at time 075 and 090. With it, the I-Ring is allowed to reset the Op position and advance; the I-Star reads out for the next storage address.

8. EXECUTE CYCLE SAME AS OPERATION H

From step 3 to end.

The Q and H operations differ only in that the A-Star must be placed in the B-Star. Both operations store the contents of the B-Star. A B-address in the instruction would read into the B-Star destroying the previous address.

I-B Star Transfer Operation (Timing Chart — Figure 9)

At the time of a branch condition, the I-address is dropped in favor of the A-address. In order that the program may return to the point of branch, the I-address is transferred to the B-address register prior to read-in. The first instruction of the branch routine must be store B-Star (H Operation).

Objectives (I-B Star Transfer Operation)

1. On Branch Condtion Gate
   - A. Sense Branch Condition
   - B. Set I-A Star Gate
   - C. Complete Operation
2. Set I-B Star Control
   - A. Set I-B Star 1 Trigger
   - B. Prevent I-Ring Op
   - C. Set I-B Star 2 Latch
3. Transfer I-Star to B-Star
   - A. Prevent A-Star Gate-Out
   - B. I-Star RO
   - C. Modifier Control Transfer
   - D. B-Star RI
4. B-Reg RI and Regen Storage Position
   - A. Normal B-reg RI
   - B. Normal Regen Inhibit Control
5. Continue Normal I-Cycle
   - A. Set I-Ring Op

Logic Detail (I-B Star Transfer Operation)

1. OP BRANCH CONDITION GATE

1A. Sense Branch Operation: In normal branch operation, if the condition is present, it is gated with the necessary operation and modifier decoding to set the PROGRAM SKIP latch. If the condition is not present, the operation advances to the next step with the I-address.

1B. Set I-A Star Gate: This gate comes directly from the PROGRAM SKIP latch (Logic 31.08.11.2). It serves to prevent the I-Star read-out, and forces the next instruction.

1C. Complete Operation: Branch operations vary in length. The setting of the PROGRAM SKIP latch forces the SET I-RING OP latch for the next instruction. If extra cycles are required, the set of the I-Ring is prevented until after completion (Logic 31.30.11.2).

2. SET I-B STAR CONTROL

2A. Set I-B Star 1 Trigger: The set is gated with a Not I-B Star 2, which serves to prevent a second cycle set. Its output prevents the set of I-Ring Op and the read-out of A-star until after the I-B Star transfer. It also gates the transfer. Reset occurs at the next time 075.

ILD 41 C3	I-B Star 1	I-A Star Gate Time 075-105 All Scan Comp

2B. Set I-B Star Latch: This latch provides an inhibit gated word-mark signalled, and prevents a second setting of I-B Star 1. The latch remains set until the I-Ring starts.

ILD 41 D3	I-B Star Intlk	I-B Star 1 Time 000-030 Delta Process

2C. Prevent I-Ring Op:

ILD 14 A5	Not Turn On I-OP	(Not) I-B Star 1

3. TRANSFER I-STAR TO B-STAR

3A. Prevent A-Star Gate-Out:

ILD 17 B1	Not A-Star Gate-Out	(Not) I-B Star 1

3B. I-Star RO to Star:

ILD 41 D3	I-Star RO	I-B Star 1

3C. Modifier Control Transfer:

ILD 19 D2	Mod Ctrl Transfer	I-B Star 2

3D. B-Star RI: The tens and hundreds positions are similarly gated in serial order. The I-Star is also read in at this time from its normal I-cycle gating.

ILD 41	B-Star RI	I-B Star 1
C5	Units	Time 000-030

4. B-REGISTER RI AND REGEN STORAGE POSITION

4A. Normal B-register RI: The latches are reset and set in the normal manner (ILD 10).

4B. Normal B-Reg Inhibit Control: Regeneration of the storage position uses the normal I-cycle control (ILD 11).

5. CONTINUE NORMAL I-CYCLE

5A. Set I-Ring Op: The I-Star address at the branching point is now in the B-address register. To save this address, the first instruction of the branch routine (that indicated by the A-address) must be an operation H.

ILD 14	Turn On	Delta I-cycle
A5	I-Op	Not I-B Star 1
		Time 015-030

Move Record Operation (Timing Chart — Figure 10)

The move record operation is similar to a normal *move* operation. The character from the specified A-address is moved to the specified B-address. The operation

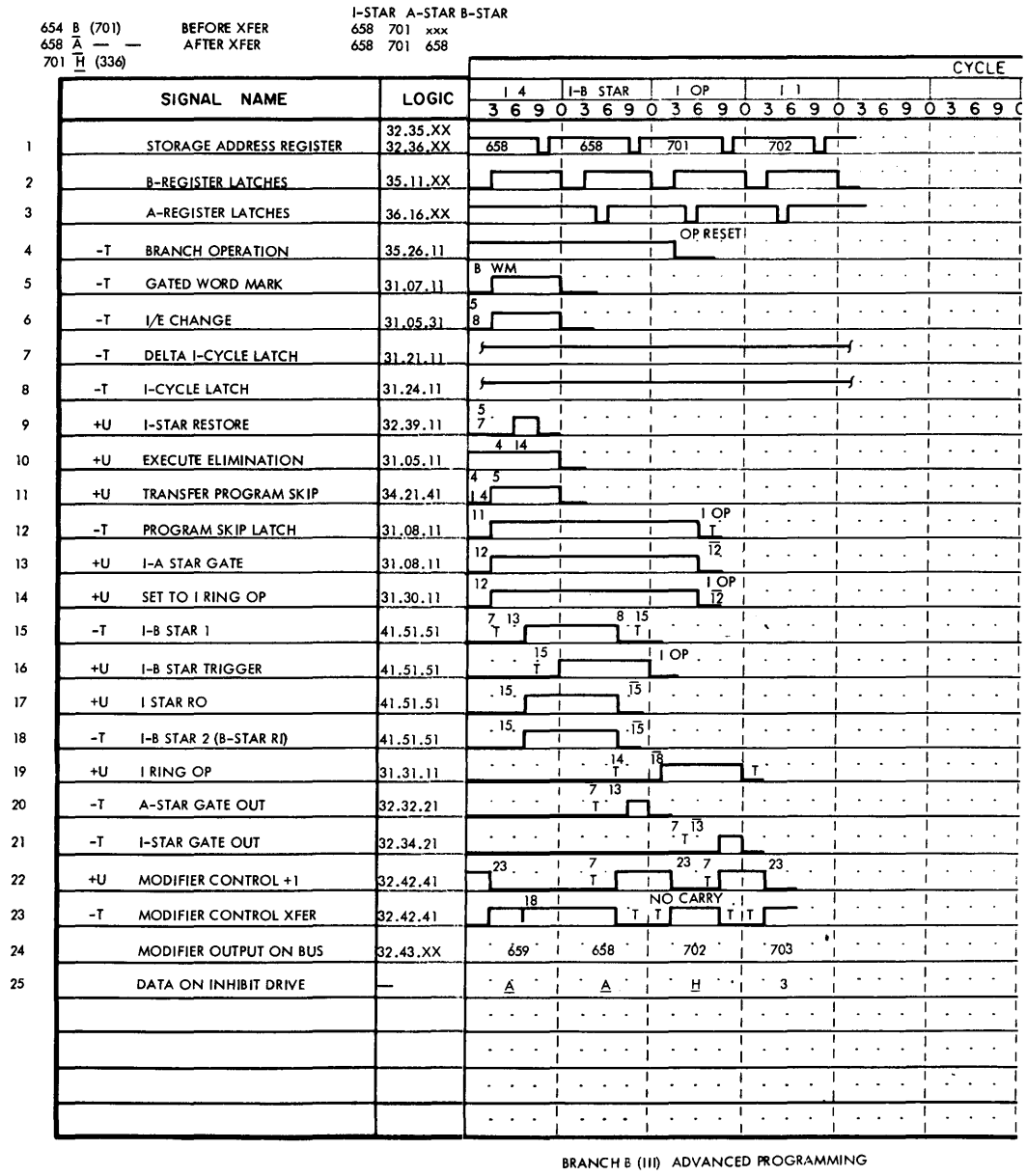


Figure 9. Branch B(III) Advanced Programming (I-B Star Transfer)

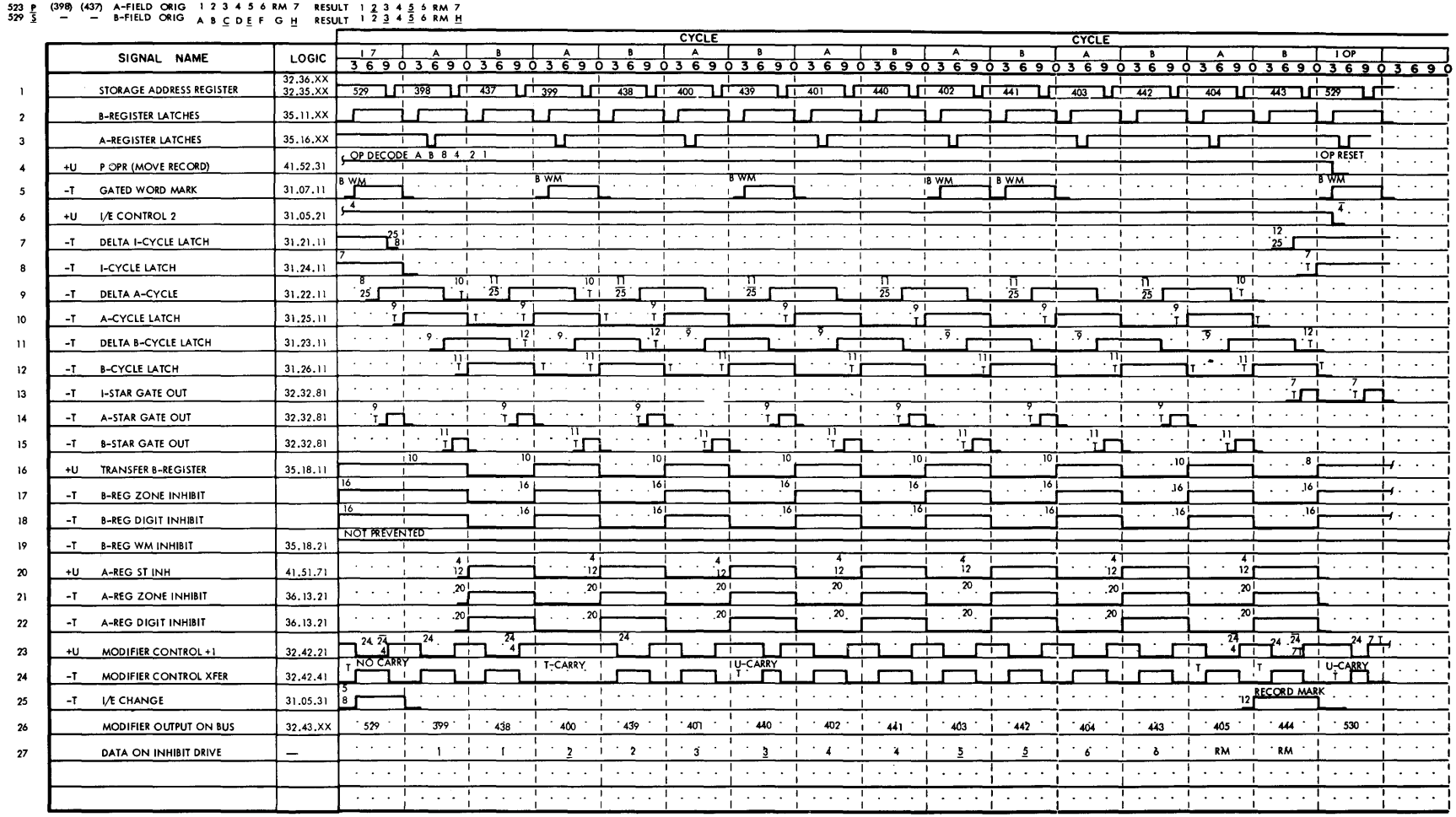


Figure 10. Move Record—P(AAA)(BBB)

differs in that word-mark control is suppressed, and the fields are reverse scanned (high-to-low order). The operation is terminated when either a record-mark or a group-mark with word-mark is detected in the A-field.

Objectives (Move Record Operation)

1. Operation Code Development
   - A. P-Operation
   - B. Prevent Normal I-E Change
   - C. Block A-reg Word Mark
   - D. +1 Modifier Control
2. Normal Instruction RI
3. Set A-Cycle
   - A. Set Delta A-cycle
   - B. Set A-cycle Latch
4. RO and Update A-Star
   - A. A-Star RO to Star
   - B. One-up Address in Modifier
   - C. A-Star RI
5. A-Register RI Storage Position
   - A. Normal B-reg RI
   - B. Normal Regen Inhibit Control
   - C. A-reg Reset Drive
   - D. A-reg Set Control
6. Set B-Cycle
   - A. Reset Delta A-cycle
   - B. Set Delta B-cycle
   - C. Set B-cycle Latch
7. RO and Up-date B-Star
   - A. B-Star RO to Star
   - B. One-up Address in Modifier
   - C. B-Star RI
8. B-Register RI Storage Position
   - A. Normal B-reg RI
   - B. Prevent Normal Regen Inhibit
9. Transfer A-Register to Storage
   - A. A-reg Store Inhibit
   - B. A-reg Inhibit Entry
10. Set A-Cycle
    - A. Reset Delta B-cycle
    - B. Set Delta A-cycle
    - C. Set A-cycle Latch
11. Repeat steps 4 through 11 to Record-Mark
12. Sense Record-Mark for I-E Change
    - A. Sense Record-Mark
    - B. Sense Group-Mark with Word-Mark

Logic Detail (Move Record Operation)

1. OPERATION CODE DEVELOPMENT

1A. P-Operation:

ILD 41 A1	P-Operation	Op Decode B Not A Op Decode 421 Not 8

1B. Prevent Normal I-E Change: The I-E change normally occurs when a B-register word-mark is detected. In order to continue the transfer, the normal I-E Change is prevented.

ILD 41 B1	Prev Norm I-E Change	P-Operation

1C. Block A-reg Word-Mark: This prevents the A-field word-mark from stopping the operation.

ILD 13 D1	Not Set A WM Latch	(Not) P-Operation

1D. Plus 1 Modifier Control: The P-operation requires a reverse scan.

ILD 41 A1	+1 Modifier Ctrl	P-Operation

2. NORMAL INSTRUCTION RI

The instruction includes both A-address and B-address but no modifier. Both addresses must be included because the previous B-address is not saved for chaining purposes.

3. SET A-CYCLE

3A. Set Delta A-cycle:

ILD 13 B4	Set Delta A Latch	I-Cycle I-E Change Not A-Cy Elim

3B. Set A-cycle Latch:

ILD 13 B5	A-cycle Latch	Delta A-cycle Time 000-060

4. RO AND UPDATE A-STAR

4A. A-Star RO to Star:

ILD 17 B1	A-Star Gate-Out	Delta A-Cycle A-Star Gate-Out Calc Time 090-000

4B. One-Up Address in Modifier: The signal developed in *1D* forces the modifier gating to plus one.

4C. A-Star RI: Tens and hundreds position follow serially using A-cycle for gating.

ILD 17 A6	A-Star Units RI	Delta A-Cycle Time 000-030

5. A-REGISTER RI STORAGE POSITION

5A. Normal B-reg RI: The latches are reset and set in the normal manner (ILD 10).

5B. Normal B-reg Inhibit Control: Regeneration of the storage position uses the normal A-cycle control (ILD 11).

5C. A-register Reset Drive:

ILD 11 B1	A-reg Reset	A-cycle Time 045-060

5D. A-register Set Control:

ILD 11 A1	A-reg Set	A-cycle Not Store Time 045-075

6. SET B-CYCLE

6A. Reset Delta A-cycle:

ILD 13 C4	Reset Delta A-cycle	A-cycle Latch Not Store Time 075-105

6B. Set Delta B-cycle Latch:

ILD 13 C4	Delta B-Latch	Reset Delta A-cycle

6C. Set B-cycle Latch:

ILD 13 C5	B-cycle Latch	Delta B-cycle Time 000-060

7. RO AND UPDATE B-STAR

7A. B-Star RO to Star:

ILD 17 C1	B-Star Gate-Out	Delta B-cycle Time 090-000 B-Star Gate-Out Calc

7B. One-Up Address in modifier: The signal developed in *1D* forces the modifier gating to plus one.

7C. B-Star RI: Tens and hundreds position follow serially using B-cycle for gating.

ILD 17 C6	B-Star RI Units	Delta B-cycle B-Star Gate-in U Calc Time 000-030

8. B-REGISTER RI STORAGE POSITION

8A. Normal B-register RI: The latches are reset and set in the normal manner (ILD 10).

8B. Prevent Normal Regen Inhibit: No normal inhibit switching for regeneration is provided on B-cycles. Individual controls for operations are shown on ILD 11.

9. TRANSFER A-REGISTER TO STORAGE

9A. A-reg Store Inhibit: This signal is so named because of its mixing with control gates used in store operations. These signals force the A-register digit and zone inhibit gates.

ILD 41 B1	A-reg Store Inhibit	P-Operation B-cycle

9B. A-register Inhibit Entry: The remaining bits gate in a similar manner to return the character minus word-mark to storage. The B-register word-mark is retained, if present.

ILD 10 A3	Inhibit Drive 1	A-reg Digit Inh A-reg 1

10. SET A-CYCLE

10A. Reset Delta B-cycle:

ILD 13 C4	Reset Delta B Latch	B-cycle Latch Not A-cycle Elim Time 075-105

10B. Set Delta A-cycle: A forced I-E change occurs at the end of the transfer to prevent an additional set.

ILD 13 B4	Delta-A Latch	Reset Delta B-Latch Not I-E Change

10C. Set A-cycle Latch:

ILD 13 B5	A-cycle Latch	Delta A-cycle Time 000-060

11. REPEAT STEPS 4 THROUGH 11 TO RECORD-MARK.

The cycle is repeated with both A- and B-addresses advancing until a record-mark is detected. The mark forces an I-E Change to end the operation.

12. SENSE RECORD-MARK FOR I-E CHANGE

12A. Sense Record-Mark:

ILD 41 B1	I-E Change	P-Operation B-cycle A-reg 2 Not 1 A-reg 8 Not 4 A-reg A Not B

12B. Sense Group-Mark with Word-Mark:

ILD 41 A1	I-E Change	P-Operation A-reg 21 A-reg 84 A-reg AB A-reg WM B-cycle

Expanded Print Edit

The basic operations of the *edit* instruction can be expanded by including the four *expanded edit optional features.*

Asterisk Protection

This optional feature permits asterisks to appear at the left of significant digits. The control word (Figure 11) is written with the asterisk at the left of the *zero-suppression zero.* On the *forward scan,* the normal editing process proceeds until the asterisk is sensed and replaced in the edited word, with the corresponding A-field character (in the example: 9). The forward-scan edit then continues normally until the B-field WM is sensed and removed.

On the *reverse scan,* zeros, blanks, and punctuation to the left of the first significant digit are replaced by asterisks (Figure 11, *Results of Edit*). The operation ends when the WM (below the 4, Figure 11), set during the forward scan, is erased.

CIRCUIT LOGIC (FIGURE 12, 34.11.21.2)

The BCD code lines for the asterisk character (4, 8, and B) are switched with BODY (since this feature controls asterisks in only the *body* portion of the word), and the switch output in turn switches with EDIT OPR to set the ASTERISK latch. These (asterisk code) lines are also switched to produce the REGEN A-REG signal to the dollar-option circuits. The ASTERISK latch output is also switched with B-REG BLANK or

A-field data	00359426
B-field control word	bbb,b*0.bb
Forward scan	003,594.26
Reverse scan	**3,594.26
Results of edit	**3,594.26

ASTERISK PROTECTION

A-field data	00359426
B-field control word	bbb,b$0.bb
First forward scan	003,594.26
Reverse scan	bb3,594.26
Second forward scan	$3,594.26
Results of edit	$3,594.26

FLOATING DOLLAR SIGN

A-field data	0359426-
B-field control word	CRbb,bb0.bb
Forward scan	CR03,594.26
Reverse scan	CRb3,594.26
Results of edit	CR 3,594.26

SIGN CONTROL LEFT

1.	A-field data	00000
	B-field control word	bbbb0
	First forward scan	000.00
	Reverse scan	bbb.00
	Second forward scan	bbb
	Results of edit	(Blank Field)
2.	A-field data	59426
	B-field control word	bbb.b0
	(First) forward scan	594.26
	Reverse scan	594.26
	Results of edit	594.26
3.	A-field data	00001
	B-field control word	bbb.b0
	(First) forward scan	000.01
	Reverse scan	bbb.01
	Results of edit	.01

THREE EXAMPLES OF DECIMAL CONTROL

Figure 11. Examples of Expanded Edit

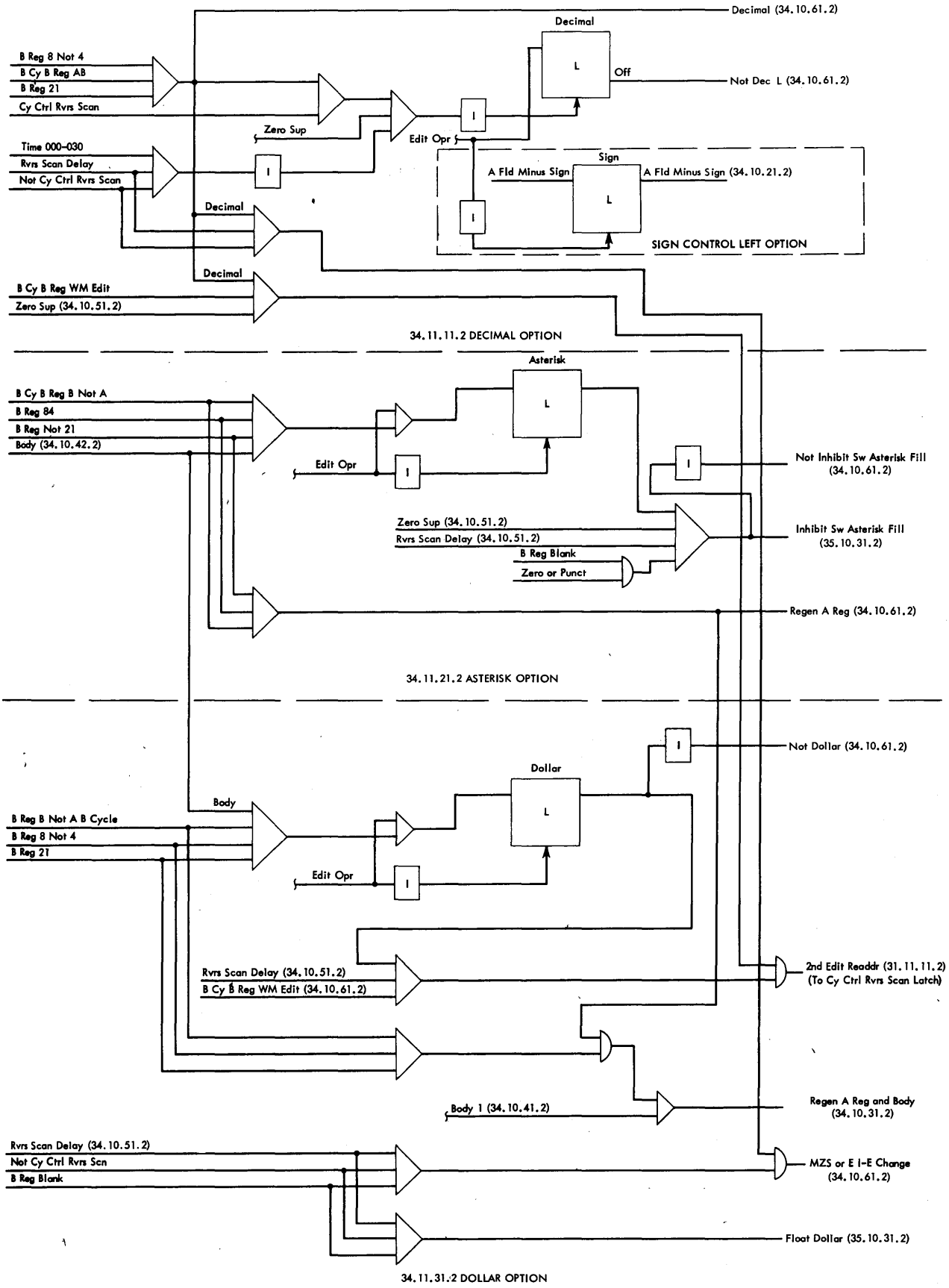


Figure 12. Expanded Print Edit Logic

ZERO OR PUNCT and RVRS SCAN DELAY and ZERO SUP to produce the INHIBIT SW ASTERISK FILL signal. This output holds down the other-edit line (Logic 35.40.11.2) to prevent normal regeneration of the B-register during the zero suppress–reverse scan operation. INHIBIT SWITCH ASTERISK FILL also is switched with NOT BLOCK OPR and B-CYCLE (Logic 35.44.11.2) to bring up the 4, 8, B lines to storage and the inhibit C-digit line to the C-bit generator (Logic 35.48.11.2), where it prevents the inhibit Drive-C line to storage from coming up while asterisks are being printed. NOT INHIBIT SW ASTERISK FILL, when down, prevents EDIT REGEN C-BIT (Logic 34.10.51.2) from coming up and reading out a C-bit from storage (Logic 35.45.11.2).

Floating Dollar Sign

Under normal edit and zero-suppress conditions, an A-field data word such as 00359426 may be printed out as $ 3594.26. The *floating dollar* optional feature *floats* the dollar sign to the right (in this example, two positions) to print out at the left of the high-order significant digit ($3594.26, in the example). The control word (Figure 11) is written with the $ at the left of the *zero-suppression zero*. As Figure 11 shows, three scans are required to complete this editing operation. On the *first forward scan*, the editing proceeds until the $ is replaced in the edited word with the corresponding A-field character. Forward-scan edit continues until the B-field WM is sensed and removed. On the reverse-scan, zeros and punctuation to the left of the first significant digit are replaced by blanks until the WM (set during the first forward scan) causes the *second forward scan* to start. The WM is erased and the scan continues until the $ replaces the first blank position sensed, when the operation is stopped.

CIRCUIT LOGIC (FIGURE 12, 34.11.31.2)

The $ BCD code lines (1, 2, 3, and B) are switched with BODY; this switch output switches with EDIT OPR to turn on the DOLLAR latch. These ($ code) lines are also switched, and the switch output mixed with REGEN A-REG from the asterisk-option circuits. The common line is switched with BODY to bring up the regen-and-body line. This line brings up the blank or zero line (Logic 34.10.31.2) which controls set and reset of the BODY latch and regeneration of the A-register. The DOLLAR latch output is switched with RVRS SCAN DELAY and B-CY B-REG WM EDIT to bring up 2nd EDIT READR, which resets the CYCLE CONTROL REVERSE SCAN latch, ending the reverse scan at 105 time, after the B-register WM is sensed. The NOT CY CTRL RVRS SCN line, thus brought up, is switched with RVRS SCAN DELAY and B-REG BLANK to bring up the MZS or E I-E change and float dollar lines. Float dollar acts similarly to INHIBIT SW ASTERISK FILL to prevent normal B-register regeneration during the second forward scan, and to bring up the $ (1, 2, 8, and B) inhibit lines to storage. The absence of C-Zone, C-Digit, and C-WM in the C-Bit Generator (Logic 35.48.11.2) brings up the necessary inhibit drive C-line to storage. NOT-DOLLAR, down when the DOLLAR latch is ON, prevents the MZS or E I-E change line (ILD 39) from coming up when the DOLLAR latch comes ON. However, the MZS or I-E change line of Figure 12 comes up at 105 time, causing the cycle-control I-E change line to come up and stop the operation when the first blank is sensed on the second forward scan.

Sign Control Left

This optional feature permits printing of a CR or − symbol to the left of a negative field. The CR or − symbol is written in the high-order position of the control word (Figure 11). The normal *forward scan* edit proceeds until the CR or − symbol is reached. If the A-field sign is minus, the symbol is printed out; if plus, it is blanked. The *reverse scan* proceeds as any edit, zero-suppress operation, ending at the WM set on the forward scan.

CIRCUIT LOGIC (FIGURE 12, 34.11.11.2)

The SIGN latch is set by, and holds, the A-field minus sign so that the inversion of its output can control the edit regen C-bit line, which controls the Inhibit C-digit and zone lines (Logic 35.45.11.2). These lines, when up, bring up the inhibit drive-C line for validity-checking purposes. Since the other-edit line (Logic 35.40.11.2) is down, no B-register regeneration occurs, and CR or − is blanked.

Decimal Control

This optional feature eliminates the decimal point if the A-field data contains all zeros. When this occurs, three scans are required; if at least one significant digit is present, only two scans are needed (Figure 11). Normal editing with zero suppression occurs after the forward and reverse scans. However, if the data word contains no significant digits, a second forward scan replaces the zeros to the right of the decimal point, and also the decimal point, with blanks. The operation stops at the decimal position.

CIRCUIT LOGIC (FIGURE 12, 34.11.11.2)

The DECIMAL latch is set ON by EDIT OPR. The BCD decimal code lines (1, 2, 8, A, and B) are switched with CY CTRL RVRS SCAN to control the decimal line.

This line is switched with ZERO SUP and the inverted output of a timing switch to control the DECIMAL latch reset. The OFF latch output, NOT DEC L, prevents an edit I-E change when the latch is ON, and permits EDIT REGEN C-BIT (Logic 34.10.51.2) to block B-register regeneration (via the other-edit line, Logic 35.40.-11.2), and bring up the inhibit drive C-line (Logic 35.48.11.2) to satisfy parity requirements. When the decimal is encountered in the reverse scan, the decimal line comes up. This line is switched with B-CY B-REG WM EDIT and ZERO SUP to bring up 2nd EDIT READR. This line resets the CYCLE CONTROL REVERSE SCAN latch, ending the reverse scan at 105 time, after the B-register WM is sensed. When the decimal line comes up again on the second forward scan, it is switched with RVRS SCAN DELAY and NOT CY CTRL RVRS SCAN to bring up the MZS or E I-E change line which causes CYCLE CONTROL to end the operation.

High-Low-Equal Alphamerical Compare and Test

The compare operation compares A-field characters with B-field characters beginning with the units position of the field. The sequence of compare is shown in Figure 13. The results of the comparison set two latches either ON or OFF. The basic machine compare circuits cause the setting of the EQUAL latch ON to indicate equal (AAA) and (BBB) fields, or OFF to indicate unequal fields. The optional high-low-equal alphamerical compare feature causes the setting of the HIGH latch ON if the (BBB) field is greater than the (AAA) field, or OFF if the (BBB) field is less than the (AAA) field. Both the EQUAL and HIGH latches remain ON, if set, until the next compare operation. In the meantime, the instruction, TEST AND TRANSFER PROGRAM, B(AAA)d may be used to interrogate the condition of the EQUAL and HIGH latches to determine whether to branch to (AAA) or not. The digit (d) in the test instruction specifies the type of test to be made:

d-character	*Transfer to address (AAA) if*
S	B = A Equal Compare
T	B < A Low Compare
U	B > A High Compare

High-Low-Equal Alphamerical Compare — C (AAA) (BBB) — Circuit Operation (Figure 14)

As in the usual compare operation, the EQUAL latch is turned ON at I-Ring-2 time, and turned OFF at B-cycle 060 time, by the first unequal comparison of the A- and B-registers. The output of the A- and B-registers

		ALPHA NUMERICAL C	B	A	8	4	2	1
9	9	X			X			X
8	8				X			
7	7					X	X	X
6	6	X				X	X	
5	5	X				X		X
4	4					X		
3	3	X					X	X
2	2						X	
1	1							X
0	0	X			X		X	
Z	0-9			X	X			X
Y	0-8	X		X	X			
X	0-7	X		X		X	X	X
W	0-6			X		X	X	
V	0-5			X		X		X
U	0-4	X		X		X		
T	0-3			X			X	X
S	0-2	X		X			X	
≠ (RM)	0-2-8			X	X		X	
R	11-9		X		X			X
Q	11-8	X	X		X			
P	11-7	X	X			X	X	X
HIGH O	11-6		X			X	X	
N	11-5		X			X		X
M	11-4	X	X			X		
L	11-3		X				X	X
K	11-2	X	X				X	
J	11-1	X	X					X
LOW 0	11-0		X		X		X	
I	12-9	X	X	X	X			X
H	12-8		X	X	X			
G	12-7		X	X		X	X	X
F	12-6	X	X	X		X	X	
E	12-5	X	X	X		X		X
D	12-4		X	X		X		
C	12-3	X	X	X			X	X
B	12-2		X	X			X	
A	12-1		X	X				X
+ 0	12-0	X	X	X	X		X	
SPECIAL CHARACTERS								
¬√‾ (TM)	7-8	X			X	X	X	X
‖	6-8				X	X	X	
:	5-8				X	X		X
@	4-8	X			X	X		
#	3-8				X		X	X
+	2-8			X				
!	0-7-8			X	X	X	X	X
'	0-6-8	X		X	X	X	X	
= (WS)	0-5-8	X		X	X	X		X
%	0-4-8			X	X	X		
,	0-3-8	X		X	X		X	X
/	0-1	X		X				X
-	11		X					
Δ	11-7-8		X		X	X	X	X
;	11-6-8	X	X		X	X	X	
)	11-5-8	X	X		X	X		X
*	11-4-8		X		X	X		
$	11-3-8	X	X		X		X	X
&	12	X	X	X				
≒ (GM)	12-7-8	X	X	X	X	X	X	X
?	12-6-8		X	X	X	X	X	
(	12-5-8		X	X	X	X		X
口	12-4-8	X	X	X	X	X		
.	12-3-8		X	X	X		X	X
Blank	No Punches	X						

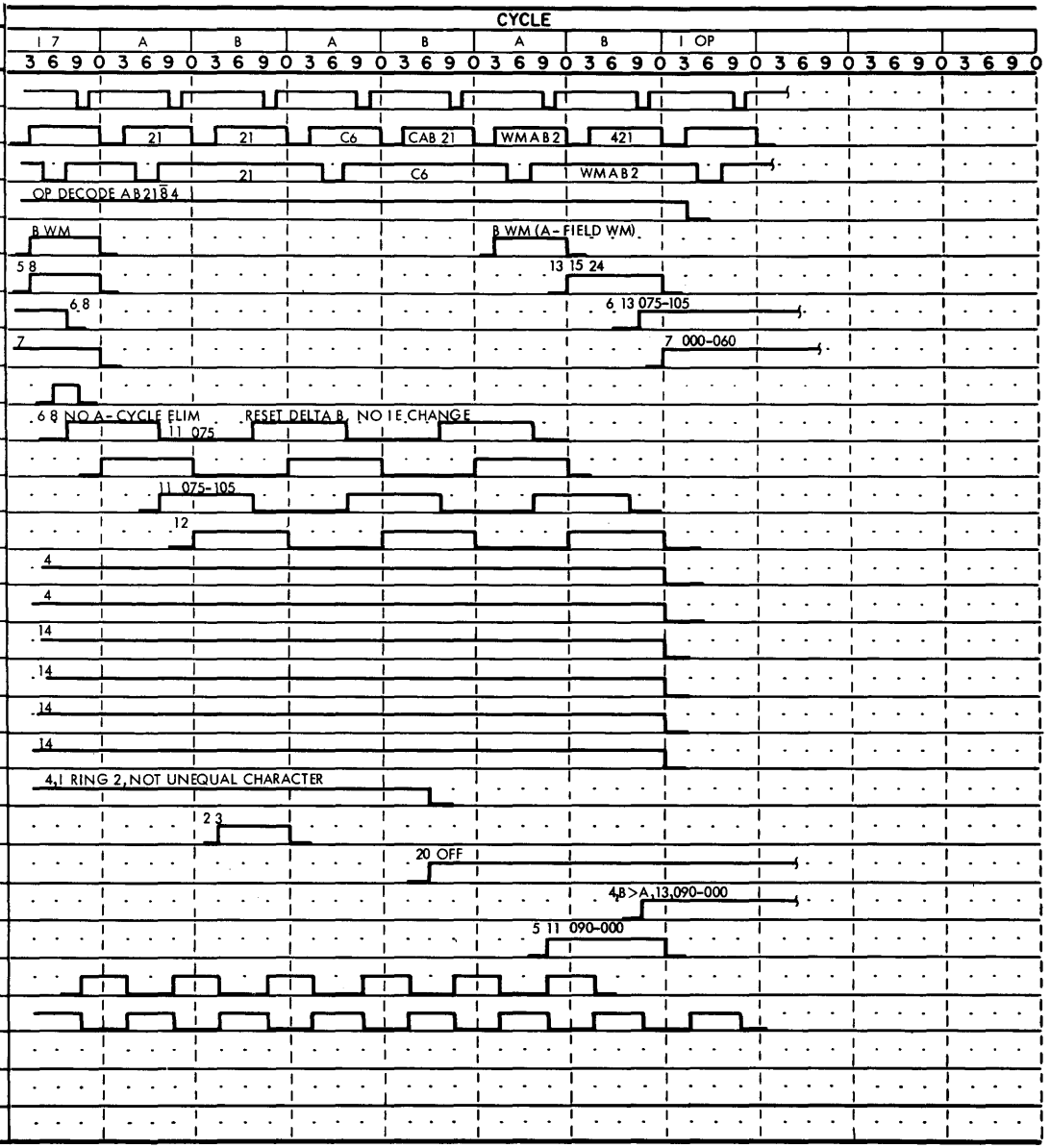
Figure 13. Compare Sequence

Figure 14. High-Low-Equal Compare Sequence

ORIGINAL A FIELD "B63" OR WM AB2 C42 21

ORIGINAL B FIELD "7C3" WM 421 CAB21 21

		SIGNAL NAME	LOGIC
1		STORAGE ADDRESS REGISTER	32.35.XX 32.36.XX
2		B-REGISTER	35.11.XX
3		A-REGISTER	35.16.XX
4	+U	COMPARE OPERATION	35.24.11
5	-T	GATED WORD MARK	31.07.11
6	-T	I/E CHANGE	31.05.31
7	+U	DELTA I CYCLE LATCH	31.21.11
8	-T	I CYCLE LATCH	31.24.11
9	+U	I STAR RESTORE	32.39.11
10	-T	DELTA A CYCLE LATCH	31.22.11
11	-T	A CYCLE LATCH	31.25.11
12	-T	DELTA B CYCLE LATCH	31.23.11
13	+U	B CYCLE LATCH	31.26.11
14	+U	TRANSFER B REGISTER	35.18.11
15	+U	I/E CONTROL 1	31.05.21
16	-T	B REG ZONE INHIBIT	36.13.21
17	-T	B REG DIGIT INHIBIT	36.13.21
18	-T	B REG C BIT INHIBIT	35.18.21
19	-T	B REG WM INHIBIT	35.40.21
20	+U	COMPARE EQ LATCH OUTPUT	34.21.21
21	-T	EQUAL CHARACTER	34.21.11
22	-T	COMPARE UNEQUAL LAT OUTPUT	34.21.21
23	+U	HIGH LATCH	44.34.21
24	+U	A REG WM LATCH	31.06.11
25	-T	MODIFIER CONTROL -1	32.42.21
26	-T	MODIFIER CONTROL TRANSFER	32.42.41



is tested at 090-000 of each B-cycle for B-less-than-A or B-greater-than-A. The B-greater-than-A signal turns the HIGH latch ON; the B-less-than-A signal turns it OFF. The condition of the HIGH latch, when the operation is stopped by a word-mark, indicates whether the B-field is greater than the A-field, or not.

The test to turn ON or turn OFF the HIGH latch is made in three stages as follows:

1. Special Character or Alphamerical Character.
   Figure 13 indicates the compare sequence from high-to-low. Note that two major divisions exist, alphameric characters and special characters. If the characters in the A- and B-register are not in the same major division, an immediate decision can be made whether B is greater-than-A or less-than-A. The various bits required to identify special characters are switched together for both the A- and B-register as shown in ILD 36. The result of the test is:
   A. B greater than A—switches with 090-000, B-CYCLE and COMPARE OP to turn on the HIGH latch.
   B. B less than A—switches same as (A) above to turn OFF the HIGH latch if it was ON.
   C. A test-zone signal if either, both, or neither A- and B-registers contents were special characters.

2. Zone Test.
   When both the A- and B-register characters are special characters, or when both are alphamerical characters, a further test is made of the zone groups to determine the result of the compare operation. The bits from the A- and B-registers are switched together as shown on ILD 36 to result in one of the following listed signals.
   A. B greater than A—switches with 090-000, B cycle and Compare Op to turn ON the HIGH latch.
   B. B less than A—switches same as (A) above to turn off the HIGH latch if it was ON.
   C. Test Digit—If the A- and B-register zones are equal, a test of the digits in the character is necessary to determine the compare results.

3. Digit Test.
   When both the special character and zone tests fail to determine the results of compare, a test is made of the digit portion of the zone group as shown in ILD 36. The result of the test may be either B-greater-than-A or B-less-than-A. In either case, Test Digit and Unequal Character (from the basic compare circuits) switch with the results of the digit compare, and in turn with B-cycle, 090-000 and Compare Op to turn ON or turn OFF the HIGH latch. The HIGH latch remains either ON or OFF until it is changed during the next compare operation.

When an A-register word-mark is sensed before a B-register word-mark, the B-field is greater than the A-field, so the HIGH latch is forced on by the switching shown in ILD 36.

Test and Transfer Program — B (AAA) d — d = U, S, T

Circuits for this operation code are the same as for other test and branch codes described in the basic *1401 CE Manual of Instruction*, Form 225-6540. The circuits required to test for high, low, and equal are illustrated in ILD 36. If the (d) portion of the instruction is U (High-Compare) and the high and unequal signals are set, a transfer program skip signal will result. The (d) = S and T variations operate in a similar manner.

Calculate Feature (Multiply-Divide)

Multiply-divide is a single optional feature which can be installed on 1401 Model B or C systems.

Two operation codes are included with the calculate option. They are:

Multiply —	@(AAA)(BBB)
Divide —	%(AAA)(BBB)

Basically, multiply is a repetitive add operation controlled by the optional multiply circuits, and divide is a repetitive subtract operation controlled by the optional divide circuits. An A-auxiliary address register (A-AUX-STAR) with A-auxiliary-cycle control, and a B-auxiliary address register (B-AUX-STAR) with B-auxiliary-cycle control is added for control of the multiply-divide (M-D) operations.

The basic arithmetic unit has been modified and used to perform the repetitive add and subtract operations. These modifications include doubling and divide-compare circuits which are discussed in the following sections.

Multiply

Data Flow

Multiplication is a controlled repetitive add operation. The multiplicand field is added to the proper position of the product field as many times as indicated by the multiplier.

Because the basic arithmetic unit has been modified, it is possible to add the multiplicand singly or doubly to the product field on each add operation. This saves one add operation each time doubling is used.

The arithmetic unit has been modified by inserting qui-binary doubler entry circuits between the A-register Translator and the A-register True/Complement Switching. By activating the 2A gate, the A-register input to the qui-binary adder enters in doubled value due to the qui-binary doubler entry circuits. If the A-gate is activated, the A-register input to the qui-binary adder enters in normal value, by passing the qui-binary doubler entry circuits.

The doubler circuits require that each multiplier digit be tested before each add operation to determine whether to add the multiplicand singly or doubly. If the multiplier digit is equal to, or greater than, two, the multiplier digit is reduced by two and the 2A gate is activated. This gates the output of the qui-binary doubler entry circuits to the qui-binary adder, and allows the multiplicand to be added doubly to the product field.

If the multiplier digit is one, the multiplier digi is reduced to zero and the A-gate is activated. This gates the output of the A-register translator directly to the qui-binary adder, bypassing the qui-binary doubler entry circuits, and allows the multiplicand to be added singly to the product field.

If the multiplier digit is zero, no add operation is initiated, but a shift in the product field position ocurs, and the next multiplier digit is tested.

The following example is discussed step-by-step to illustrate the basic principles of 1401 multiply. It is not intended to show actual machine operation, so that signs, word-marks, and field assignments are not shown.

1. The problem to be used is shown in the example:

M'cand field	36
M'plier field	× 203
Product field	000

2. The units multiplier digit (3) is tested, reduced by 2, and the 2A gate is activated. After the multiplicand is added doubly to the product field, the results are:

M'cand field	36	
M'plier field	201	2A gate
	000	
	72	
Product field	072	

3. The units multiplier digit (1) is tested, reduced to zero, and the A-gate is activated. After the multiplicand is added singly to the product field, the results are:

M'cand field	36	
M'plier field	200	A-gate
	072	
	36	
Product field	108	

4. The units multiplier digit (0) is tested and found to be zero. This causes the product field to be shifted one position to the left. Therefore, the multiplicand should be added to the tens position of the product field on the next add operation.

The tens multiplier digit (0) is tested and found to be zero. This causes another shift in the product field so that the multiplicand is added to the hundreds position of the product field on the next add operation. The hundreds multiplier digit (2) is tested, reduced to zero, and the 2A gate is activated. After the multiplicand is added doubly to the hundreds position of the product field, the results are:

M'cand field	36	
M'plier field	000	2A Gate
	108	
	72	
Product field	7308	

The last multiplier digit has been reduced to zero and the multiply operation is ended.

Whenever an add (add-A or add-2A) operation is initiated, it is necessary to address the units position of the multiplicand (A-field), and also address some low-order position of the product (B-field). The position of the B-field addressed depends on how many shifts have occurred. The A-AUX-STAR and the B-AUX-STAR are used to store the proper positions of the A-field and B-field to be addressed at the beginning of each add operation.

The A-field address is transferred from the A-STAR to the A-AUX-STAR on the first A-cycle, and the B-field address is transferred from the B-STAR to the B-AUX-STAR on the first B-cycle. The A-AUX-STAR address does not change during the multiply operation. The B-AUX-STAR address is decreased by one whenever a zero multiplier indicates need for a shift in the product field.

Prior to initiating a multiply instruction—@(AAA)-(BBB), a B-field must be provided in storage. The B-field is used to store the multiplier in the high-order position at the beginning of the multiply operation. The B-field length must be equal to the sum of the multiplier digit positions plus the multiplicand digit positions plus one. The B-field sign need not be in standard form, but a WM is required in the high-order position. (AB for plus and B for minus are considered standard form in the 1401 manual.)

The multiplier is placed in the high-order position of the B-field by a move, load, or reset-add operation code. The low order positions of the B-field are cleared and set to zero in the beginning of the multiply operation.

The A-field contains the multiplicand and must have a high-order WM. The A-field sign need not be stored in standard form.

A - FIELD						457	458
						3	B 6

B - FIELD		673	674	675	676	677	678
		2	0	3	3	2	1

Figure 15. A-Field and B-Field Before Multiply

For the multiply operation presented in this section, the following program is used.

(900)	L	(T83)	(675)
(907)	@	(458)	(678)
(914)	2		

After the load operation (L T83 675) is completed, the A- and B-fields appear as shown in Figure 15.

The multiplicand is −36 (A-field). The multiplier, loaded into (675) from (T83), is +203 (high-order positions of B-field). The low-order positions of the B-field (___321) remain from previous operations and are cleared and set to zero in the beginning of the multiply operation. After the multiply operation @(458)(678) is completed, the A- and B-fields appear as shown in Figure 16.

The A-field is not changed by the operation. The B-field multiplier has been reduced to zero, and the product, with a standard sign, is located in the low-order positions of the B-field.

Multiply Operations — @ (AAA) (BBB)

The major objective of a mutliply operation is to multiply the A-field (multiplicand) by the multiplier located in the high-order positions of the B-field, and to develop a product in the low-order positions of the B-field. Algebraic sign control is used, and the product sign is produced in standard form.

Figure 17 is a completed work sheet of the problem to be presented here. The multiply operation is performed by a series of A- and B-cyles, which accomplish the following minor objectives:

A - FIELD						457	458
						3	B 6

B - FIELD		673	674	675	676	677	678
		0	0	7	3	0	B 8

Figure 16. A-Field and B-Field After Multiply

A-FIELD ADDRESS				457	458
A-FIELD BEFORE OPERATION				3	B_6
A-FIELD AFTER OPERATION				3	B_6

INSTRUCTIONS: (900) L T83 675
(907) @ 458 678
(914) 2

B-FIELD ADDRESS			673	674	675	676	677	678
B-FIELD BEFORE OPERATION			2	0	3	3	2	1
B-FIELD AFTER CLEAR OPERATION			2	0	1	0	0	0
B-FIELD AFTER ADD 2A OPERATION			2	0	0	0	7	B_2
B-FIELD AFTER ADD A OPERATION			0	0	0	1	0	B_8
B-FIELD AFTER ADD 2A OPERATION			0	0	7	3	0	B_8

CYCLE LATCH	I	A	B	A	B	A	B	B	B	A Aux	B Aux	A	B	B	B	A Aux	B Aux	A	B	B	B	B Aux	B	B Aux	B	A Aux	B Aux	A	B	B	B	I
STAR	914	458	678	458	678	457	677	676	675	458	678	457	677	676	675	458	678	457	677	676	675	678	674	677	673	458	676	457	675	674	673	914
B-REGISTER	2	B_6	1	B_6	0	3	2	3	3	B_6	0	3	0	0	1	B_6	B_2	3	7	0	0	B_8	0	0	2	B_6	1	3	0	0	0	2
A-REGISTER	8	B_6		B_6		3		C	C	B_6		3		C	C	B_6		3		C	C	C	C	C	C	B_6		3		C	C	0
TO INHIBIT	2	B_6	0	B_6	0	3	0	0	1	B_6	B_2	3	7	0	0	B_6	B_8	3	0	1	0	B_8	0	0	0	B_6	3	3	7	0	0	2
A-STAR	458	458		457		456				457		456				457		456								457		456				
B-STAR	678		678		677		676	675	674		677		676	675	674		677		676	675	674		673		672		675		674	673	672	
A-AUX-STAR	xxx	458																														
B-AUX-STAR	yyy		678																			677		676								
SIGN TRIGGER		On																														Off

SET AUX-STARS
CLEAR B-FIELD OPR
TESTS UNITS MULT DIGIT
ADD 2A OPR
TEST UNITS MULT DIGIT
ADD A OPR
TEST UNITS MULT DIGIT
SHIFT PROD FIELD
TEST TENS MULT DIGIT
SHIFT PROD FIELD
TEST HUNDS MULT DIGIT
ADD 2A OPR
TEST HUNDS MULT DIGIT

Figure 17. @(AAA)(BBB) Multiply Worksheet

1. Prevent address modification and set the AUX-STARS on the first A- and B-cycle.
2. Set the low-order B-field positions to zero and test the units (greater than 2) multiplier digit.
3. Perform an add-2A operation and test the units (equal to 1) multiplier digit.
4. Perform an add-A operation and test the units (zero) multiplier digit.
5. Shift the product field and test the tens (zero) multiplier digit.
6. Shift the product field and test the hundreds (equal to 2) multiplier digit.
7. Perform an add-2A operation, test the hundreds (equal to 0) multiplier digit, and end the multiply operation.

In the following discussion of cycle-by-cycle multiply operations, mention of the various control latches and the cycles in which they are activated is included. The section under *Summary of Multiply-Divide Switching and Controls* contains a detailed summary and ILD reference of multiply-divide switching and controls, and should be used as reference for exact timing and switching conditions.

On the last I-cycle of I-phase, the AUX-STAR SET latch and the CLEAR B-FIELD latch are set to control the first minor objective. The fields and registers appear as shown in Figure 18 during the last I-cycle.

The AUX-STARS are set in the beginning of E-phase (on the first A- and B-cycle). E-phase begins with a normal A-cycle during which the units position of the A-field is read into the B-register and A-register, and transferred back to storage. The A-field sign is analyzed on this cycle and the sign trigger is turned ON if the sign is minus. TRANSFER STAR is activated preventing minus-one address modification of the A-STAR. The unmodified output of the STAR is gated into the A-STAR and into the A-AUX-STAR.

The following cycle is a normal B-cycle and the units position of the B-field is read into the B-register. A zero is forced back to storage as the low-order B-field position is cleared and set to zero under control of the CLEAR B-FIELD latch. TRANSFER STAR is still activated preventing minus-one address modification of the B-STAR. The unmodified output of the STAR is gated into the B-STAR and into the B-AUX-STAR.

TRANSFER STAR and the AUX-STAR read-in are controlled by the AUX-STAR SET latch. The zero was forced to the B-field under control of the CLEAR B-FIELD latch. This accomplishes the first minor objective of setting the AUX-STARS. The fields and registers appear as hown in Figure 19 during the first B-cycle.

The low-order B-field positions are set to zero as alternate A- and B-cycles continue (using minus-one

A-FIELD						457	458
						3	B_6

B-FIELD		673	674	675	676	677	678
		2	0	3	3	2	1

STAR	914
A-STAR	458
B-STAR	678
A-AUX-STAR	XXX
B-AUX-STAR	YYY

Figure 18. Multiply Fields and Registers Chart #1

address modification). On each A-cycle, the A-field data is read into the B-register and A-register, and transferred back to storage. On each B-cycle, the B-field data is read into the B-register and replaced with zeros under control of the CLEAR B-FIELD latch. The A-field WM (required for multiply) indicates the end of the A-field. A-cycles are eliminated and B-cycles continue.

A multiplier digit is always read into the B-register on the third B-cycle after the A-field WM is detected, if the fields have been properly set. On the first B-cycle after the A-field WM, the M-D A-CYCLE ELIMINATE latch is set and the B-field position addressed is cleared and set to zero.

On the second B-cycle after the A-field WM, the X-POSITION latch (extra position latch) is set, the B-field position addressed is cleared and set to zero, and the A-register is reset and set to blank (C-bit).

On the third B-cycle after the A-field WM, the MULTIPLIER latch is set, identifying the B-field position addressed as the multiplier digit. Because the multiplier digit is analyzed and found to be greater than 2 (units multiplier digit is 3), the multiplier digit is reduced by 2 and read back into storage (1 is read back as the new units multiplier digit). The A-register is reset and set to blank (C-bit) and the 2A latch is et to activate an add-2A operation. The M-D A-CYCLE ELIMINATE latch, the X-POSITION latch and the MULTIPLIER latch are reset when a significant multiplier digit is found.

The sign of the multiplier is analyzed and the sign trigger is flipped (binary coupled set pulse) if the

A-FIELD						457	458
						3	B_6

B-FIELD		673	674	675	676	677	678
		2	0	3	3	2	0

STAR	678
A-STAR	458
B-STAR	678
A-AUX-STAR	458
B-AUX-STAR	678

Figure 19. Multiply Fields and Registers Chart #2

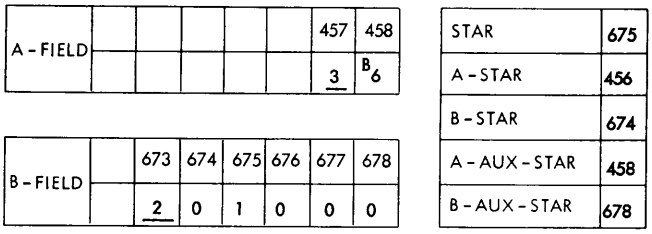


Figure 20. Multiply Fields and Registers Chart #3

sign is minus. Because the multiplicand is minus and the multiplier is plus for the example used, the sign trigger remains ON.

The set PRODUCT SIGN latch is also set on only this third B-cycle after the A-field WM. This allows a sign to be set in the units position of the product field as determined by the sign trigger on the next B-cycle or B-auxiliary cycle (B-CYCLE latch is also on during B-auxiliary cycles).

Because the units position of the A- and B-fields are to be addressed on the following cycles to initiate the add 2A operation, the DELTA A-AUX-CYCLE latch is set to allow an A-AUX-CYCLE.

The low-order B-field positions have been cleared and set to zero and the units (greater than 2) multiplier has been tested to complete the second minor objective. The fields and registers appear as shown in Figure 20 during the third B-cycle after the A-field WM.

The add-2A operation begins with an A-auxiliary-cycle in which the units position of the A-field is addressed. The contents of the A-AUX-STAR was read into the STAR on the Delta A-auxiliary cycle. The modified output (minus one address modification) of the STAR is read into the A-STAR but not into the A-AUX-STAR.

The units position of the A-field is read into the B-register and A-register and transferred back to storage.

The DELTA B-AUX-CYCLE latch is set to allow the B-field units position address in the B-AUX-STAR, to be read into the STAR on the following cycle. Therefore, the following cycle is a B-auxiliary cycle.

On the auxiliary cycles, the A- or B-auxiliary latches are set in addtiion to the normal A- or B-cycle latches to control the AUX-STARS and allow additional control of the A- or B-STARS.

On the B-auxiliary cycle, the units position of the product field is read into the B-register. The qui-binary adder develops an output on the B-auxiliary cycle, which is the sum of the true input from the B-register plus the true and doubled input from the A-register and a carry or no-carry condition. The output of the qui-binary adder is gated back to storage along with the sign. The sign is gated by the SET PRODUCT SIGN latch and determined by the sign trigger. The modified output of the STAR is read into the B-STAR but not into the B-AUX-STAR.

Alternate A- and B-cycles continue with the A-field data being read into the B-register and A-register and transferred back to storage. On the B-cycles, the output of the qui-binary adder is gated back to storage as the add-2A operation develops the partial product.

On the first B-cycle after the A-field WM, the M-D A-CYCLE ELIMINATE latch is set, and continuous B-cycles occur. On the second B-cycle after the A-field WM, the X-POSITION latch is set. The A-register is reset and set to blank to allow the qui-binary adder to develop the correct output on this last cycle of the add-2A operation.

On the third B-cycle after the A-field WM, the MULTIPLIER latch is set and the multiplier digit is analyzed. Because the multiplier digit is equal to one (units multiplier digit is 1), the multiplier digit is reduced to zero and read back into storage (0 is read back as the new units multiplier digit). The A-register is also reset and set to blank.

The M-D A-CYCLE ELIMINATE latch, the X-POSITION latch and the MULTIPLIER latch are reset when the significant multiplier digit is found.

The A-latch is set to activate an add-A operation and the DELTA A-AUX-CYCLE latch is set to allow an A-auxiliary cycle, and the beginning of the add-A operation.

Thus, an add-2A operation has been completed and the equal-to-1 multiplier digit has been tested, completing minor objective three. The fields and registers appear as shown in Figure 21 during the third B-cycle after the A-field WM.

The add-A operation is basically the same as the add-2A operation just covered. The first cycle is an A-auxiliary cycle in which the units position of the A-field is read into the B-register and A-register, and transferred back to storage. The modified output of the STAR is read into the A-STAR but not the A-AUX-STAR. The DELTA B-AUX-STAR latch is set and the units position address of the product field is read into the STAR.

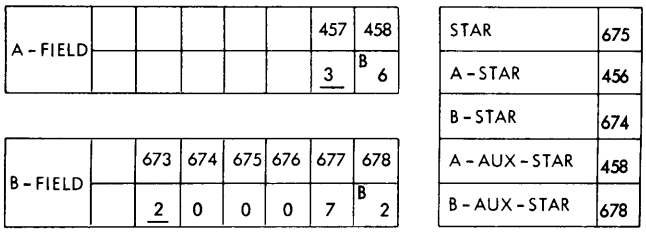


Figure 21. Multiply Fields and Registers Chart #4

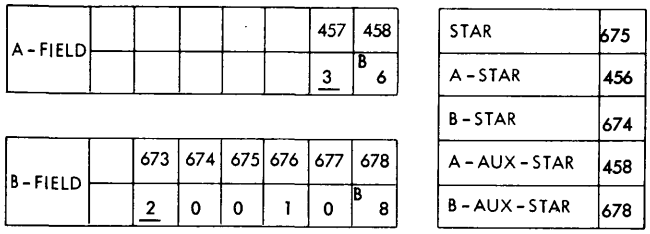


Figure 22. Multiply Fields and Registers Chart #5

On the following B-auxiliary cycle, the true and normal value input from the A-register is combined with the true input from the B-register, and a carry or no-carry condition. The qui-binary adder output is gated to storage together with the sign from the B-register.

The modified output of the STAR is read into the B-STAR but not into the B-AUX-STAR.

Alternate A- and B-cycles continue, and the multiplicand is added to the product field singly.

The A-field WM signifies the end of the A-field, and the M-D A-CYCLE ELIMINATE latch is set on the following B-cycle.

On the second B-cycle after the A-field WM, the X-POSITION latch is set and the A-register is reset and set to blank. This is also the last cycle of the add-A operation.

On the third B-cycle after the A-field WM, the MULTIPLIER latch is set, and the multiplier digit in the B-register is analyzed. Because the multiplier digit has been reduced to zero (units multiplier digit is 0), a shift in the product field is necessary. This is accomplished by modifying the B-AUX-STAR address by minus one. Therefore, the DELTA B-AUXILIARY CYCLE latch is set to allow the following cycle to be a B-auxiliary cycle. A zero is forced back to the B-field on this cycle, and the A-register is also reset and set to blank.

This completes the add-A operation and the test of the zero multiplier, thus completing the fourth minor objective. The fields and registers appear as shown in Figure 22 during the third B-cycle after the A-field WM.

A B-auxiliary cycle is required to shift the product field. The next cycle is a B-auxiliary cycle in which the units position of the partial product is read into the B-register and transferred back to the B-field. The purpose of the cycle is to read the modified output from the STAR into the B-AUX-STAR but not into the B-STAR. This decreases the B-AUX-STAR by one, which effectively shifts the product field. Because the B-STAR was not read in, it still contains the address of the next multiplier digit. The A-register is also reset and set to blank.

The modification of the B-AUX-STAR was possible on this B-auxiliary cycle because a significant multipler had not been found, and the M-D A-CYCLE ELIMINATE latch, the X-POSITION latch, and the MULTIPLIER latch are still ON.

A normal B-cycle is required to test the next multiplier digit. Because A-cycles are still eliminated, the next cycle is a normal B-cycle in which the next multiplier digit (tens multiplier digit is 0) is read into the B-register. Because it is a zero multiplier, the M-D A-CYCLE ELIMINATE latch, X-POSITION latch, and MULTIPLIER latch are not deactivated, and no add-A or add-2A operation can be initiated. A zero is forced back to the B-field on this cycle, and the A-register is also reset and set to blank. The zero multiplier digit also indicates that another product shift is required.

The DELTA B-AUX-CYCLE latch is set to allow a B-auxiliary cycle in which to decrease the B-AUX-STAR and effect a product shift. Because the multiplier is also located in the B-field, the next multiplier position to be tested is also shifted.

Minor objective five has been accomplished, and the fields and registers appear as shown in Figure 23 during the normal B-cycle.

Another B-auxiliary cycle is required to shift the product field. The next cycle is a B-auxiliary cycle in which the tens position of the partial product is read into the B-register and transferred back to the B-field. The modified output of the STAR is read into the B-AUX-STAR but not into the B-STAR. This accomplishes shifting of the product field. Because the B-STAR was not disturbed, it contains the address of the next multiplier digit. The A-register is also reset and set to blank.

A normal B-cycle is required to test the next multiplier digit. Because A-cycles are still eliminated, a normal B-cycle is initiated and a significant digit (hundreds multiplier digit is 2) is read into the B-register. The MULTIPLIER latch is still ON, so the B-register digit is recognized as a multiplier digit. Because the hundreds multiplier digit is equal to 2,

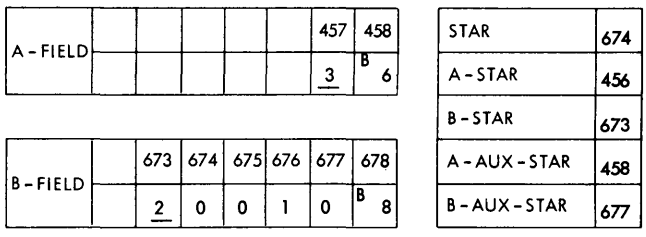


Figure 23. Multiply Fields and Registers Chart #6

the 2A latch is set, and an add-2A operation is initiated. The multiplier digit is reduced to zero and stored in the B-field together with the WM transferred from the B-register.

The M-D A-CYCLE ELIMINATE latch, the X-POSITION latch, and the MULTIPLIER latch are reset because a significant multiplier was located. The DELTA A-AUX-CYCLE latch is set to allow an A-auxiliary cycle which initiates the final add-2A operation.

This completes minor objective six, and finds the fields and registers as shown in Figure 24 during the normal B-cycle.

The final add-2A operation is the same as the first add-2A operation discussed.

The first two cycles are an A-auxiliary cycle followed by a B-auxiliary cycle. On each auxiliary cycle, the modified output of the STAR is not read into the corresponding AUX-STAR, but into the corresponding A- OR B-STAR.

On the A-auxiliary cycle and a A-cycles, the multiplicand is read into the B-register and A-register, and transferred back to storage.

On the B-auxiliary cycle and B-cycles, the output of the qui-binary adder is gated to the B-field. Because this is an add-2A operation, the A-register input to the qui-binary adder is doubled so that the multiplicand is added doubly to the partial product field.

When the A-field WM is detected, the following B-cycle is used to set the M-D A-CYCLE ELIMINATE latch. On the second B-cycle following the A-field WM, the X-POSITION latch is set, and the A-register is reset and set to blank.

On the third and last B-cycle of this multiply operation, the MULTIPLIER latch is set and the high-order (WM) position of the B-field is read into the B-register. The digit (hundreds multiplier digit is *0*) is analyzed and found to be a zero with a WM which indicates the end of the multiply operation. An I-E change is initiated ending the multiply operation.

With the last minor objective complete, the fields and registers during the last cycle of the E-phase appear as shown in Figure 25.

Figure 17 is a complete work sheet showing the cycle-by-cycle results of the multiply problem just discussed.

A-FIELD						457	458
						3	B_6

B-FIELD		673	674	675	676	677	678
		0	0	0	1	0	B_8

STAR	673
A-STAR	456
B-STAR	672
A-AUX-STAR	458
B-AUX-STAR	676

Figure 24. Multiply Fields and Registers Chart #7

A-FIELD						457	458
						3	B_6

B-FIELD		673	674	675	676	677	678
		0	0	7	3	0	B_8

STAR	673
A-STAR	456
B-STAR	672
A-AUX-STAR	458
B-AUX-STAR	676

Figure 25. Multiply Fields and Registers Chart #8

A zero multiplier or multiplicand is handled in a normal manner. In the case of a zero muliplicand, the multiply operation begins by setting the AUX-STARS and clearing the low-order B-field positions. Each multiplier digit is tested as the operation progresses, initiating an add-A or add-2A operation or shifting as required. The zero multiplicand is added singly or doubly on each add operation. When the last multiplier digit (identified by a WM) is reduced to zero, an I-E Change is initiated which ends the multiply operation. The net result is to reduce the multiplier to zero, and develop a zero product with a standard sign.

With a zero multiplier, no add-A or add-2A operation is ever initiated. The multiply operation begins by setting the AUX-STARS and clearing the low-order B-field positions. The first multiplier digit tested initiates a shift in the product field by taking a B-auxiliary cycle. This is followed by a B-cycle in which the next zero multiplier is tested. This pattern repeats until the last zero multiplier digit (identified by a WM) is tested, and an I-E Change is initiated which ends the multiply operation. The net result is a zero product with a standard sign.

A special problem exists if the zero multiplier is a single multiplier digit. On the B-cycle that the zero multiplier digit is addressed, the B-register contains a zero and a WM. Because the product sign is still in the sign trigger, an I-E change must be presented, and a B-auxiliary cycle initiated to allow the product sign to be set in storage. This is done by preventing an I-E change on the third B-cycle after the A-field WM, and initiating a B-auxiliary cycle. The sign is gated to storage on the B-auxiliary cycle and an I-E change is indicated. The net result is still a zero product field with a sign in standard form as determined by algebraic sign control.

The ARITHMETIC CHECK latch is set if an error occurs in the qui-binary adder when the ARITH DIGIT GATE CALC is activated. This results in a logic-check light on the 1401 console. The machine stops on the following cycle.

Divide

Data Flow

Division is a controlled repetitive subtract operation. The divisor is subtracted from the proper partial dividend positions until the final partial dividend (remainder) is less than the divisor. A quotient is developed by maintaining a count, in each quotient position, of each possible subtract operation in each dividend position.

As mentioned previously, the arithmetic unit has been modified by inserting qui-binary doubler circuits between the A-register translator and the A-register true/complement switching. Therefore, it is possible to subtract the divisor singly or doubly from each partial dividend.

Subtraction is accomplished by complementing (9's complement) the qui-binary adder input from the A-register, and forcing a carry condition for the units position of each subtract operation. Complementing occurs in the A-register true/complement switching as in the basic 1401, and is initiated by activating the complement line during each subtract operation.

The qui-binary adder input from the A-register is a normal value input if the A-gate is activated; or is a doubled input if the 2A gate is activated. As in multiply, if the A-gate is activated, the input to the qui-binary adder comes directly from the A-register translator, bypasing the qui-binary doubler circuits. If the 2A gate is activated, the input to the qui-binary adder comes directly from the qui-binary doubler circuits.

To determine whether the divisor should be subtracted from the partial dividend singly or doubly requires divide-compare circuitry which allows the divisor to be compared with the partial dividend. If the partial dividend is less than the divisor ($B<A$), no subtract operation is initiated, a 0 is added to the quotient digit, and the partial dividend field is shifted to allow the divisor to be compared with a larger partial dividend field.

If the partial dividend is equal to or greater than the divisor, but less than twice the divisor ($A\geq B<2A$), the quotient digit is increased by 1, and a subtract-A operation is initiated. As the divisor is subtracted singly from the partial dividend, the output developed by the qui-binary adder (reduced partial dividend) is simultaneously gated back to storage and compared with the divisor to determine whether another subtract operation, or a shift in dividend field, is required.

The divide-compare operation which occurs simultaneously with a subtract-A operation should always indicate that the reduced partial dividend is less than the divisor, and so the following operation is always a shift in the dividend field.

The simultaneous subtract and divide-compare operations save processing time by eliminating separate divide-compare operations, and are used whenever possible.

If the partial dividend is equal to or greater than twice the divisor ($B\geq 2A$), the quotient digit is increased by 2, and a simultaneous subtract-2A and divide-compare operation is initiated. As the divisor is subtracted doubly from the partial dividend, the output of the qui-binary adder (reduced partial dividend) is simultaneously gated back to storage and compared with the divisor. A subtract-2A operation may be followed by: a simultaneous subtract-2A and divide-compare operation; a simultaneous subtract-A and divide-compare operation; or a dividend field shift. Therefore, the divide-compare made during the subtract-2A operation may indicate that the reduced partial dividend is still equal to or greater than twice the divisor ($B\geq 2A$); or that the reduced partial dividend is equal to or greater than the divisor but less than twice the divisor ($A\leq B<2A$); or that the reduced partial dividend is less than the divisor ($B<A$).

The following example is discussed step-by-step to illustrate the basic principles of 1401 divide. It is not intended to show actual machine operation, so that signs, word-marks, and field assignments are not shown.

1. The problem to be used is shown in this example:

```
                 000   Quotient
Divisor    27 | 340   Dividend
```

2. The divisor (27) is compared (divide-compare operation) with the first partial dividend (3). The result of the comparison indicates that the dividend is less than the divisor ($B<A$).

```
                 000   Quotient
Divisor    27 | 340
                 ^     Partial Dividend
```

3. As a result of the divide-compare operation, a zero is added to the high-order quotient digit position and a shift in the dividend field is initiated.

```
                +0
                 000   Quotient
Divisor    27 | 340
                  ^    Partial Dividend
```

4. The divisor (27) is compared with the new partial dividend (34). The result of the divide-compare operation indicates that the dividend is equal to or greater than the divisor but less than twice the divisor ($A\leq B<2A$).

```
                 000   Quotient
Divisor    27 | 340
                  ^    Partial Dividend
```

A - FIELD						457	458
						2	B 7

B - FIELD		673	674	675	676	677	678
		0	0	0	3	4	B 0

Figure 26. A-Field and B-Field Before Divide

5. As a result of the divide-compare operation, a one is added to the next lower quotient digit position (due to shift in dividend field).

```
                +1
                000     Quotient
Divisor    27 | 340
                ^       Partial Dividend
```

6. A subtract-A operation and a divide-compare operation are initiated simultaneously. The divisor (27) is subtracted singly (27) from the partial dividend (34). Simultaneously, the divisor (27) is compared with the reduced partial dividend (7). The result of the comparison indicates that the reduced partial dividend is less than the divisor (B<A).

```
                010     Quotient
Divisor    27 | 340
A-Gate        27
              7
              ^         Reduced Partial Dividend
```

7. As a result of the divide-compare operation, a zero is added to the second quotient digit position and a shift in the dividend field is initiated.

```
                +0
                010     Quotient
Divisor    27 | 34
              27
              70        Partial Dividend
              ^
```

8. The divisor (27) is compared with the next partial dividend (70). The result of the divide-compare operation indicates that the dividend is equal to or greater than twice the divisor (B≥2A).

```
                010     Quotient
Divisor    27 | 34
              27
              70        Partial Dividend
              ^
```

9. As a result of the divide-compare operation, a two is added to the units quotient digit position (due to the second shift in the dividend field).

```
                +2
                010     Quotient
Divisor    27 | 34
              27
              70        Partial Dividend
              ^
```

10. A subtract-2A operation and a divide-compare operation are initiated simultaneously. The divisor (27) is subtracted doubly (54) from the partial dividend (70). Simultaneously, the divisor (27) is compared with the reduced partial dividend (16). The result of the divide-compare operation indicates that the reduced partial dividend is less than the divisor (B<A).

```
                012     Quotient
Divisor    27 | 34
2A Gate       27
              70
              54
              16        Reduced Partial Dividend
              ^
```

11. As a result of the divide-compare operation, a zero is added to the units quotient digit. The divide operation ends because the units position of the dividend field has been detected (indicated by the dividend sign in standard form) and the last partial dividend (remainder) reduced so that it is less than the divisor (B<A).

```
                +0
                012     Quotient
Divisor    27 | 34
              27
              70
              54
              16        Remainder
              ^
```

In actual operation, algebraic sign control is used. The original dividend sign in standard form becomes the sign of the remainder. A sign in standard form is developed for the quotient as a result of analyzing the divisor and dividend signs. The divisor sign need not be in standard form.

Whenever a subtract (subtract-A or subtract-2A) operation is initiated, it is necessary to address the units position of the divisor (A-field), and also address some position of the dividend (B-field). The position of the B-field to be addressed depends on how many shifts have occurred. The A-AUX-STAR and the B-AUX-STAR are used to store the proper position of the A-field and B-field to be addressed at the beginning of each subtract operation.

A - FIELD						457	458
						2	B 7

B - FIELD		673	674	675	676	677	678
		0	1	AB 2	0	1	B 6

Figure 27. A-Field and B-Field After Divide

At the end of I-phase for the divide instruction, %(AAA)(BBB), the A-field address is in the A-STAR, and the B-field address is in the B-STAR. The first A-cycle and first B-cycle are used to transfer the addresses from the A-STAR and B-STAR to the A-AUX-STAR and B-AUX-STAR.

The address in the A-AUX-STAR is used during the divide operation to locate the units position of the divisor, and is not changed during the divide operation. The address in the B-AUX-STAR is used during the divide operation to locate the units position of the partial dividend, and is changed during the divide operation. Whenever a divide-compare operation indicates B<A, the dividend field is shifted by increasing the B-AUX-STAR by one. This allows comparison of the divisor to a larger dividend field, and also affects the position in which the next quotient digit is entered.

Prior to initiating the divide instruction, %(AAA)(BBB), a B-field must be provided in storage. The length of the B-field must be equal to the sum of the divisor digit positions plus one. The B-field contains the dividend in the low-order positions at the beginning of the divide operation. At the end of the divide operation, the B-field contains the quotient in the high-order positions and the remainder in the low-order positions.

The B-field does not require a WM in the high-order position, but the units position of the B-field (units position of the dividend) must have a sign in standard form. The high-order positions of the B-field, beyond the dividend, must be zeros or blanks. Therefore, the dividend is best placed in the B-field by a reset-add operation. This insures high-order zeros and a standard units sign. The reset-add operation requires that the high-order position of the B-field be defined by a WM. Although the B-field WM is not actually required for the divide operation, its presence has no effect on the divide operation.

The A-field contains the divisor and must be defined with a high-order WM. The sign of the A-field does not have to be in standard form.

For this divide operation presented in this section, the following program is assumed.

```
(900)   +0      T83     678
(907)   %       458     676
(914)   2
```

After the reset-add +0(T83)(678) operation, the A- and B-fields appear as shown in Figure 26.

The divisor is +27 (A-field). The dividend is +340, and is placed in B-field location 678 from T83 by the reset-add operation. Notice that the B-field address of the divide instruction should be the high-order position (676) of the dividend (in the B-field), rather than the units position of the B-field. After the divide operation %(458)(676) is completed, the A- and B-fields appear as shown in Figure 27.

The A-field is not changed by the operation. The dividend in the low-order position of the B-field is reduced to a remainder (with the standard sign of the dividend) that is less than the divisor. The quotient (with a standard sign) is developed and is in the high-order B-field positions.

The address of the remainder is the units position of the B-field. The address of the quotient is the units position of the B-field minus the number of divisor digit positions minus one.

Divide Operation — %(AAA)(BBB)

The major objective of a divide operation is to cause the dividend located in the low-order B-field positions to be divided by the divisor located in the A-field. A quotient is developed in the high-order B-field positions and the remainder (reduced dividend), if any, is located in the low-order B-field positions.

Algebraic sign control is used, and the quotient sign is produced in standard form. The sign of the remainder is the sign of the original dividend which is required in standard form for divide operations.

Figure 28 is a completed work sheet of the problem to be presented here. The divide operation is performed by a series of A- and B-cycles which accomplish the following minor objectives.

1. Prevent address modification and set the AUX-STARS on the first A- and B-cycle.
2. Compare (divide-compare) the divisor with the first partial dividend (B<A), and add 0 to the hundreds quotient digit position.
3. Shift the dividend field (and quotient field).
4. Compare (divide-compare) the divisor with the next partial dividend (A≤B<2A), and add 1 to the tens quotient digit position.
5. Simultaneously perform a subtract-A operation and compare (divide-compare) the divisor with the reduced partial dividend (B<A), and add 0 to the tens quotient digit position.
6. Shift the dividend field (and quotient field).
7. Compare (divide-compare) the divisor with the next partial dividend (B≤2A), and add 2 to the units quotient digit position.
8. Simultaneously perform a subtract-2A operation and compare (divide-compare) the divisor with the

A-FIELD ADDRESS				457	458
A-FIELD BEFORE OPERATION				2	B_7
A-FIELD AFTER OPERATION				2	B_7

INSTRUCTIONS: (900) $\frac{+}{0}$ T83 678
(907) % 458 676
(914) 2

B-FIELD ADDRESS			673	674	675	676	677	678
B-FIELD BEFORE OPERATION			0	0	0	3	4	B_0
B-FIELD AFTER DIVIDE-COMPARE			0	0	0	3	4	B_0
B-FIELD AFTER DIVIDE-COMPARE			0	1	0	3	4	B_0
B-FIELD AFTER SUBT A & DIV-COMP			0	1	0	0	7	B_0
B-FIELD AFTER DIVIDE-COMPARE			0	1	AB_2	0	7	B_0
B-FIELD AFTER SUBT 2A & DIV-COMP			0	1	AB_2	0	1	B_6

CYCLE LATCH	I	A	B	A	B	A	B	B	B	B Aux	A Aux	B Aux	A	B	B	B	A Aux	B Aux	A	B	B	B	B Aux	A Aux	B Aux	A	B	B	B	A Aux	B Aux	A	B	B	B	I
STAR	914	458	676	458	676	457	675	674	673	676	458	677	457	676	675	674	458	677	457	676	675	674	677	458	678	457	677	676	675	458	678	457	677	676	675	914
B-REGISTER	2	B_7	3	B_7	3	2	0	0	0	3	B_7	4	2	3	0	0	B_7	4	2	3	0	1	7	B_7	B_0	2	7	0	0	B_7	B_0	2	7	0	AB_2	2
A-REGISTER	8	B_7		B_7		2		C	C	C	B_7		2		C	C	B_7		2		C	C	C	B_7		2		C	C	B_7		2		C	C	0
TO INHIBIT	2	B_7	3	B_7	3	2	0	0	0	3	B_7	4	2	3	0	1	B_7	7	2	0	0	1	7	B_7	B_0	2	7	0	AB_2	B_7	B_6	2	1	0	AB_2	2
A-STAR	458	458		457		456					457		456				457		456					457		456				457		456				
B-STAR	676		676		675		674	673	672	677		676		675	674	673		676		675	674	673	678		677		676	675	674		677		676	675	674	
A-AUX-STAR	xxx	458																																		
B-AUX-STAR	yyy		676							677													678													
SIGN TRIGGER		On																							Off										

SET AUX-STARS
DIV-COMP OPR (B < A)
SET QUOT DIGIT 0
SHIFT DIVIDIEND FIELD
DIV-COMP OPR (A ≤ B < 2A)
SUBT A & DIV-COMP OPR (B < A)
SHIFT DIVIDEND FIELD
SET QUOT DIGIT 1
SET QUOT DIGIT 0
DIV-COMP OPR (B ≥ 2A)
SET QUOT DIGIT 2
SUBT 2A & DIV-COMP OPR (B < A)
SET QUOT DIGIT 0

Figure 28. %(AAA)(BBB) Divide Worksheet

reduced partial dividend (B<A), add 0 to the units quotient digit position, and end the divide operation.

The following discussion of cycle-by-cycle divide operations also includes mention of the various control latches and the cycles in which they are activated. The section under *Summary of Multiply-Divide Switching and Controls* contains a detailed summary and ILD reference and should be used as reference for exact timing and switching conditions.

On the last I-cycle of I-phase, the AUX-STAR SET latch is set to control the first minor objective. During the last I-cycle, the fields and registers appear as shown in Figure 29.

The AUX-STARS are set in the beginning of E-phase (on the first A- and B-cycle). E-phase begins with an A-cycle in which the units position of the divisor is read into the B-register and A-register, and transferred back to storage. The A-field sign is analyzed and the sign trigger is turned ON because the A-field sign is minus. No address modification occurs because TRANSFER STAR is activated under control of the AUX-STAR SET latch. The unmodified output of the STAR is read into the A-STAR and the A-AUX-STAR.

The following B-cycle is a normal B-cycle in which the high-order position of the dividend is read into the B-register and transferred back to storage. With the TRANSFER STAR line still activated, no address modification occurs. The unmodified output of the STAR is read into the B-STAR and the B-AUX-STAR.

Comparison of the divisor and the first partial dividend does not begin until the second B-cycle. This accomplishes the first minor objective of setting the AUX-STARS. Figure 30 shows the fiields and registers as they appear during the first B-cycle.

The first divide-compare operation begins on the second B-cycle when the divisor units position is compared with the units position of the first partial dividend. Alternate A- and B-cycles occur (with minus-one address modification) until the A-field WM is sensed.

On each A-cycle, the A-field position addressed is read into the B-register and A-register, and transferred back to storage. On each B-cycle, the corresponding B-field position is addressed, read into the B-register, and transferred back to storage.

The compare gate is activated on all B-cycles of a divide operation except on the first B-cycle of the divide operation, on the third B-cycle after the A-field WM, and on the B-auxiliary cycles during which the dividend field is shifted. This allows the divisor to be compared with the first partial dividend digit-by-digit,

A-FIELD						457	458
						2	B 7

B-FIELD		673	674	675	676	677	678
		0	0	0	3	4	B 0

STAR	914
A-STAR	458
B-STAR	676
A-AUX-STAR	XXX
B-AUX-STAR	YYY

Figure 29. Divide Fields and Registers Chart #1

beginning with the units position of the divisor and the units position of the first partial dividend (B-address).

Briefly, the comparison of the divisor with the partial dividend is accomplished by controlling two *A-Compare* latches and two *2A-Compare* latches. The two *A-Compare* latches are the B-LESS-THAN-A latch (B<A) and the SUBTRACT-A latch. The two *2A-Compare* latches are the B-LESS-THAN-2A latch (B<2A) and the SUBTRACT-2A latch. On each B-cycle that the compare gate is activated, these four latches are turned ON and OFF as determined by the output of the arithmetic unit divide-compare circuits. Operation details of the arithmetic divide-compare circuits are covered in the section under *Arithmetic Divide-Compare Unit.*

The objective of the divide-compare circuits is to determine the status of the partial dividend as compared with the divisor. At the completion of the divide-compare operation (and during the third B-cycle after the A-field WM), one of the following conditions exists:

1. The B-LESS-THAN-A latch is ON indicating that the partial dividend is less than the divisor (B<A).
2. The SUBT-A latch is ON and the SUBT-2A latch is OFF indicating that the partial dividend is equal to or greater than the divisor but less than twice the divisor (A≤B<2A).
3. The SUBT-2A latch is ON and the SUBT-A latch is ON or OFF, indicating that the partial dividend is equal to or greater than twice the divisor (B≥2A).

On the first B-cycle after the A-field WM is sensed, the M-D A-CYCLE ELIMINATE latch is turned ON. The

A-FIELD						457	458
						2	B 7

B-FIELD		673	674	675	676	677	678
		0	0	0	3	4	B 0

STAR	676
A-STAR	458
B-STAR	676
A-AUX-STAR	458
B-AUX-STAR	676

Figure 30. Divide Fields and Registers Chart #2

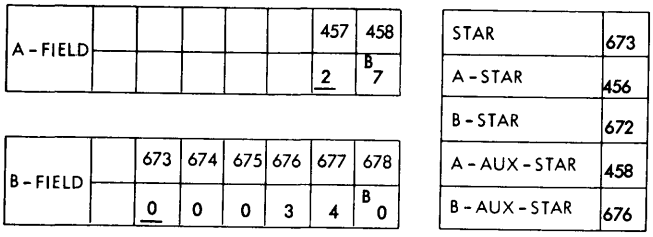


Figure 31. Divide Fields and Registers Chart #3

B-field position addressed should be a zero (high-order B-field positions must be set to zero prior to the divide instruction), which is transferred back to the B-field from the B-register. A-cycles are eliminated and continuous B-cycles are taken.

On the second B-cycle after the A-field WM, the X-POSITION latch is turned ON. The A-register is also reset and set to blank. The B-field position addressed should be a zero which is transferred back to the B-field from the B-register.

On the third B-cycle after the A-field WM, the quotient trigger is turned ON. During this cycle, the quotient field is increased by 0, by 1, or by 2 depending on the condition of the *A- and 2A-Compare* latches.

The quotient digit is increased by use of the qui-binary adder. The A-register is reset and set to blank at this time, and the output of the A-register translator is set to 0, to 1 or to 2 as determined by the *A- and 2A-Compare* latches. The B-register contains the contents of the quotient field position addressed. The output of the qui-binary adder is gated back to storage, thus increasing the quotient field position.

Because the first partial dividend (3) of the example being discussed is less than the divisor (27), the B-LESS-THAN-A latch is ON during the third B-cycle after the A-field WM. The A-register is reset and set to blank. The A-register translator output is set to 0 because the B-LESS-THAN-A, latch and QUOTIENT TRIGGER are ON. Because this is the first quotient-add cycle, the high-order quotient field position (also high-order B-field position) is being addressed and should be a zero. The output of the qui-binary adder is a zero which is gated back to storage as the high-order (hundreds) position of the quotient field.

Because the first partial dividend is less than the divisor, a shift in the dividend field is necessary. The DELTA B-AUXILIARY-CYCLE latch is set to allow the following cycle to be a B-auxiliary cycle in which to increase the B-AUX-STAR. Because plus-one address modification is required to increase the B-AUX-STAR, the M-D REVERSE SCAN latch is turned on late in this cycle. The REVERSE SCAN latch is used to activate *Modify Control Plus One* and deactivate *Modify Control Minus One.*

This accomplishes the divide-compare operation and the second minor objective is complete. The fields and registers appear as shown in Figure 31 during the third B-cycle after the A-field WM.

A B-auxiliary cycle must be taken to shift the dividend field. The B-AUX-CYCLE latch is set and the contents of the B-AUX-STAR is read into the STAR. The modified output (plus-one address modification because the M-D REVERSE SCAN latch is activated) of the STAR is read into the B-AUX-STAR thus increasing the address by one. The modified address is also read into the B-STAR, but is not used. B-STAR read-out is prevented on the B-auxiliary cycles in which the dividend field is shifted. Data in the B-field position addressed is transferred back to storage from the B-register. The A-register is also reset and set to blank.

The net result of the cycle is to increase the B-AUX-STAR by plus-one. The next partial dividend is 34, because the B-AUX-STAR addresses 677 as the units position of the partial dividend field. Because the quotient field is also located in the B-field, the position in which the next quotient digit is stored is effectively shifted one position to the right.

The DELTA A-AUX-CYCLE latch is set to initiate an A-AUXILIARY-CYCLE on the following cycle. Both an A-AUXILIARY-CYCLE and B-AUXILIARY-CYCLE are necessary to address the units positions of the divisor and the new partial dividend, and begin the next divide-compare operation.

The third minor objective is complete. During the B-AUXILIARY-CYCLE, the fields and registers appear as shown in Figure 32.

Another divide-compare operation must be initiated to compare the new partial dividend with the divisor. With the A-AUXILIARY-CYCLE latch set, the address in the A-AUX-STAR (units position of the divisor), is read into the STAR.

Data is read into the B-register and A-register, and transferred back to storage. The modified output of the STAR is read into the A-STAR but not into the A-AUX-STAR. The DELTA B-AUX-CYCLE latch is always set on A-AUXILIARY-CYCLES.

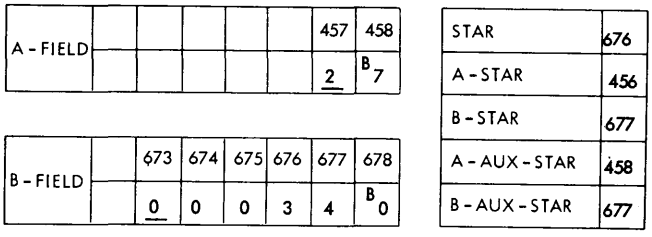


Figure 32. Divide Fields and Registers Chart #4

On the following cycle, the B-AUXILIARY-CYCLE latch is set and the units position of the new partial dividend is addressed. Data in the B-field position is read into the B-register and transferred back to storage. The compare-gate is active and the *A- and 2A-Compare* latches are controlled. On this cycle, the output of the STAR is read into the B-STAR but not into the B-AUX-STAR.

Alternate A- and B-cycles continue until the A-field WM is sensed. On each B-cycle, the compare-gate is active, and divide-compare operation progresses. Data is read out of storage on the A- and B-cycles and transferred back to storage unchanged.

On the first B-cycle after the A-field WM, the A-CYCLE-ELIMINATE latch is set, and only B-cycles are taken. B-field data in this position is transferred back to storage.

On the second B-cycle after the A-field WM, the X-POSITION latch is set, and the A-register is also reset and set to blank. Data in the B-field position is transferred back to storage. This is also the last cycle in which 34 (partial dividend) is compared with 27 (divisor).

On the third B-cycle after the A-field WM, the quotiont trigger is set, and the A-register is reset and set to blank. As a result of the divide-compare operation, the SUBTRACT-A latch is ON indicating that the partial dividend is equal to or greater than the divisor but less than twice the divisor ($A \leq 2 < A$). The quotient digit is increased by one and a subtract-A operation is initiated to occur simultaneously with a divide-compare operation.

The simultaneous subtract-A and divide-compare operation begins on an A-auxiliary cycle in which the units position of the divisor is read into the B-register and A-register, and transferred back to storage. Because subtraction is accomplished in the qui-binary adder by complement-adding, the carry trigger must be forced ON when the units positions of the two fields are complement-added. The carry trigger is forced ON during any A-auxiliary cycle of a divide operation although it is not used until the following B-auxiliary cycle.

On the following cycle, the B-AUX-CYCLE latch is set and the units position of the partial dividend field is read into the B-register. During B-cycles of the subtract operation, the complement-gate is activated. Input to the qui-binary adder from the A-register is entered singly (A-gate) and in complement form (complement-gate). With a true input from the B-register, the carry trigger ON, and the complement input from the A-register entered singly, the qui-binary adder output represent the units position of the reduced partial dividend.

Because comparison is performed digit-by-digit, it is possible to compare the divisor with the reduced partial dividend as it is being developed. This is done by activating the compare-gate and gating the qui-binary adder output both to storage and to the arithmetic unit divide-compare circuits.

The simultanenous subtract-A and divide-compare operation continues as alternate A- and B-cycles occur. On each A-cycle, the A-field data (divisor) is read into both the B-register and A-register, and transferred back to storage. On each B-cycle, the B-field data (partial dividend) is read into the B-register. The output of the qui-binary adder (reduced partial dividend) is directed to storage and to the divide-compare circuits.

On the first B-cycle after the A-field WM, the M-D A-CYCLE ELIMINATE latch is set and A-cycles are elimnated.

The quotient digit is increased by 1 on this cycle by use of the qui-binary adder. Because the SUBTRACT-A latch and QUOTIENT TRIGGER are ON and the SUBTRACT-2A latch is OFF, the A-register translator is set to 1. The quotient digit read into the B-register on this cycle is 0, and the A-register is set to blank. The qui-binary adder output is the first significant quotient digit (1) developed on this divide operation. Note that the quotient digit is gated back to storage before the divisor (27) is subtracted singly (subtract-A) from the partial dividend (34).

To initiate the simultaneous subtract-A and divide-compare operation requires addressing the units position of the divisor and units position of the partial dividend. The DELTA A-AUX-CYCLE latch is set to initiate an A-auxiliary cycle. Because a subtract-A operation is to occur, the A-latch is set to allow the A-gate to be activiated and the divisor subtracted singly from the partial dividend.

The second divide-compare operation is complete, and minor objective four has been accomplished. The fields and registers appear as shown in Figure 33 during the third B-cycle after the A-field WM.

On the second B-cycle after the A-field WM, the X-POSITION latch is set and the A-register is reset and set to blank. This is the last cycle of this simultaneous

A-FIELD						457	458
						2	B_7

B-FIELD		673	674	675	676	677	678
		0	1	0	3	4	B_0

STAR	674
A-STAR	456
B-STAR	673
A-AUX-STAR	458
B-AUX-STAR	677

Figure 33. Divide Fields and Registers Chart #5

subtract-A and divide-compare operation. This extra B-cycle is necessary to take any carries into the high-order of the partial dividend field into account. During complement-add operations, a carry also occurs out of the high-order of the partial dividend field. The high-order carry must be ignored because it would increase the next position addressd, which is the quotient digit. Only the results of the divide-compare operation should be effective in changing the quotient digit.

On any complement-add operation during a divide operation, a high-order carry occurs on the second B-cycle after the A-field WM, because the B-field is always equal to or greater than the A-field. This carry is not allowed to turn ON the carry trigger. No carry trigger gate is available.

On the third B-cycle after the A-field WM, the quotient trigger is turned ON. As a result of comparing the divisor (27) with the reduced partial dividend (7), the B-LESS-THAN-A latch is ON. This indicates the quotient digit in this shift should not be increased, and the A-register translator is set to 0. The A-register is reset and set to blank, the B-register contains the first significant quotient digit developed (1), and the carry trigger is OFF. Output from the qui-binary adder (1) is gated back to the tens position of the quotient digit field.

The DELTA B-AUX-CYCLE latch and the REVERSE SCAN latch are set in preparation for the following cycle. The B-AUX-STAR must be increased and a shift in both the dividend and quotient field effected.

With the simultaneous subtract-A and divide-compare operation complete, the fifth minor objective is finished. During the third B-cycle after the A-field WM, the fields and registers appear as shown in Figure 34.

To shift the dividend field (and quotient field), a B-auxiliary cycle must be taken. On the B-auxiliary cycle, the contents of the B-AUX-STAR is read into the STAR (B-STAR read-out is prevented). The output of the STAR is modified by plus-one (REVERSE SCAN latch is ON) and read into the B-AUX-STAR and B-STAR. The modified address in the B-STAR is not used. Data in the B-field

A - FIELD						457	458
						2	B_7

B - FIELD		673	674	675	676	677	678
		0	1	0	0	7	B_0

STAR	674
A - STAR	456
B - STAR	673
A - AUX - STAR	458
B - AUX - STAR	677

Figure 34. Divide Fields and Registers Chart #6

A - FIELD						457	458
						2	B_7

B - FIELD		673	674	675	676	677	678
		0	1	0	0	7	B_0

STAR	677
A - STAR	456
B - STAR	678
A - AUX - STAR	458
B - AUX - STAR	678

Figure 35. Divide Fields and Registers Chart #7

position addressed is read into the B-register and transferred back to storage. The A-register is also reset and set to blank.

Because the B-AUX-STAR is increased by one, the partial dividend field (and quotient field) is shifted one position to the right. B-field position 678 is addressed as the units position of the next partial divident (70). The next quotient digit developed is added to B-field position 675.

To begin the next divide-compare operation requires a following A-auxiliary cycle. Therefore, the DELTA A-AUX-CYCLE latch is set under control of the B-AUX-CYCLE latch and the B-LESS-THAN-A latch.

Minor objective six is complete. During the B-auxiliary cycle, the fields and registers appear as shown in Figure 35.

To determine whether a subtract operation is possible with the next partial dividend (70) requires a divide-compare operation. With the DELTA A-AUX-CYCLE latch activated from the previous cycle, the A-AUX-CYCLE latch is set to begin the divide-compare operation.

On the A-auxiliary cycle, the units position of the divisor is read into the B-register and A-register, and transferred back to storage.

On the B-auxiliary cycle, the units position of the partial dividend is read into the B-register. With the compare-gate activated, the divide-compare operation begins and the A- and 2A-Compare latches are controlled. Because the units position of the B-field is addressed on this cycle, other action is necessary.

The sign trigger is flipped if the units position of the B-field has a minus sign (dividend sign is minus). In the example being developed, the sign trigger was turned ON by the B-bit in the A-field (divisor sign is minus) during the first A-cycle. The sign trigger is turned OFF on this B-auxiliary cycle, indicating that the quotient sign is plus. The quotient sign is set in storage when the units position of the quotient field is addresed on the quotient trigger cycle.

Also, late in the same B-auxiliary cycle after the sign trigger is set, the END DIVIDE latch is set. Although the divide operation cannot end until the last partial dividend is less than the divisor, the END DIVIDE latch

stays ON until the end of the divide operation, and is used to indicate that the last dividend shift has occurred. Divide-compare and subtract operations continue until the B-LESS-THAN-A latch is activated on the quotient trigger cycle, indicating that the last partial dividend is less than the divisor. With the B-LESS-THAN-A latch, the QUOTIENT TRIGGER, and the END DIVIDE latch all activated, an I-E Change can finally be initiated.

The divide-compare operation progresses with alternate A- and B-cycles. A-field data is read into the B-register and A-register, and transferred back to storage. B-field data is read into the B-register, transferred back to storage, and compared with the A-register data by controlling the *A-and 2A-Compare* latches.

On the first B-cycle after the A-field WM, the M-D A-CYCLE-ELIMINATE latch is set. A-cycles are eliminated and B-cycles continue.

On the second B-cycle after the A-field WM, the X-POSITION latch is set and the A-register is reset and set to blank. This is the last cycle of divide-compare.

On the third B-cycle after the A-field WM, the quotient trigger is set and the A-register is reset and set to blank. The SUBTRACT 2-A latch is ON during this cycle indicating that the quotient digit should be increased by 2, and a simultaneous divide-compare and subtract-2A operation initiated.

The A-register is blank and the A-register translator output is set to 2 because the SUBTRACT-2A latch is ON. The B-register contains a zero quotient digit. Output from the qui-binary adder is a 2, which becomes the units quotient digit. At the same time the set-sign gate is activated and the quotient sign is set plus because the sign trigger is OFF.

Because the subtract-2A operation begins on the following cycle, the DELTA-A-AUX CYCLE latch and 2A latch are set. This allows an A-auxiliary cycle in which the subtract-2A operation begins, and activates the 2A-gate which allows the divisor to be subtracted doubly from the last partial dividend.

This completes minor objective seven. The fields and registers appear as shown in Figure 36 during the third B-cycle after the A-field WM.

The subtract-2A operation is basically the same as the subtract-A operation, except that the divisor is subtracted doubly from the partial dividend.

A-FIELD						457	458
						2	B_7

B-FIELD		673	674	675	676	677	678
		0	1	AB 2	0	7	B_0

STAR	675
A-STAR	456
B-STAR	674
A-AUX-STAR	458
B-AUX-STAR	678

Figure 36. Divide Fields and Registers Chart #8

A-FIELD						457	458
						2	B_7

B-FIELD		673	674	675	676	677	678
		0	1	AB 2	0	1	B_6

STAR	675
A-STAR	456
B-STAR	674
A-AUX-STAR	458
B-AUX-STAR	678

Figure 37. Divide Fields and Registers Chart #9

The STAR is addressed from the A-AUX-STAR and the units position of the divisor is read into the B-register and A-register, and transferred back to storage. Also, the carry trigger is forced ON for use on the following B-auxiliary cycle. On the B-auxiliary cycle, the units position of the partial dividend is read into the B-register. With the complement-gate activated, the input from the A-register (divisor) is in complement form and enters the qui-binary adder doubly. The input from the B-register (partial dividend) is in true form and the carry trigger is ON. The output of the qui-binary adder is the reduced partial dividend which is gated back to storage, and also to the arithmetic unit divide compare circuits. With the compare-gate activated on the B-cycles, the divide-compare operation occurs simultaneously. The simultaneous subtract-2A and divide-compare operations continue with alternate A- and B-cycles.

On the first B-cycle after the A-field WM, the M-D A-CYCLE-ELIMINATE latch is set and only B-cycles continue.

On the second B-cycle after the A-field WM, the X-POSITION latch is set and the A-register is reset and set to blank. This is the last cycle of the subtract-2A and divide-compare operation. An adder carry occurring in this cycle is not allowed to turn ON the carry trigger.

On the third B-cycle after the A-field WM, the quotient trigger is set and the A-register is reset and set to blank. For the example presented here, the partial dividend has been reducd to less than the divisor and becomes the remainder, retaining the sign of the original dividend. Therefore, the B-LESS-THAN-A latch is ON and the A-register translator output is set to zero. The quotient digit in this shift is not changed, as a zero is added to the units quotient digit position. The quotient sign is gated back to the units quotient digit position from the B-register.

Because the B-LESS-THAN-A latch is on during the quotient trigger cycle, and the END DIVLDE latch is ON, an I-E Change is initiated ending the divide operation.

This completes the last minor objective. During the last B-cycle of the divide operation, the fields and registers appear as shown in Figure 37.

Note that the quotient field has a standard sign. The address of the quotient field (units position) is the address of the dividend field (units position) minus the number of divisor field positions, minus one. The remainder is the last reduced partial dividend with the standard sign of the original dividend.

Figure 28 is a work sheet showing the cycle-by-cycle results of the divide problem just discussed.

A zero dividend is handled in a normal manner. The AUX-STARS are set on the first A- and B-cycles, followed by continuous divide-compare operations. On each divide-compare operation the divisor is compared to zero, and the B-LESS-THAN-A latch is ON during each quotient trigger cycle. A quotient digit of 0 is added, the dividend (and quotient) field is shifted, and another divide-compare is initiated.

When the units position of the dividend is addressed, the END DIVIDE latch is set. The following divide-compare operation results in setting the B-LESS-THAN-A latch. Because the END DIVIDE latch, the B-LESS-THAN-A latch and the QUOTIENT TRIGGER are ON, an I-E Change is initiated ending the zero dividend divide operation.

A zero divisor results in an arithmetic overflow and an I-E Change. The divide operation with a zero divisor begins normally. The AUX-STARS are set on the first A- and B-cycles, and are followed by a normal divide-compare operation.

As a result of comparing the divisor with the first partial dividend, the SUBTRACT-2A latch is set (this is also true with a zero divisor and a zero partial dividend). The quotient digit is increased by 2, and a simultantous subtract-2A and divide-compare operation is initiated. Tht second divide-compare operation also results in setting the SUBTRACT-2A latch and initiating another simultaneous subtract-2A and divide-compare operation.

This pattern continues to repeat until an arithmetic overflow is indicated on the fifth divide-compare (fourth simultaneous subtract-2A and divide-compare operation). At the beginning of the fifth quotient trigger cycle, the quotient digit is an 8 and the SUBTRACT-2A latch is ON. Because four continuous subtract-2A operations are the most that can occur on a correct divide operation, these conditions indicate a divide by zero condition or a programming error. Quotient trigger, B-register 8, and SUBTRACT-2A latch are switched to cause an arithmetic overflow and to set the OVERFLOW latch. The quotient digit 8 is also increased by 2 on this cycle resulting in a final quotient digit of 0. The carry is not used.

Overflow in turn signals an I-E Change ending the divide operation with a zero divisor. The quotient field is zero and has no sign.

The OVERFLOW latch is normally set by a divide-by-zero condition on a divide operation, but can also be set by certain types of programming errors.

For example, assume a dividend of (98) and a divisor of 2. If the B-field address of the divide instruction was set incorrectly, it would be possible to compare the divisor (2) with the complete dividend (98) on the first divide-compare operation. Twenty-four subtract-2A operations would be possible; however, an Overflow and I-E Change would occur on the fifth quotient trigger cycle.

A branch-on-overflow (B III d with a Z d-character) following a divide instruction (% AAA BBB) can normally be used to interrogate for a zero divisor (assuming correct programming of the B-field address).

A 1401 data processing system does not check for programming errors. If the B-field is too small, an incorrect quotient may be developed as this condition is not checked by the machine. The programmer must provide a B-field length equal to the sum of the divisor digit positions plus the dividend digit positions, plus one. The B-address of the divide instruction should be the high-order position of the dividend field rather than the units position of the B-field.

Functional Components

A-Auxiliary Address Register

An A-Auxiliary Address Register (A-AUX-STAR) is included with the Calculate Option to simplify addressing of the A-field. Because the A-field must be forward-scanned many times during a multiply or divide operation, it is desirable to be able to change from the high-order position A-field address to units position A-field address without a series of reverse-scan cycles.

With the units position A-field address stored in the A-AUX-STAR, it is possible to address the units position of the A-field on any E-phase cycle by reading the A-AUX-STAR contents into the STAR. The modified output of the STAR is then read into the A-STAR but not into the A-AUX-STAR. After the cycle is complete, the A-STAR contains the tens position A-field address to be used on the next A-cycle, but the address stored in the A-AUX-STAR is not changed.

The A-AUX-STAR is a storage address register essentially identical to the A-STAR. Three positions of latch storage are used to store the three character units position A-field address. The A-AUX-STAR contains: eight latches in the units positions (Units 1, Units 2, Units 4, Units 8, Units CD, Units A, Units B, and Units CZ); five latches in the tens position (Tens 1,

Tens 2, Tens 4, Tens 8, and Tens CD); and eight latches in the hundreds position (Hundreds 1, Hundreds 2, Hundreds 4, Hundreds 8, Hundreds CD, Hundreds A, Hundreds B, and Hundreds CZ).

Figure 38 shows the switching necessary to read in or read out of the A-AUX-STAR. Unlike the A-STAR, the units position A-field address is not read into the A-AUX-STAR during I-phase. During the first E-phase cycle of a multiply or divide operation, a normal A-cycle is taken in which the A-STAR address is transferred to the A-AUX-STAR. Address modification is prevented on this first A-cycle by activating TRANSFER STAR.

Because the STAR output during the first A-cycle is not modified, the output of the modifier may be read into both the A-AUX-STAR and A-STAR. Therefore, at the end of the first A-cycle, the units position A-field address is in the A-STAR and A-AUX-STAR thus accomplishing the address transfer.

The AUX-STAR SET latch, which controls reading into the A-AUX-STAR, is activated from 090 time of the last I-cycle to 090 time of the first B-cycle on multiply or divide operations.

Figure 38 illustrates how the AUX-STAR SET latch controls the A-AUX-STAR read-in. Because the A-AUX-STAR SET latch is not activated on other A-cycles, the address read into the A-AUX-STAR during the first A-cycle remains the same throughout the multiply or divide operation.

Whenever the A-field must be scanned, the units position A-field address in the A-AUX-STAR is gated into the STAR. The read-out of the A-AUX-STAR is controlled by the DELTA A-AUX-CYCLE latch. Therefore, as the multiply or divide operation progresses and the A-field is to be scanned, an A-auxiliary cycle is initiated by setting the DELTA A-AUX-CYCLE latch. As shown in Figure 38, the DELTA A-AUX-CYCLE latch not only allows gating the A-AUX-STAR contents to the STAR, but also blocks the normal A-STAR read-out.

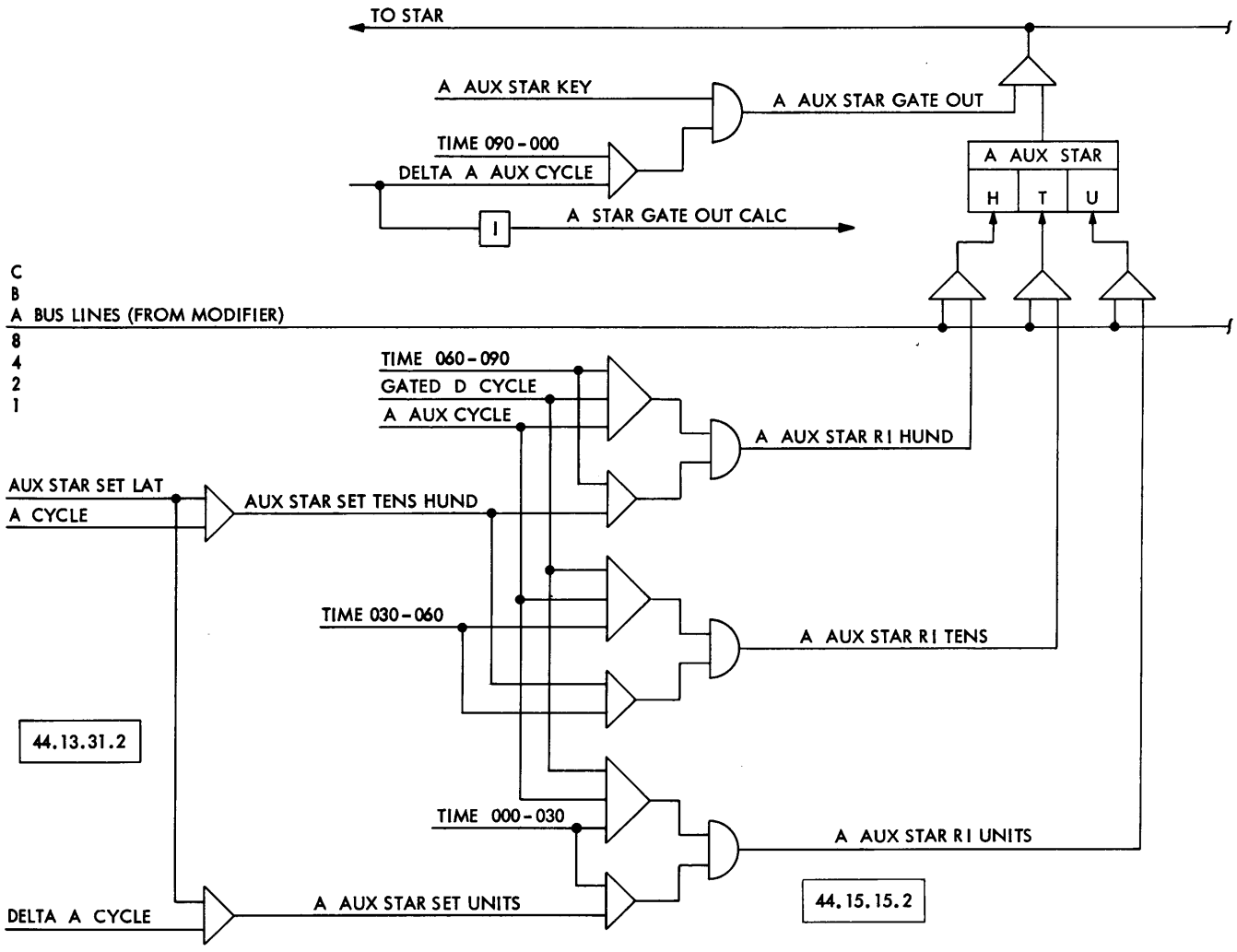


Figure 38. A-Aux Star and Controls

The address in the A-AUX-STAR may be displayed manually by pressing the A-AUX-STAR key. This allows the address in the A-AUX-STAR to be gated into the STAR as shown by the switching in Figure 38. Because the contents of the STAR are visible on the 1401 data processing system console, the contents of the A-AUX-STAR may be viewed.

Alter mode is used to change the address in the A-AUX-STAR manually. *Modify control transfer* is activated by the mode switch in alter mode. With the A-auxiliary-key light ON (set on by pressing the A-AUX-STAR key which sets the A-AUX-CYCLE latch), the address entered into the STAR from the manual address switches is gated through the modifier without modification (*modify control transfer is activated*) and entered into the A-AUX-STAR. This is controlled by ALTER and A-AUX-CYCLE switched with time as shown in Figure 38.

B-Auxiliary Address Register

A B-Auxiliary Address Register (B-AUX-STAR) is also included in the Calculate Option to simplify addressing of the B-field. During a multiply or divide operation, it is necessary to forward-scan the B-field many times beginning with different B-field positions. Therefore, it is desirable to be able to address the correct B-field position at any time without taking extra cycles.

The B-AUX-STAR is used to store the address at which each forward-scan of the B-field should begin. Provision is also made to decrease or increase the address in the B-AUX-STAR by one, so that the different starting addresses desired during multiply or divide operations can be developed. With this arrangement, the position at which the forward scan of the B-field should begin can be determined by addressing storage from the B-AUX-STAR.

On the cycle that storage is addressed from the B-AUX-STAR, the output of the STAR is modified by minus- or plus-one depending on the objective of the B-auxiliary cycle in progress. The modified output on the bus lines may be read into both the B-STAR and B-AUX-STAR, or only the B-STAR, or only the B-AUX-STAR. Each situation is discussed later in this section.

The B-AUX-STAR is also a three position register using latches to store the three character B-field addresses. The B-AUX-STAR contains eight latches in the units position (Units 1, Units 2, Units 4, Units 8, Units CD, Units A, Units B, and Units CZ); five latches in the tens position (Tens 1, Tens 2, Tens 4, Tens 8, and Tens C); and eight latches in the hundreds position (Hundreds 1, Hundreds 2, Hundreds 4, Hundreds 8, Hundreds CD, Hundreds A, Hundreds B, and Hundreds CZ).

Figure 39 shows the switching necessary to read-in and read-out of the B-AUX-STAR. During I-phase, the B-field address of the multiply or divide instruction is read into only the B-STAR. On the first B-CYCLE of either instruction, the B-STAR address is transferred to the B-AUX-STAR under control of the AUX-STAR SET latch (activated at 090 time of last I-cycle until 090 time of first B-cycle during multiply or divide operations). TRANSFER STAR is also activated to prevent address modification on the first A- and B-cycle. By reading the unmodified STAR output into the B-STAR and B-AUX-STAR, the B-field address is transferred to the B-AUX-STAR. This switching is shown in Figure 39. AUX-STAR-SET latch is switched with DELTA B-CYCLE and B-CYCLE to allow B-AUX-STAR read-in.

A Delta B-auxiliary cycle is required to address storage from the B-AUX-STAR whenever the B-field is to be scanned or shifted. Note also, that the B-STAR is not allowed to read out on a B-auxiliary cycle. Other details of B-auxiliary-cycle control are covered in the section under *B-Auxiliary Cycle Control.*

During multiply operations, the address in the B-AUX-STAR is used at the beginning of each B-field scan when an add-A or add-2A operation is initiated. The modified output (minus-one address modification) of the STAR is read into the B-STAR but not into the B-AUX-STAR. This allows the B-AUX-STAR to retain its B-field address until a product field shift is required. The B-STAR address is changed to the next lower B-field address, so that the scan may continue by addressing storage from the B-STAR. As shown in Figure 39, no B-AUX-STAR read-in is provided at this time, and the B-STAR read-in is not blocked. The B-STAR read-in is blocked on a B-auxiliary cycle only if the MULTIPLIER latch and B-AUXILIARY-CYCLE latch are activated at the same time.

During multiply operations whenever the multiplier digit addressed is a zero, the B-field (product field) must be shifted. The B-field shift is accomplished by modifying the B-AUX-STAR address by minus-one. The modified output of the STAR is read into the B-AUX-STAR but not into the B-STAR. This allows the B-STAR to retain the address of the next possible multiplier digit. The B-AUX-STAR address is changed to the next lower B-field address, so that on the next add-A or add-2A operation, the multiplicand is added to the next lower B-field position. In effect, the product field has been shifted. This is accomplished by switching MULTIPLIER latch and B-AUX-CYCLE as shown in Figure 39.

During divide operations, the address in the B-AUX-STAR is used at the beginning of each B-field scan on other than the first divide-compare operation, when a divide-compare or simultaneous subtract and divide-compare operation is initiated.

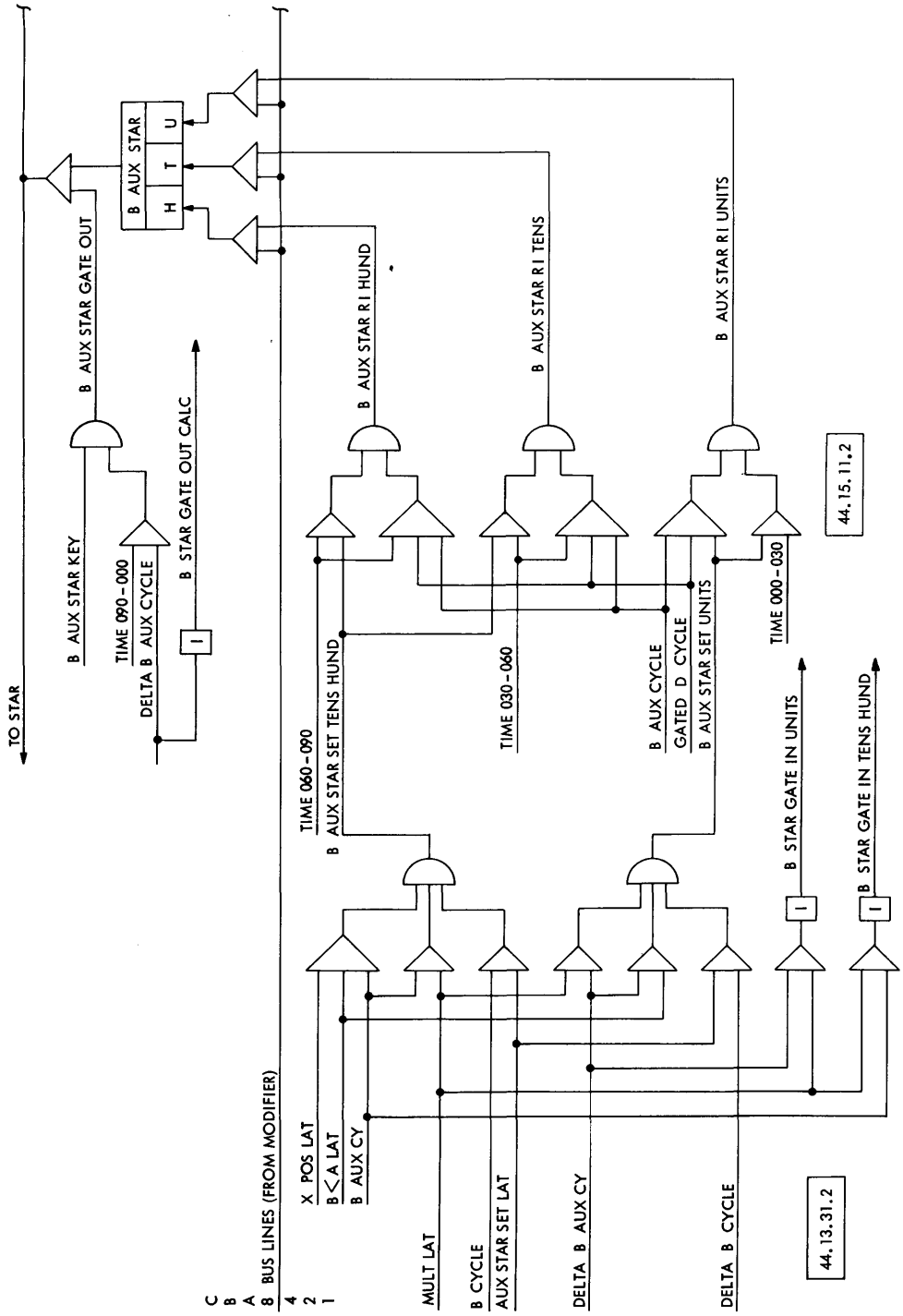


Figure 39. B-Aux Star and Controls

The modified output (minus-one address modification) of the STAR is read into the B-STAR but not into the B-AUX-STAR. This allows the B-AUX-STAR to retain its B-field address until a dividend field shift is required. The B-STAR address is changed to the next lower B-field address, so that the scan may continue by addressing storage from the B-STAR. Figure 39 shows that no switching is provided for B-AUX-STAR read-in at this time, and the B-STAR read-in is not blocked.

During divide operations, when a divide-compare operation indicates that the partial dividend is less than the divisor (B<A), the dividend field (and quotient field) must be shifted. Both shifts are accomplished by modifying the B-AUX-STAR address by plus-one. The M-D REVERSE SCAN latch is set to force plus-one address modification. The modified output of the STAR is read into the B-AUX-STAR and into the B-STAR, although the modified address in the B-STAR is not used. The B-AUX-STAR address, after modification, has been increased by plus one so that a larger dividend field is scanned on the next divide-compare or simultaneous subtract and divide-compare operation. As shown in Figure 39, the switching to accomplish the B-AUX-STAR read-in is X-POSITION latch, B-LESS-THAN-A latch, and B-AUX-CYCLE.

The contents of the B-AUX-STAR may be displayed manually on the 1401 DPS console, by pressing the B-AUX-STAR key and gating the address into the STAR.

To change the B-AUX-STAR address manually, the mode switch is set to ALTER MODE. Modify-control-transfer is activated by the mode switch in ALTER MODE. With the B-AUX-STAR light on (set on by pressing the B-AUX-STAR key which sets the B-AUX-CYCLE latch), the address entered into the STAR from the manual address switches is gated through the modifier without modification (modify-control-transfer is activated), and entered into the B-AUX-STAR. This is controlled by ALTER and B-AUX-CYCLE switched with time as shown in Figure 39.

Arithmetic Doubler Unit

Qui-binary doubler circuits have been added to the basic arithmetic unit between the A-register translator and A-register true/complement switching. Input to the A-register true/complement switching is under control of an A-gate and a 2A-gate.

If the A-gate is activated, the input to the A-register true/complement switching comes directly from the A-register in single form. If the 2A-gate is activated, the input to the A-register true/complement switching comes from the qui-binary doubler circuits in doubled form. Data in the A-register is available to the divide-compare circuits at all times in single or double form without A- or 2A-gating.

With the addition of the doubler circuits, multiply-add or divide-subtract operations may be performed singly or doubly. Whenever possible, the doubled entry is used to save processing time.

Figure 40 is an example of qui-binary doubling to be discussed here. Figure 41 is a combination figure showing the addition of the doubler circuits and the divide-compare circuits to the arithmetic unit. Reference to these figures is helpful in understanding how doubling is accomplished, and the relationship of the various units.

The doubled quinary value is determined by switching the A-register binary bit with the A-register quinary bit. The doubled binary value is determined by the status of the *doubler carry trigger*. When the

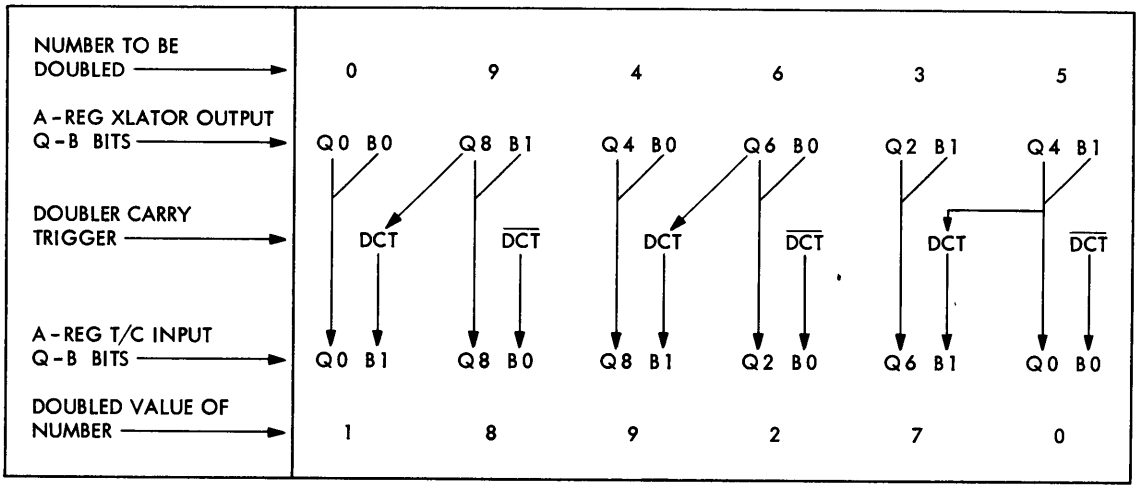


Figure 40. Example of Qui-Binary Doubling

doubler carry trigger is ON, the doubled binary value is one (A-reg 1); likewise, when the doubler carry trigger is OFF, the doubled binary value is zero (A-reg Not 1).

Control of the binary value by a carry trigger is possible because any digit value doubled is even, unless a carry occurs from the previous digit position. Because even numbers contain a binary zero and odd numbers contain a binary one, the doubler carry trigger can be used to store the doubler carries, and thus to determine the binary bit value.

Figure 40 illustrates how 094635 is doubled. Notice that the doubled quinary bits are always determined by switching the A-register quinary and binary bits together. The qui-binary doubler shown in Figure 41 is located on ALD 44.18.11.2.

Whenever the doubler carry trigger is OFF (Figure 40), the binary input to the A-register true/complement switching is zero (A-reg Not 1). Whenever the A-register value is five or greater, a carry will occur during doubling. Therefore, as shown in Figure 40, a Q8 (8 or 9), or a Q6 (6 or 7), or a Q4 and B1 (5), are switched to turn ON the doubler carry trigger. Figure 41 also shows the control of the doubler carry trigger (ALD 44.18.21.2), and its position in the arithmetic unit. It should also be noted that the doubler carry trigger is reset during each cycle if no carry occurs.

The true/complement binary entry shown in Figure 41 (ALD 44.18.31.2) also illustrates other switching used to set the output of the A-register translator.

During divide operations, the *set A-register translator output to 2* and *set A-register translator output to 1* circuits are used to develop the quotient digit. When the divide-compare operation indicates that the partial dividend is equal to or greater than twice the divisor ($B \geq 2A$), then QUOTIENT TRIGGER and SUBTRACT-2A latch are switched to set the A-register translator output to 2. This causes a 2 to be added to the quotient digit in the B-register. When the divide-compare operation indicates that the partial dividend is equal to or greater than the divisor but less than twice the divisor ($A \leq B < 2A$), then QUOTIENT TRIGGER, SUBTRACT-A latch, and NOT SUBTRACT-2A latch are switched to set the A-register translator output to 1. This increases the quotient digit in the B-register by 1.

For multiply operations, the *set A-register translator output to 1* is used to decrease the greater than two multiplier digit by 2. Because the complement gate is also activated at this time, the input to the A-register true/complement switching is actually on 8 (9's complement of 1), which is added to the B-register digit. The resultant qui-binary adder output is a carry and a digit two less than the B-register digit. Because the carry is not used, the net result is to reduce the greater than two multiplier digit by 2. The actual switching is shown in Figure 41 by PROCESS, NOT B-AUXILIARY-CYCLE, MULTIPLIER latch and B-REGISTER GREATER THAN TWO. (Note – Force 8-2 is used to set a zero in storage if the multiplier digit is a 1 or 2).

In addition, Figure 41 also shows a circuit to set the A-register translator output to 0 if NOT A-GATE; NOT 2A-GATE, NOT SET TO 1, and NOT SET TO 2 are activated. This is used during divide, when a divide-compare operation is initiated alone. The zero input to the A-register true/complement switching allows the B-register digit to pass through the qui-binary adder enroute to the divide-compare circuits without being changed.

Although it is not shown in Figure 41, it should be noted that the A-gate is also activated by NOT MULTIPLY OPERATION and NOT DIVIDE OPERATION. This blocks the *set A-register translator output to 0*, and allows a normal single value input into the qui-binary adder from the A-register on other than multiply and divide operations.

B-Register Digit Generator

During multiply operations, it is necessary to know the relative size of the multiplier digit at various stages of the operation. A *B-register digit generator*, as shown in Figure 42, is used for this purpose. A brief description of the uses of the various lines produced by the B-register digit generator is discussed in this section.

A multiplier digit of zero or blank indicates that the product field must be shifted (initiate a B-auxiliary cycle), or the multiply operation ended (I-E Change) if a WM is present.

A multiplier digit of one (Not B-reg greater than 2) signals the multiplier digit should be set to zero (Force 8-2), and an add-A operation initiated (B-reg 1). A multiplier digit of two (Not B-reg greater than 2) signals the multiplier digit should be set to zero (Force 8-2), and an add-2A operation initiated (B-reg 2 or greater).

A multiplier digit of two or greater causes the multiplier digit to be reduced by two (complement gate and set A-register translator to 1), and an add-2A operation initiated (B-reg 2 or greater).

A significant digit indicates other than a zero or blank multiplier digit, which ends the search for a multiplier digit, and initiates an add operation (A-auxiliary cycle).

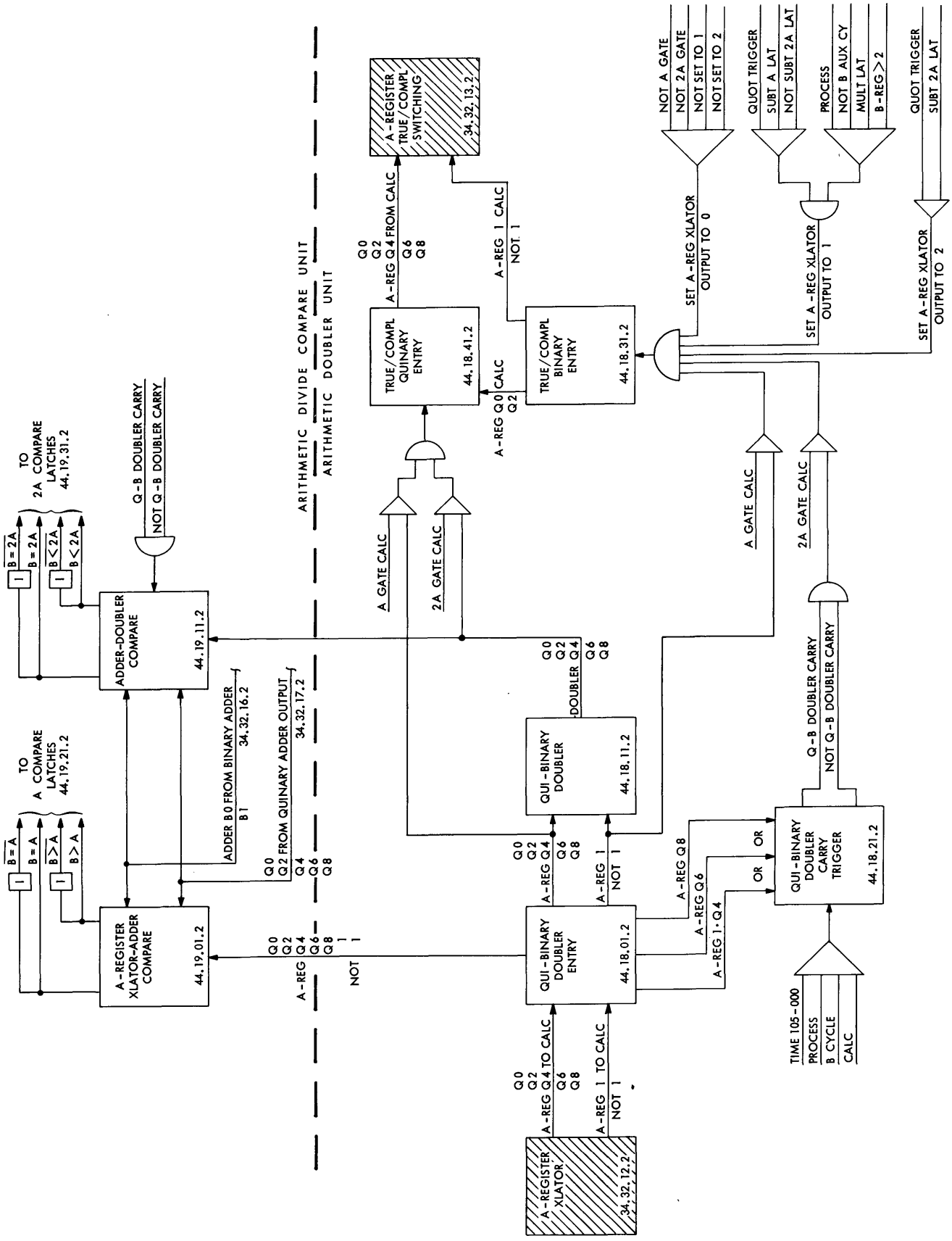


Figure 41. Arithmetic Doubler Unit and Divide-Compare Unit

Other multiply controls affected by the B-register digit generator are the ARITH DIGIT GATE and ARITH CHECK. The various B-register signals and the switching to generate them are shown in Figure 42.

Arithmetic Divide-Compare Unit

Divide-compare circuits have been added to the arithmetic unit for use during divide operations. They are used to make a digit-by-digit comparison of the divisor with the partial dividend under consideration. Divide-compare operations may occur alone, or simultaneously with a divide-subtract operation.

The net result of a divide-compare operation is to determine whether: the partial dividend is less than the divisor ($B<A$); the partial dividend is equal to or greater than the divisor but less than twice the divisor ($A\leq B<2A$); or the partial dividend is equal to or greater than twice the divisor ($B\geq 2A$).

Because the comparison is so thorough, each step in the divide operation can be accurately determined. If $B<A$, the quotient digit is not increased, and a dividend field shift is initiated; if $A\leq B<2A$, the quotient digit is increased by one and a simultaneous subtract-A and divide-compare operation is initiated; and if $B\geq 2A$, the quotient digit is increased by two and a simultaneous subtract-2A and divide-compare operation is initiated.

Figure 41 shows the arithmetic divide-compare unit, and the input and output of the circuitry. As shown, the divide-compare unit consists of an A-register translator, adder compare unit, and an adder-doubler compare unit.

Input to the A-register translator-adder compare is from the A-register translator and from the output of the qui-binary adder. The outputs of the A-register translator-adder compare unit control the A-compare latches.

Input to the adder-doubler compare unit is from the qui-binary doubler and from the qui-binary adder. The outputs of the adder-doubler compare unit control the 2A-compare latches.

Input to the divide-compare circuitry from the A-register translator and qui-binary doubler appears logical because the divisor (A-field) is compared with the partial dividend singly and doubly. However, the inputs from the qui-binary adder may not appear obvious. It may be well to note that there is no input to the divide-compare circuitry from the B-register. Because a divide-compare operation may occur alone or simultaneously with a divide-subtract operation, the qui-binary adder is used in both cases to supply an input to the divide-compare circuitry from the dividend field.

Consider a simultaneous divide-compare and divide-subtract operation. The objective of the divide-subtract operation is to subtract the divisor from the partial dividend. Therefore, the output of the qui-binary adder is the reduced partial dividend. Rather than store the reduced partial dividend in storage until the next divide-compare operation, it is possible to simultaneously compare the divisor with the reduced partial dividend as it is developed and sent to storage. This arrangement is possible because the divide-compare

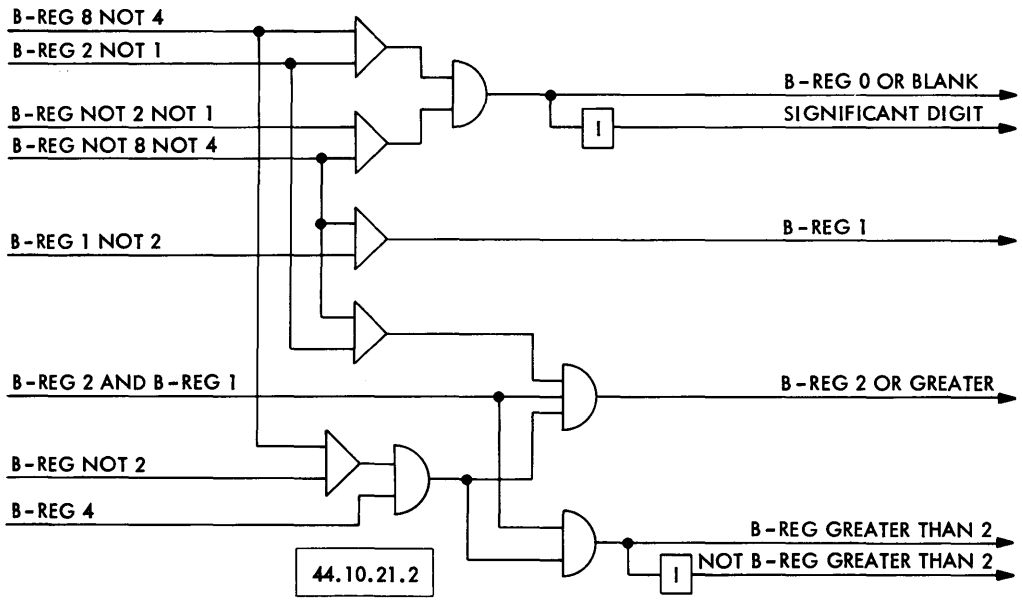


Figure 42. B-Register Digit Generator

operation is accomplished digit-by-digit, and it is done late in the cycle (Time 075-090) when the qui-binary adder output is available.

For a divide-compare operation performed alone, the A-register translator output is set to zero, and the partial dividend is read into the B-register. Therefore, the output of the qui-binary adder during a divide-compare operation alone is effectively the partial dividend from the B-register. The switching to set the A-register translator output to 0 is *Not A-Gate, Not 2A-Gate, Not Set to 1,* and *Not Set to 2* as shown in Figure 41.

Four outputs (B=A, $\overline{B=A}$, B<A, and $\overline{B<A}$) from the A-register translator-adder compare unit are used to control the A-COMPARE latches. Four outputs (B=2A, $\overline{B=2A}$, B<2A, and $\overline{B<2A}$) are also available to control the 2A-COMPARE latches. Figure 43 shows the A- and 2A-COMPARE latches and the sets and resets which control them.

Outputs are used from only three of the four compare latches to control the divide operation. The fourth COMPARE latch (B-LESS-THAN-2A latch) is used only to control the SUBTRACT-2A latch and is not used directly in the divide operation.

As mentioned previously, one of three possible conditions (B<A or A$\leq$B<2A, or B$\geq$2A results from a divide-compare operation. On the third cycle after the A-field WM, the B-LESS-THAN-A latch is ON if B<A; the SUBTRACT-A latch is ON and the SUBTRACT-2A latch is OFF if A$\leq$B<2A; and the SUBTRACT-2A latch is ON and the SUBTRACT-A latch is ON or OFF if B$\geq$2A.

During the divide-compare operation, the four compare latches are turned ON and OFF as each digit is compared. Figure 44 is a sequence chart of a divide-compare operation in which a divisor of 81485268 (A-field) is compared with a partial dividend of 98180276 (B-field). If the switching to set and reset the compare latches is reduced to the simplest form, it appears as follows:

B-LESS-THAN-A LATCH

Set: (B<A)
Reset: (B>A)

SUBT-A LATCH

Set: $(B = A)\,\overline{(B<A\ \text{Lat})} + (B>A)$
Reset: (B<A)

B-LESS-THAN-2A LATCH

Set: (B<2A)
Reset: (B>2A)

SUBT-2A LATCH

Set: $(B = 2A)\,\overline{(B<2A\ \text{Lat})} + (B>2A)$
Reset: (B<2A)

This switching is also shown in Figure 44 and should aid in understanding how the comparison is accomplished digit-by-digit. The sequence chart (Figure 44) is a detailed chart of digit-by-digit comparison which is self-explanatory. The following paragraphs cover a general discussion of divide-compare.

On any cycle of a divide-compare operation, if the B-register digit is less than the A-register digit, then the whole B-field *up to that position* is less than the whole A-field *up to that position,* regardless of the low-order field position. Therefore, the B-LESS-THAN-A latch is set.

On any cycle of a divide-compare operation, if the B-register digit is equal to the A-register digit and the low-order B-field positions are not less than the low-order A-field positions, then the whole B-field *up to that position* is equal to or greater than the whole A-field *up to that position.* In other words, the B-LESS-THAN-A latch must be OFF indicating that the low-order B-field positions are equal to or greater than the low-order A-field positions. Therefore, if (B=A) $(\overline{B<A\ \text{Lat}})$, the SUBTRACT-A latch is set.

On any cycle of a divide-compare operation, if the B-register digit is greater than the A-register digit, then the whole B-field *up to that position* is greater than the whole A-field *up to that position,* regardless of the low-order field positions. Therefore, the SUBTRACT-A latch is set.

The status of the B-field as compared to the status of twice the A-field is doubled. A doubler carry may occur and must be considered in the comparision. Therefore, an extra B-cycle is required to complete the comparison. The divide-compare operation ends on the second B-cycle after the A-field WM. Notice that the compare gate shown in Figure 44 is activated for the first and second B-cycle after the A-field WM. The results of the comparison are used on the third B-cycle after the A-field WM.

On the second B-cycle after the A-field WM of a divide-compare operation, if the B-register digit is less than twice the A-register, than the whole B-field is less than twice the whole A-field regardless of the low-order field positions. Therefore, the SUBTRACT-2A latch is not set.

On the second B-cycle after the A-field WM of a divide-compare operation, if the B-register digit is equal to twice the A-register digit and the low-order B-field positions are not less than twice the low-order A-field positions, the whole B-field is equal to or greater than twice the whole A-field. In other words, the B-LESS-THAN 2A latch must be OFF indicating that the low-order B-field positions are equal to or greater

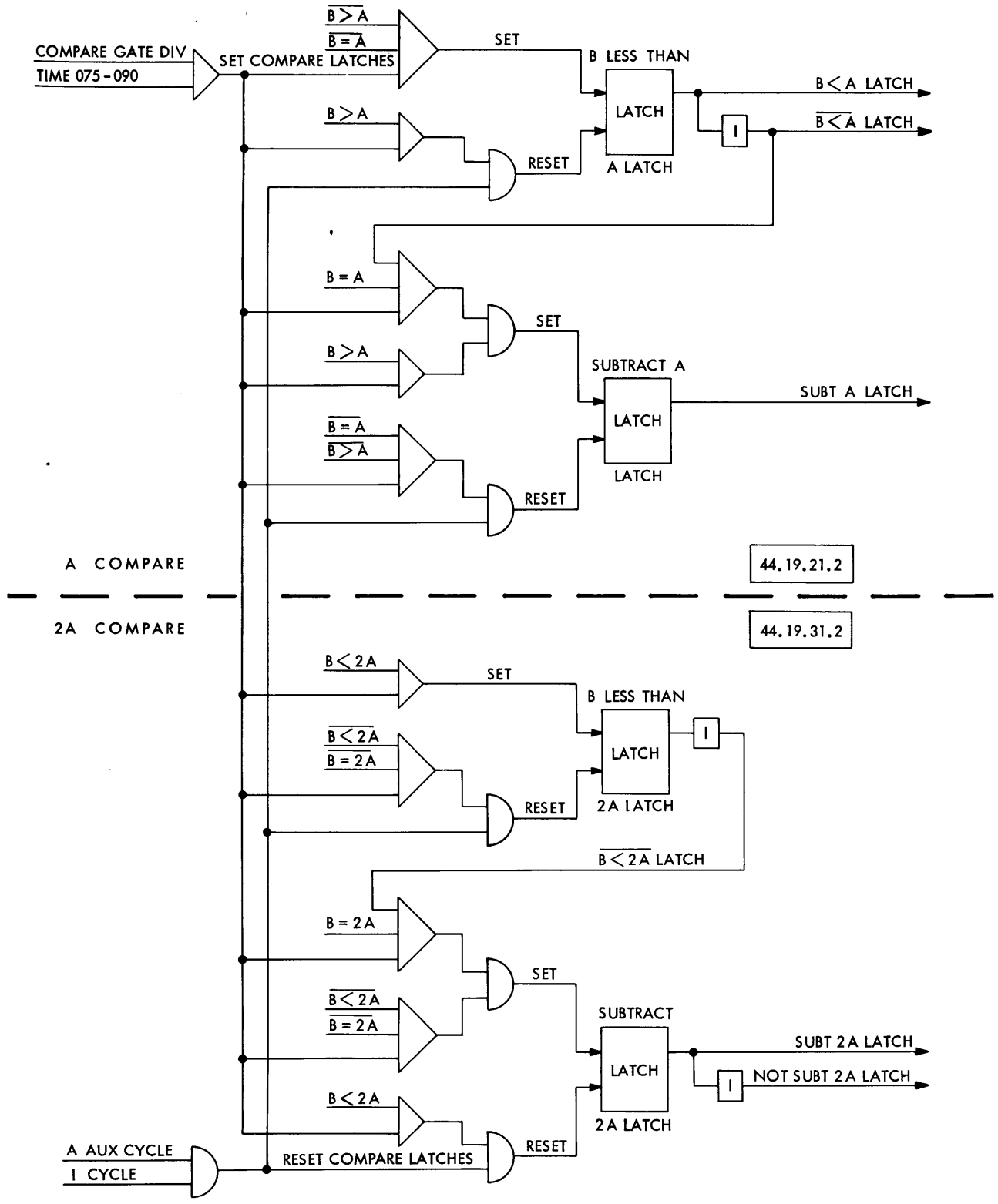


Figure 43. A and 2A Compare Latches

Figure 44. Divide-Compare Sequence Chart

B-FIELD 0 0 0 0 0 0 0 0 0 9 8 1 8 0 2 7 6
A-FIELD 8 1 4 8 5 2 6 8

	SIGNAL NAME	LOGIC	A	B	A	B	A	B	A	B	A	B	A	B	A	B	A	B	B	B	A Aux
			3 6 9	3 6 9	3 6 9	3 6 9	3 6 9	3 6 9	3 6 9	3 6 9	3 6 9	3 6 9	3 6 9	3 6 9	3 6 9	3 6 9	3 6 9	3 6 9	3 6 9	3 6 9	3 6 9
	B-REGISTER DIGIT			6		7		2		0		8		1		8		9	0		
	A-REGISTER DIGIT			8		6		2		5		8		4		1		8	0		
	DOUBLED A-REGISTER DIGIT			6		3		5		0		7		9		2		6	1		
	A-REG XLATOR-ADDER COMPARE OUTPUT	44.19.01.2		B<A		B>A		B = A		B<A		B = A		B<A		B>A		B>A	B = A		
	ADDER-DOUBLER COMPARE OUTPUT	44.19.11.2		B=2A		B>2A		B<2A		B=2A		B>2A		B<2A		B>2A		B>2A	B<2A		
	COMPARE GATE	44.13.01.2																			
A COMPARE	B < A LAT	44.19.21.2																			
A COMPARE	SET: (B < A)																				
A COMPARE	RESET: (B > 2A)																				
A COMPARE	SUBT A LAT	44.19.21.2																			
A COMPARE	SET: (B = A) ($\overline{B < A\ LAT}$)																				
A COMPARE	SET: (B > A)																				
A COMPARE	RESET: (B < A)																				
2A COMPARE	B < 2A LAT	44.19.31.2																			
2A COMPARE	SET: (B < 2A)																				
2A COMPARE	RESET: (B > 2A)																				
2A COMPARE	SUBT 2A LAT	44.19.31.2																			
2A COMPARE	SET: (B = 2A) ($\overline{B < 2A\ LAT}$)																				
2A COMPARE	SET: (B > 2A)																				
2A COMPARE	RESET: (B < 2A)																				

than the low-order A-field positions. Therefore, if (B=2A) ($\overline{B<2A\ Lat}$), the SUBTRACT-2A latch is set.

On the second B-cycle after the A-field WM of a divide-compare operation, if the B-register digit is greater than twice the A-register digit, then the whole B-field is greater than twice the whole A-field, regardless of the low-order field position. Therefore, the SUBTRACT-2A latch is set.

It is possible to set both the SUBTRACT-A latch and SUBTRACT-2A latch; however, the switching determines whether to initiate a subtract-A or subtract-2A operation. If the SUBTRACT-2A latch is ON during the third B-cycle after the A-field WM, a subtract-2A operation is initiated regardless of the status of the SUBTRACT-A latch. To initiate a subtract-A operation requires that the SUBTRACT-A latch be ON and the SUBTRACT-2A latch be OFF the third B-cycle after the A-field WM of a divide-compare operation.

Cycle Controls

A-Auxiliary Cycle Control

A-auxiliary cycle control consists basically of a DELTA A-AUX-CYCLE latch and an A-AUX-CYCLE latch. The purpose of the A-auxiliary cycle control latches is to provide additional control on the cycles in which storage is addressed from the A-AUX-STAR.

E-phase still consists of A- and B-cycles. However, the A-cycles on which the A-AUX-CYCLE latch is activated are referred to as A-auxiliary cycles. As in basic cycle control, the DELTA A-AUX-CYCLE latch is activated prior to the A-AUX-CYCLE latch. ILD 27 shows the switching to set DELTA A-AUX-CYCLE latch at 045-060 time, and set the A-AUX-CYCLE latch at 000-030 time.

The DELTA A-AUX-CYCLE latch is set at 045 time and the DELTA A-CYCLE latch is set at 075 time of the same cycle. The DELTA A-AUX-CYCLE output is used to prevent the A-STAR from reading-out into the STAR on A-auxiliary cycles as shown in ILD 27.

During multiply operations, an A-auxiliary cycle is necessary to address the units position of the A-field from the A-AUX-STAR at the beginning of an add-A or add-2A operation. On the B-cycle in which the MULTIPLIER latch is ON, the B-register digit is analyzed to determine if it is a significant multiplier digit equal to or greater than two, or equal to one. If the multiplier digit is significant (not zero or blank), the DELTA A-AUX-CYCLE latch is set at 045 time to allow the units position of the A-field (multiplicand) to be addressed on the following A-auxiliary cycle, and an add-A or add-2A operation initiated.

As shown in ILD 27, this is done by switching SIGNIFICANT DIGIT, PROCESS, NOT B-AUX-CYCLE, and MULTIPLIER latch to set the DELTA A-AUX-CYCLE latch. The NOT B-AUX-CYCLE is switched to distinguish between a multiplier digit cycle and a product shift cycle. In event of a zero multiplier, the MULTIPLER latch remains on and a B-auxiliary cycle is taken to shift the product field. If a significant digit is addressed on the B-auxiliary cycle, the DELTA A-AUX-CYCLE latch is not set. The add-A or add-2A operation is not initiated unless a need is indicated by a significant multiplier digit. DELTA PROCESS, TIME 000-030, and the DELTA A-AUX-CYCLE latch are switched to set the A-AUX-CYCLE latch if the various STAR keys, STORAGE SCAN and PRINT-OUT are not in use.

During divide operations, an A-auxiliary cycle is used to address the units position of the A-field from the A-AUX-STAR at the beginning of a simultaneous subtract and divide-compare operation. If a divide-compare operation indicates a subtract operation is possible, the quotient digit is increased as determined by the A- and 2A-COMPARE latches, and the DELTA A-AUX-CYCLE latch is set to initiate a simultaneous subtract and divide-compare operation. The switching to set the DELTA A-AUX-CYCLE latch at this time is QUOTIENT TRIGGER, NOT B-LESS-THAN-A latch, and TIME 045-060, and is shown in ILD 27.

If the divide-compare operation does not indicate a possible subtract operation, the quotient digit is increased by 0 and a B-auxiliary cycle is initiated to shift the dividend field. Because another divide-compare operation is necessary to compare the divisor with the next partial dividend, the DELTA A-AUX-CYCLE latch must be set to initiate the operation. ILD 27 shows that the B-LESS-THAN-A latch is switched with the B-AUX-CYCLE output and TIME 045-060 to set the DELTA A-AUX-CYCLE latch.

With the DELTA A-AUX-CYCLE latch activated, the A-AUX-CYCLE latch is set at the following time 000-030 for a full cycle, if the various STAR keys and STORAGE SCAN and PRINT-OUT are not in use.

The DELTA A-AUX-CYCLE latch is normally reset at 060 time of the following A-auxiliary cycle, but may also be reset by pressing the start reset key or by a gated load key, or by a load tape operation.

The A-AUX-CYCLE latch may also be set by the A-AUX-STAR key. Note also that the A-AUX light is ON whenever the A-AUX-CYCLE latch is activated.

B-Auxiliary Cycle Control

B-auxiliary cycle control consists basically of a DELTA B-AUX-CYCLE latch and a B-AUX-CYCLE latch. The purpose of the B-auxiliary cycle control is to provide control in addition to the basic B-cycle control so that storage is addressed from the B-AUX-STAR.

E-phase still consists of A- and B-cycles. However, the B-cycles in which the B-AUX-CYCLE latch is activated are referred to as B-auxiliary cycles. As in A-auxiliary-cycle control, the DELTA B-AUX-CYCLE latch is set on the cycle prior to the B-AUX-CYCLE latch. The set time for the DELTA B-AUX-CYCLE latch is 060-090 as shown in ILD 28, while the B-AUX-CYCLE latch is set at 000-030 time.

The DELTA B-AUX-CYCLE latch is set at 060 time, and the DELTA B-CYCLE latch is set at 075 time of the same cycle. The DELTA B-AUX-CYCLE latch output is used to block the normal B-STAR gate-out as shown in ILD 28.

During multiply or divide operations, a B-auxiliary cycle always follows an A-auxiliary cycle. Therefore, on the A-auxiliary cycle, the DELTA B-AUX-CYCLE latch is set as shown in ILD 28. The switching used is A-AUX-CYCLE and TIME 060-090.

Whenever the DELTA B-AUX-CYCLE latch is set, the B-AUX-CYCLE latch is set at Time 000-030 of the following cycle, if the various STAR keys, STORAGE SCAN, and PRINT-OUT are not in use.

During multiply operations, a B-auxiliary cycle is initiated to shift the product field if the multiplier digit addresed is zero. The switching shown in ILD 28 to initiate the Delta B-auxiliary cycle is TIME 060-090, B-REG 0 OR BLANK, NOT B-AUX-CYCLE, MULTIPLIER latch, and GATED WM AND B-CYCLE and NOT SET PRODUCT SIGN latch.

NOT B-AUX-CYCLE switching prevents initiating another B-auxiliary cycle if a zero is addressed in the product field during the first B-auxiliary cycle. The product shift is initiated only by a zero in the multiplier field.

GATED WM AND B-CYCLE and NOT SET PRODUCT SIGN latch switching prevents initiating another B-auxiliary cycle at the same time an I-E Change is initiated. All conditions mentioned are met on the third B-cycle after the A-field WM of a normal multiply operation. The inverted output blocks the setting of the DELTA B-AUX-CYCLE latch.

If the multiply operation is a special case of a single digit zero multiplier, then the *set product sign* is ON during the third B-cycle after the A-field WM when the zero multiplier with a WM is addressed. This allows the DELTA B-AUX-CYCLE latch to be set, and a B-auxiliary cycle taken in which to set a standard sign in storage for the zero product field.

During divide operations, a B-auxiliary cycle is iniated to shift the dividend field (and quotient field) if the divide-compare operation indicates that the partial dividend is less than the divisor (B<A).

The switching in ILD 28 to set the DELTA B-AUX-CYCLE latch is B-LESS-THAN-A latch, QUOTIENT TRIGGER, PROCESS, TIME 060-090, and NO END DIVIDE latch. If the switching conditions are met with the END DIVIDE latch set, an I-E Change is initiated instead, which ends the divide operation.

The DELTA B-AUX-CYCLE latch is normally reset at 060 time of the following B-auxiliary cycle, but also may be reset by pressing the start reset key, or activating gated load key or load tape.

The B-AUX-CYCLE latch may also be set by the B-AUX-STAR key. Note that the B-AUX light is ON when the B-AUX-CYCLE latch is activated.

Summary of Multiply-Divide Switching and Controls

This section consists of a written summary of the switching and controls used for multiply and divide operations. The expressions are written in Boolean Algebra and are handled in a fashion similar to algebra. The plus signs are *or* conditions and the parentheses are *and* conditions. Each expression in the large bracket is *anded* with the expression outside of the bracket. The absence of a condition is indicated by *not* or an overhead bar, as *Not B-cycle* or $\overline{\text{B-cycle}}$.

For example, item #1 is:

AUX-STAR SET LATCH
ON: (I-Cycle) (Gated WM) (Mult Opr + Div Ops)
(Time 090-000)
OFF: (B-Cycle) (Time 090-000) + (Start Reset)

The expression indicates that the AUX-STAR SET latch is turned ON during a multiply operation by switching I-cycle and gated WM and Mult Opr and Time 090-000. During a divide operation, I-cycle and gated WM and Div Opr and Time 090-000 are switched to turn on the AUX-STAR SET latch. To reset the AUX-STAR SET latch requires switching B-cycle and time 090-000; press the start reset key to activate START RESET.

To illustrate the use of brackets, the switching to turn ON the DELTA A-AUX-CYCLE latch is used in the following example.

DELTA A-AUX-CYCLE LATCH
ON: [$\overline{(B < A\text{Lat})}$ (Quot Tr) (Time 045-060) + (B-Aux-Cycle)
(B< A Lat) (Time 045-060) + (Mult Lat)
(Significant Digit) (B-Aux-Cycle) (Time 045-060)
(Process)] (Process)

Three different ways to turn ON the DELTA A-AUX-CYCLE latch are included in the expression. NOT B-LESS-THAN-A latch and QUOTIENT TRIGGER and TIME 045-060 and PROCESS are switched in one method to set the latch. A second method to set the latch is to switch B-AUXILIARY-CYCLE and B-LESS-THAN-A latch and TIME 045-060 and PROCESS. The third method to set the latch is to switch MULTIPLIER latch and SIGNIFICANT DIGIT and NOT B-AUXILIARY-CYCLE and TIME 045-060 and

PROCESS and PROCESS. Note that PROCESS is switched with each set at least once. On the third set, PROCESS is actually switched twice. This may be verified on the ALD's by referring to Process Circuit 1 (44.10.01.2), and Process (31.02.41.2).

To aid in further understanding and handling these expressions, consider $\overline{(\text{Mult Opr} + \text{Div Opr})}$. Actual switching might appear as in Figure 45A.

An example similar to this is found on ALD 44.11.01.2.

The expressions $\overline{(\text{Mult Opr} + \text{Div Opr})}$ may also be rewritten as $(\overline{\text{Mult Opr}})\ (\overline{\text{Div Opr}})$. The end result is the same; however, the actual switching would now appear as in Figure 45B.

Because the functional result is the same, it may be easier to understand involved functions by reducing them to simpler form. Therefore, when several functions are *or'd* together under a common bar, the function may be reduced by breaking the bar and changing each *or* to an *and*. For example, $\overline{(A + B + C + D)}$ becomes $(\bar{A})\ (\bar{B})\ (\bar{C})\ (\bar{D})$. Likewise, when several functions are *anded* together under a common bar, the function may be reduced by breaking the bar and changing each *and* to an *or*. For example, $\overline{(A)\ (B)}\ \overline{(C)\ (D)}$ becomes $\bar{A} + \bar{B} + \bar{C} + \bar{D}$. Likewise, $\overline{(A + B)}\ \overline{(C + D)}$ becomes $(\bar{A})\ (\bar{B}) + (\bar{C})\ (\bar{D})$.

The Boolean expressions contained in this section are generally in reduced form, but the techniques of handling written circuit expressions are also valuable in machine diagnostics.

To aid in locating the desired expression, the various switching and controls which follow are grouped by latches, triggers, signals, and gates. Each item is numbered and is in approximate alphabetical order in each group as a further aid in locating the desired switching.

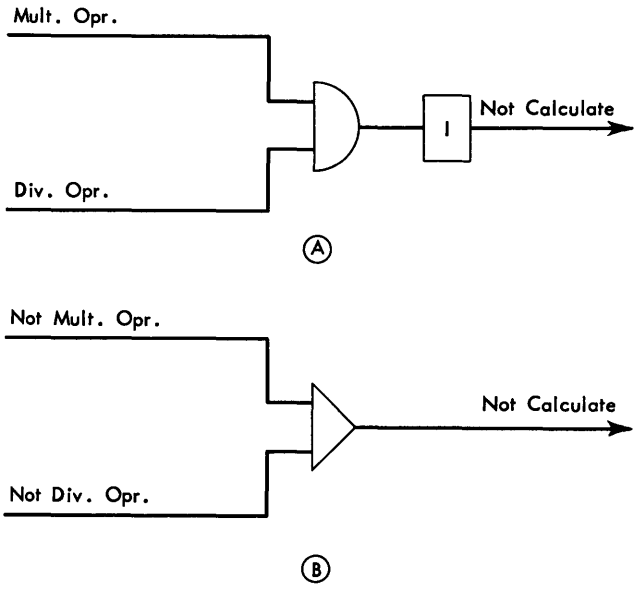


Figure 45. Examples of Logic

M-D Switching and Controls

LATCHES

1. AUX-STAR Set Latch
2. Delta A-Aux Cycle Latch
3. A-Aux Cycle Latch
4. Delta B-Aux Cycle Latch
5. B-Aux-Cycle Latch
6. A-Cycle Eliminate Latch (M-D)
7. A Latch
8. 2A Latch
9. B A Latch
10. B 2A Latch
11. Clear B-field Latch
12. End Divide Latch
13. Multiplier Latch
14. Reverse Scan Latch (M-D)
15. Set Product Sign Latch
16. Subtract A Latch
17. Subtract 2A Latch
18. X-Position Latch
19. Single Zero Latch

TRIGGERS

20. Quotient Trigger
21. Sign Trigger

SIGNALS

22. Block Zone Adder (Calculate)
23. Force Adder Carry
24. Force 8 2
25. I-E Change
26. B-register Digit Generation
27. Reset A-register
28. Reset Compare Latches
29. Set A-bit in Storage
30. Set B-bit in Storage
31. Set CZ in Storage
32. Set A-register C-bit
33. Set A-reg Translator Output to 0
34. Set A-reg Translator Output to 1
35. Set A-reg Translator Output to 2
36. Set Arith Check Latch
37. Set Overflow Latch

GATES

38. A-AUX-STAR Set (Units Gate)
39. A-AUX-STAR Set (T & H Gate)
40. A-AUX-STAR Gate Out
41. A-STAR Gate Out
42. B-AUX-STAR Set (Units Gate)
43. B-AUX STAR Set (T & H Gate)
44. B-AUX-STAR Gate Out
45. B-STAR Gate In (Units Gate)

46. B-STAR Gate In (T & H Gate)
47. B-STAR Gate Out
48. A Gate
49. 2A Gate
50. Arith Digit Gate
51. Arith Zone Gate
52. B-reg Zone Inh Gate
53. B-register Transfer Gate
54. Carry Trigger Gate
55. Complement Gate
56. Compare Gate
57. Set Sign Gate
58. Transfer STAR Calc Gate

1. AUX-STAR SET LATCH — ILD 30 A3

ON: (I-cycle) (Gated WM) (Mult Opr + Div Opr) (Time 090-000)

OFF: (B-cycle) (Time 090-000) + (Start Reset)

2. DELTA A-AUX-CYCLE LATCH — ILD 32 A3

ON: [$\overline{(B < A\ Lat)}$ (Quote Tr) (Time 045-060) ($\overline{I\text{-}E\ Change}$) + (B-Aux-Cycle) (B < A Lat) (Time 045-060) + (Mult Lat) (Significant Digit) (B-Aux-Cycle) (Time 045-060) (Process)] (Process)

OFF: (A-Aux-Cycle Lat) (Process) (Time 060 090) + (Start Reset) + (Load Tape) + (Gated Load Key)

3. A-AUX-CYCLE LATCH — ILD 32 B3

ON: (Δ A-Aux-Cycle Lat) (Δ Process) (Time 000-030) + (A-AUX-STAR Key)

OFF: (Δ Process) (Time 000-015) + (B-AUX-STAR Key) + (I-STAR Key) + (A-STAR Key) + (B-STAR Key) + (Storage Scan) + (Storage Print-Out)

4. DELTA B-AUX-CYCLE LATCH — ILD 32 A5

ON: (Mult Lat) (B-reg 0 or Blank) ($\overline{B\text{-}Aux\text{-}Cycle}$) (Process) (Time 060-090) ($\overline{Gated\ WM}$) (B-cycle) ($\overline{Not\ Set\ Prod\ Sign\ Lat}$) + (A-Aux-Cycle) (Process) (Time 060-090) + ($\overline{B < A\ Lat}$) (Quot Tr) (Process) (Time 060-090) ($\overline{End\ Div\ Lat}$)

OFF: (B-Aux-Cycle Lat) (Process) (Time 060-090) + (Start Reset) + (Load Tape) + (Gated Load Key)

5. B-AUX-CYCLE LATCH — ILD 32 B5

ON: (Δ B-Aux-Cycle Lat) (Δ Process) (Time 000-030) + (B-AUX-STAR Key)

OFF: (Δ Process) (Time 000-015) + (A-AUX-STAR Key) + (I-STAR Key) + (A-STAR Key) + (B-STAR Key) + (Storage Scan) + (Storage Print-Out)

6. A-CYCLE ELIMINATE LATCH (M-D) — ILD 30 A4

ON: (A-reg WM) (B-cycle) (Time 060-090) (Mult Opr Div Opr) ($\overline{Aux\text{-}STAR\ Set\ Lat}$)

OFF: ($\overline{B < A\ Lat}$) (Quot Tr) (Time 045-060) + (I-cycle) + (B < A Lat) (Time 045-060) ($\overline{B\text{-}Aux\text{-}Cycle}$) + (Mult Lat) (Significant Digit) ($\overline{B\text{-}Aux\text{-}Cycle}$) (Time 045-060) (Process)

7. A-LATCH — ILD 31 C2

ON: (Subt A Lat) ($\overline{Subt\ 2A\ Lat}$) (Δ A-Aux-Cycle) + (Mult Lat) (B-reg 1) ($\overline{B\text{-}Aux\text{-}Cycle}$) (Process)

OFF: (X-Position Lat) (Process) (Time 000-015) + (Start Reset)

8. 2A-LATCH — ILD 31 D2

ON: (Subt 2A Lat) (Δ A-Aux-Cycle) + (Mult Lat) (B-Aux-Cycle) (B-reg 2 or Greater) (Process)

OFF: (X-Position Lat) (Process) (Time 000-015) + (Start Reset)

9. B-LESS-THAN-A LATCH — ILD 29 B6

ON: ($\overline{B = A}$) ($\overline{B > A}$) (Compare Gate) (Time 075-090)

OFF: (B > A) (Compare Gate) (Time 075-090) + (I-Cycle) + (A-Aux-Cycle)

10. B-LESS-THAN-2A LATCH — ILD 29 D6

ON: (B < 2A) (Compare Gate) (Time 075-090)

OFF: ($\overline{B < 2A}$) ($\overline{B = 2A}$) (Compare Gate) (Time 075-090) + (I-cycle) + (A-Aux-Cycle)

11. CLEAR B-FIELD LATCH — ILD 30 C5

ON: (Mult Opr) (I-cycle) (Gated WM)

OFF: (Set Prod Sign Lat) (Process) (Time 000-030) + (Start Reset)

12. END DIVIDE LATCH — ILD 31 A2

ON: (B-cycle) (B reg B) (Div Opr) (Time 060-090)

OFF: (I-cycle)

13. MULTIPLIER LATCH — ILD 30 C3

ON: (X-Position Lat) (Mult Opr) (Process) (Time 015-030)

OFF: ($\overline{A\text{-}Cycle\ Elim\ Lat}$) (Process) (Time 000-030)

14. REVERSE SCAN LATCH (M-D) — ILD 31 B2

ON: (B < A Lat) (Quot Tr) (Time 105-000)

OFF: (A-Cycle Elim Lat) (Time 090-000)

15. SET PRODUCT SIGN LATCH — ILD 30 D5

ON: (Clear B-field Lat) (Mult Lat) (Time 060-090)

OFF: ($\overline{Δ\ B\text{-}Aux\text{-}Cycle}$) ($\overline{Δ\ A\text{-}Aux\text{-}Cycle}$) (Process) (Time 000-030)

16. SUBTRACT-A LATCH — ILD 29 C6

ON: (B > A) (Compare Gate) (Time 075-090) + (B = A) ($\overline{B < A\ Lat}$) (Compare Gate) (Time 075-090)
OFF: ($\overline{B > A}$) ($\overline{B = A}$) (Compare Gate) (Time 075-090) + (I-cycle) + (A-Aux-Cycle)

17. SUBTRACT-2A LATCH — ILD 29 D6

ON: (B = 2A) ($\overline{B < 2A\ Lat}$) (Compare Gate) (Time 075-090) + (B = 2A) ($\overline{B < 2A}$) (Compare Gate) (Time 075-090)
OFF: (B < 2A) (Compare Gate) (Time 075-090) + (I-cycle) + (A-Aux-Cycle)

18. X-POSITION LATCH — ILD 30 C3

ON: (A-cycle Elim Lat) (Time 030-045) (Process)
OFF: ($\overline{A\text{-cycle Elim Lat}}$) (Time 000-030) (Process)

19. SINGLE ZERO LATCH — ILD 30 B5

ON: (B reg 0 or Blank) (Mult Lat) (Clear B-field Lat) (Gated WM)
OFF: (I-cycle)

20. QUOTIENT TRIGGER — ILD 31 A2

ON GATE: ($\overline{Quot\ Tr}$) (X-Position Lat) (A-cycle Elim Lat)
BINARY SET: (Div Opr) (Process) (Time 000-030)
OFF GATE: (Quot Tr)
RESET: (Start Reset)

21. SIGN TRIGGER — ILD 30 D3

ON GATE: [(A-cycle) (AUX-STAR Set Lat) + (B-cycle) (Div Opr) + (Clear B-field Lat) (B-cycle) (Mult Lat)] ($\overline{Sign\ Tr}$)
BINARY SET: (Time 030-060) (B-reg B Not A) ($\overline{End\ Div\ Lat}$)
OFF GATE: [(Div Opr) + (Clear B-field Lat) (Mult Lat)] (B-cycle) (Sign Tr)
RESET: (I-cycle)

22. BLOCK ZONE ADDER (CALCULATE) — ILD 25 B1

(Mult Opr + Div Opr)

23. FORCE ADDER CARRY — ILD 32 B1

(Div Opr) (A-Aux-Cycle) (Time 090-000)

24. FORCE 8 2 — ILD 30 C4

(Clear B-field Lat) ($\overline{Mult\ Lat}$) (Not B 1 Opr and B-cycle) + (Mult Lat) (B-reg > 2) (Not B 1 Opr and B-cycle) (B-Aux-Cy)

25. I-E CHANGE — ILD 31 A3

(End Div Lat) (B < A Lat) (Quot Tr) + (Single Zero Lat) (B-Aux-Cy) (Gated WM) (Mult Lat) (B-reg 0 or Blank) ($\overline{Clear\ B\text{-}field\ Lat}$) + (Subt 2A Lat) (B-reg 8 Not 4) (B-reg Not 2 Not 1) (Quot Tr)

26. B-REGISTER DIGIT GENERATION — ILD 30 B1

See section under *B-Register Digit Generator*, Figure 42.

27. RESET A-REGISTER — ILD 30 A5

(A-Cycle Elim Lat) (Time 000-015)

28. RESET COMPARE LATCHES — ILD 29 A6

(A-Aux-Cycle) + (I-Cycle)

29. SET A-BIT IN STORAGE — ILD 31 C4

(Set Sign Gate) ($\overline{Sign\ Tr}$) (Not B1 Opr and B-cycle)

30. SET B-BIT IN STORAGE — ILD 31 D4

(Set Sign Gate) (Not B1 Opr and B-cycle)

31. SET C_z IN STORAGE — ILD 31 C4

(Set Sign Gate) ($\overline{Sign\ Tr}$) (Not B1 Opr and B-cycle)

32. SET A-REGISTER C-BIT — ILD 30 A4

(A-Cycle Elim Lat) (Time 030-045)

33. SET A-REG TRANSLATOR OUTPUT TO 0 — ILD 29 C3

($\overline{A\text{-}Gate}$) ($\overline{Not\ 2A\ Gate}$) ($\overline{Set\ to\ 1}$) ($\overline{Set\ to\ 2}$)

34. SET A-REG TRANSLATOR OUTPUT TO 1 — ILD 31 C5

(Quot Tr) (Subt A Lat) ($\overline{Subt\ 2A\ Lat}$) + (B reg > 2) (Mult Lat) ($\overline{B\text{-}Aux\text{-}Cycle}$) (Process)

35. SET A-REG TRANSLATOR OUTPUT TO 2 — ILD 31 C5

(Quot Tr) (Subt 2A Lat)

36. SET ARITH CHECK LATCH — ILD 31 C6

(Arith Digit Gate Calc) (Arith Check) (Process) (Time 075-090)

37. SET OVERFLOW LATCH — ILD 31 B1

(Quot Tr) (B-reg 8 Not 4) (Subt 2A Lat) (B-reg Not 2 Not 1) (Process)

38. A-AUX-STAR SET (UNITS GATE) — ILD 32 B1

(AUX-STAR Set Lat) (Δ A-cycle)

39. A-AUX-STAR SET (T & H GATE) — ILD 32 B1

(AUX-STAR Set Lat) (A-cycle)

40. A-AUX-STAR GATE-OUT — ILD 27 C2

(Δ A-Aux-Cy) (Time 090-000) + (A-AUX-STAR Key)

41. A-STAR GATE-OUT CALC — ILD 27 D2

$\overline{\text{(Delta A-Aux-Cycle)}}$

42. B-AUX-STAR SET (UNITS GATE) — ILD 32 A1

(Δ B-Aux-Cycle) (B<A Lat) + (AUX-STAR Set Lat)
(Δ B-cycle) + (Δ B-Aux-Cycle) (Mult Lat)

43. B-AUX-STAR SET (T & H GATE) — ILD 32 A1

(B-Aux-Cycle) (X-Position Lat) (B<A Lat) +
(AUX-STAR Set Lat) (B-cycle) + (B-Aux-Cycle) (Mult Lat)

44. B-AUX-STAR GATE-OUT — ILD 32 C3

(Δ B-Aux-Cycle) (Time 090-000) + (B-Aux-Star Key 1)

45. B-STAR GATE-IN (UNITS GATE) — ILD 32 B1

$\overline{\text{(Mult Lat) (Δ B-Aux-Cycle)}}$

46. B-STAR GATE-IN (T & H GATE) — ILD 32 B1

$\overline{\text{(Mult Lat) (B-Aux-Cycle)}}$

47. B-STAR GATE-OUT — ILD 32 D3

$\overline{\text{(Δ B-Aux-Cycle)}}$

48. A-GATE — ILD 31 C2

$\overline{\text{(X-Position Lat)}}$ (A Lat) + $\overline{\text{(Mult Opr + Div Opr)}}$

49. 2A-GATE — ILD 31 D2

(Mult Lat) (Quot Tr) (2A Lat)

50. ARITH DIGIT GATE — ILD 31 C6

[(Quot Tr) + (A Lat + 2A Lat) (Mult Lat) + (Mult Lat)
(B-reg >2) (B-Aux-Cycle)] (Not B1 Opr and B-cycle)

51. ARITH ZONE GATE — ILD 31 B6

[(Set Sign Gate) + (Div Opr) (Complement from Calc) +
(Mult Opr) (Not Mult Lat) (Not Set Prod Sign Lat)
(A-Lat + 2A Lat)] (Not B1 Opr and B-cycle)

52. B-REGISTER ZONE INHIBIT GATE — ILD 31 B6

[(Div Opr) (Complement from Calc) + (Mult Opr)
(Not Mult Lat) (Not Set Prod Sign Lat) (A Lat + 2A Lat)]
(Not B1 Opr and B-cycle)

53. B-REGISTER TRANSFER GATE — ILD 31 B6

$\overline{\text{(Arith Digit Gate)}}$ (Div Opr) (Not B1 Opr and B-Cy) +
(X-Pos Lat) (B-Aux-Cy) (Mult Opr)
(Not B1 Opr and B-cycle)

54. CARRY TRIGGER GATE-ON CALC — ILD 30 B3

(Mult Opr + Div Opr) (Adder Carry) (Arith Digit Gate)
(X-Position Lat) (Time 060-090)

55. COMPLEMENT FROM CALC (GATE) — ILD 31 B5

(A-Lat + 2A Lat) (B-cycle) (Div Opr) $\overline{\text{(Quote Tr)}}$ +
(Mult Lat) (B-reg >2) (B-Aux-Cycle) (Process) +
(Compl to Calc)

56. COMPARE GATE — ILD 31 B4

(B-cycle) (Div Opr) $\overline{\text{[(AUX-STAR Set Lat) + (Quot Tr) +}}$
$\overline{\text{(B-Aux-Cycle) (B<A Lat) (X-Positions Lat)]}}$

57. SET SIGN GATE — ILD 31 C3

(End Div Lat) (Quote Tr) + (Set Prod Sign Lat)
(B-Aux-Cycle)

58. TRANSFER STAR CALC GATE — ILD 30 A3

(AUX-STAR Set Lat)

General Information

The tape configuration for the IBM 1401 is actually a selective feature rather than an option. The simple card-oriented configuration has been considered the basic 1401 system. Tape has been considered as an add-on rather than an alternate. The IBM 1401 CE Instruction Manual (Form 225-6540) covers only the basic card system. This discussion of the tape system is intended to supplement the basic manual. The basic data flow, components, and programming concept remain unchanged.

The tape feature can be used in various combinations. Tape-drives along with a 1401 processing unit can do various tape-to-tape processes without other units. Adding card units, a printer, serial I-O attachments, or disk storage allows transfers between units with or without processing during the transfer. In this function, the 1401 may serve either by itself or as an off-line printer or data converter for larger data processing systems.

Machine Language

The 1401 system performs all of its internal functions using the binary coded decimal (BCD) notation. The C-bit (Check) is used to produce odd-redundancy count. An eighth or word-mark (WM) bit is carried in the high-order character of each word to define its limits. The word-marked character has its check bit alternated to retain odd redundancy with the extra bit.

Recording on tape for commercial systems is binary coded decimal notation using even redundancy. Scientific systems, using straight binary notation, break the larger word into six bit sections with the check added to produce odd redundancy. The 1401 normally reads and writes tape records in BCD notation with even redundancy. Provision is made to allow reading and writing binary notation with odd redundancy for transfer or conversion operations.

The word-mark bit is not recorded on tape. Its removal, and the change from odd-bit redundancy to even during a write operation, is accomplished in the tape I/O control circuits. Likewise, the change from even-bit redundancy on tape to odd-bit is done in the same area. The retention of B-field word-marks during a read operation is controlled in the processing unit.

A redundancy change is gated by a redundancy latch set from the instruction detail. The redundancy latch also controls the gating of even and odd redundancy validity checks in TAU. The check character is always checked for even redundancy. When writing odd redundancy types, an even number of characters must always be written.

Data Flow

The data flow of the processing area remains unchanged from the card system. The tape area receives data to write from the B-register as it reads from core storage. Data read from tape enters the A-register for entry into core storage (Figure 46).

In writing, an addressed character reads out of core storage to the B-register, and is regenerated. It is gated through the tape I/O control to the read-write register in the Tape Adapter Unit. The check bit is adjusted in transfer for even-redundancy and possible word-mark removal. The TAU drives the tape writing circuits to record on tape. Validity is checked in the TAU at the read-write register, at echo-test from the write circuits, and at the skew registers from a read-after-write.

In a read operation the tape is read into the read registers and then transferred to the read-write register. It is gated through the tape I-O control to the A-register. The check bit is adjusted during the transfer. The character reads into core storage to replace the character read out of the B-field. If a word-mark is to be added, it enters the inhibit drive switching and the check bit is adjusted.

Tape Instructions

Four operation codes are used to control the tape feature. Two are used to read and write tape. They are modifications of the basic Move (M) and Load (L) instructions. A unit-control (U) instruction is used to perform tape operations not requiring data movement. These include rewind, backspace, erase, and write tape mark. Two additional d-character codes are provided for the *Branch* (B) instruction to test tape indicators. Except for the branch operation, the complete instruction requires the characters in the Op Register, the A-Register, and the A-Star.

M(% UX)(BBB)d — MOVE MAGNETIC TAPE

The M (Move) operation is used to read and write tape without reference to word-marks. If word separators are present, or desired, on tape, the L code must be used. The normal (AAA) address is used to designate a tape operation (% UX). The % sign designates

tape I-O control and the U indicates a normal even-redundancy tape operation. The U is replaced with other letter characters for optional tape and I-O features. The units position X is the numerical designation assigned to the required tape unit. The d-character of R or W designates a read or write operation respectively. During operation the A-address register remains unchanged. The d-character is lost with the first use of the A-register, but the operation is retained in a latch.

The write operation starts by transferring the data at the (BBB) address from core storage to tape. The address is modified by +1 and the transfer continued. The operation stops when a group-mark with word-mark is sensed as the B-field character. The group-mark is not written on tape. Instead, it serves to terminate the record and stop the tape unit, resulting in a record gap.

The read operation starts by transferring the first tape character to the (BBB) address in core storage. The address is modified by +1 and the transfer continued. The operation stops when either a group-mark with word-mark is encountered in the B-field or a tape record gap is detected. If the end of the storage area is reached first, transmission stops but the tape unit continues to the end of the record. If the end of record comes first, a group-mark is forced into the following position of the B-field in core storage.

L (% UX)(BBB)d — LOAD MAGNETIC TAPE

The L (Load) operation is used to read and write tape with word-marks. A word-mark writes as a word separator (A-841) before the associated character on tape. A word separator on tape reads as a word-mark with the following character. A tape record with word separators must be read with the load code.

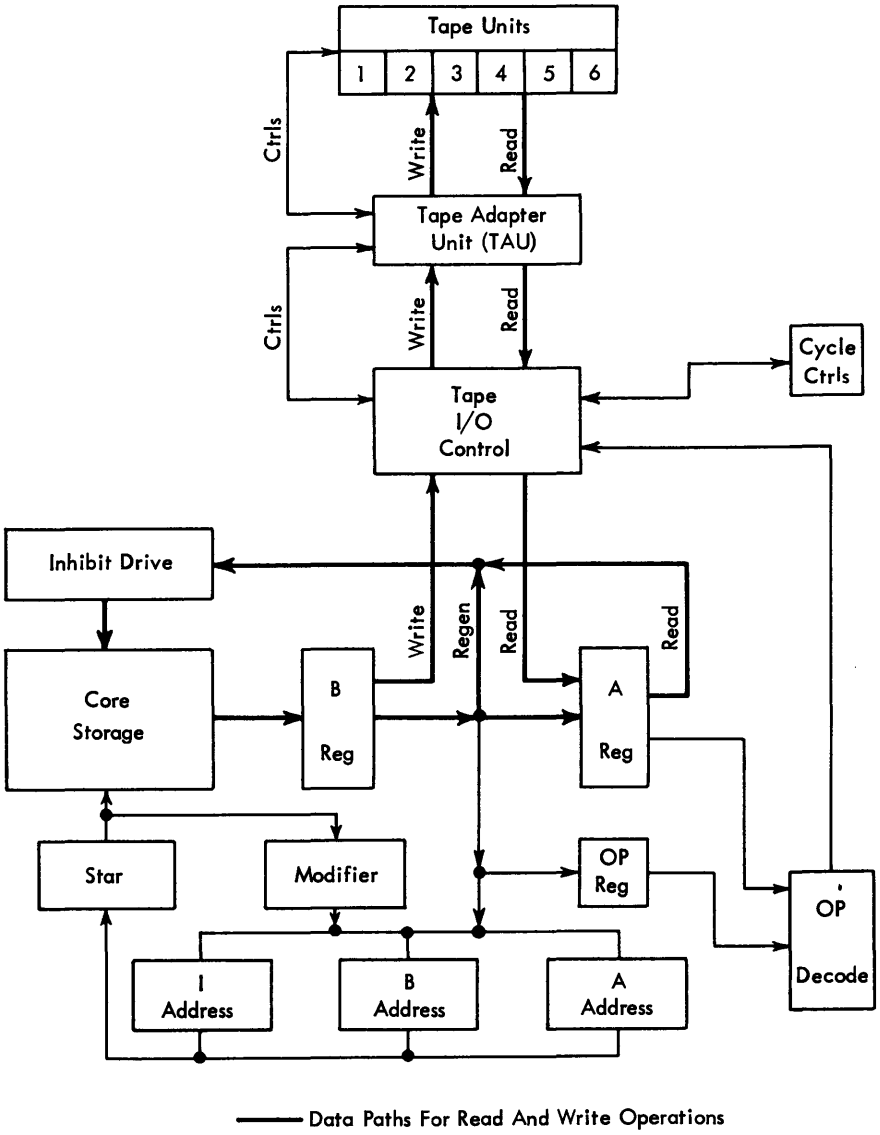


Figure 46. Data Flow Schematic for IBM 1401 Tape System

The breakdown of the (AAA) address and the d-character are the same as for the M-operation. The data flow and control differs only in the handling of the word-mark. During a write operation the two tape characters are written from one storage location. On a read operation the word separator is first stored and then replaced by the next character with a word-mark. The start and end of record controls are identical to the M-operation.

U (% UX) d — UNIT CONTROL

The U (Unit Control) operation code is used to control tape operations other than data transmission. The (AAA) address is used to designate the tape unit to be controlled. The same (% UX) used in the Move and Load instructions determines the selection. The U-character specifies tape control, while other letters in this position are used for optional I-O features. No (BBB) address is used, since data is not being transferred. The d-character designates the operation to be performed as follows:

d-character	*Operation*
A	Diagnostic read instruction causes a tape record to be read without transmission of data. Used for the inter-record gap (IRG) test.
B	Backspace tape one record. Tape is read in the reverse direction, similar to a read operation with stop on the inter-record gap. No data is transmitted. The operation is controlled by the TAU.
E	Erase or Blank and Skip 3½" of tape before the next write operation. A latch is set in the TAU to remember the skip but not the tape unit. The proper write instruction must follow immediately for proper operation.
M	Write Tape Mark (end-of-file for the tape reel). The operation is similar to writing a one character record. The tape mark character (8421) is forced in TAU.
R	Rewind tape operation causes the tape to rewind to the starting point. The tape unit is left in ready status with the tape loaded and ready for use again.
U	Rewind tape and unload operation performs the rewind to the starting point, but leaves the tape unit in the unloaded status. The tape unit cannot be used again until it is manually loaded.

B (III) d — TEST AND BRANCH

The test and branch operation is identical to the basic system tests. Two new d-characters are added to test the tape conditions. These tests should be performed immediately after each read and write operation to be sure an error condition did not develop.

d-character	*Operation*
K	Test end-of-file indicator and branch to the (III) address if ON. The tape indicator is set ON by sensing a tape mark on read or sensing the end of reel reflective tape on write. The indicator is reset by the next tape operation.
L	Test Tape Error indicator and branch to the (III) address if ON. The indicator is set from various transmission errors detected in the TAU. The indicator is reset after the test.

Tape Units and Speeds

The tape configuration for the IBM 1401 can accommodate up to six magnetic tape units (IBM 729-II), IBM 729-IV or IBM 7330). All units on the system must be of the same model. All models may be operated in high-character density or low-character density. Figure 47 shows the relative speeds of the various combi-

	729 - II		729 - IV		7330	
	Low Density	High Density	Low Density	High Density	Low Density	High Density
Density (Characters/Inch)	200	556	200	556	200	556
Character Rate (Characters/Second)	15,000	41,667	22,500	62,500	7,200	20,000
Tape Speed (Inches/Second)	75		112.5		36	
Inter - Record Gap (Inches)	3/4		3/4		3/4	
Start/Stop Time, Read/Write Operation (Millisecond)	10.8		7.3		20.8	

Figure 47. Speeds and Character Rates for IBM Tape Drives

nations. The high-character density offers greater speed and provides a significant storage advantage from the reduction in the number of reels. The lower density is used when IBM 727 tapes are to be read or prepared. A tape must be read with the same density control as it was written.

The tape units are connected to the system through a Tape Adapter Unit (TAU) incorporated in the 1401 processing unit. A tape unit along with the TAU forms an automatic cycling unit. Multiple tape units are parallel-connected to the TAU and selected by gating from the system tape controls. Various tape operations are selected by gating the proper call line to the TAU. Only one of the tape units can function at one time, except for the rewind operation, followed by the same or any other unit at a later interval.

Functional Controls

The IBM 1401 Tape Features contain no new components. It is composed of logic circuitry located in the processing unit. The circuits are divided into two groups, whose duties can be considered separately.

Tape Adapter Unit

The Tape Adapter Unit (TAU) is a standardized circuit package that can be adapted for use in any tape processing system. Early 1401 systems used TAU-2, which operates with IBM 729 Tape Drives Models 2 and 4. Later systems use TAU-9, which allows use of the IBM 7330 Tape Drive in addition to the IBM 729 models. The two units are functionally the same. The *CE Manual of Instruction on Tape Adapter Units* (Form 223-6847) details the function of the unit. The following paragraphs show its relation within the system.

On one side the TAU handles the data and control to and from the tape units. On the other side it handles data to and from the system. It controls the tape operations on command from the system, and serves to interlock and synchronize the system. Internally the TAU contains the clocking and cycle controls for all tape operations. A register system serves as the buffer for both read and write operations. Validity-checks are provided at several positions to insure accurate data transmission.

Clocking within TAU must provide for tape read, tape write, and delays of both long and short term. A four stage binary clock, driven with a clamped oscillator, serves to control the read operation. The write clock is a four stage binary ring driven by a gated free-running oscillator. These two oscillators are of like frequency to read and write a given speed and density. Each speed-density combination requires a pair of oscillators, which are selected to feed the clocks. The delay counter is driven by a high-frequency oscillator for delays in the microsecond range, and with a low frequency oscillator for delays in the millisecond range. The speeds of the oscillators are matched to the model of the tape unit.

A read-write register, composed of seven bit triggers, serves as a buffer between the tape unit and the system. It functions for both read and write operations with transfers gated by the operation controls. Validity is checked on all entries.

Two read registers, named for their entry from tape, are used to determine if a valid character has been sensed. Read-A is biased to read high-level signals well above the noise level. Read-B is biased to read just above the optimum noise level. A good signal with low noise reads correctly in both registers. The read-A register is validity-checked. Read-A is normally used for transfer unless found invalid. If read-B is used, it is checked after transfer to the R-W register. On a read-after-write, the two registers are compared for proof-of-validity.

Validity is further checked on a write operation by checking the echo signal from the write drivers in the tape unit. On both read and read-after-write, a count of the bits is made in the longitudinal redundancy check register. At the end of the write operation, the reset of the write triggers writes a character to serve as a check on subsequent readings. On a valid operation, the register ends with a blank condition after reading the check character (both read and write).

Controls throughout the tape system are interlocked on a call-and-response basis. An instruction is sent to TAU from the system. After controls are set, it relays the call to the proper tape unit. When the tape unit has been conditioned, it responds to TAU to advance the controls. TAU in turn signals the system, when data is being transferred or when the operation is completed.

Tape I-O Controls

The Tape I-O Control is composed of the circuits required to control the Tape Adapter Unit (TAU) from available system instructions. The same selection circuits are also used for the optional serial input-output feature. The two operations are serviced by the same basic group of operation codes.

The operation code is decoded in the normal manner and is used to gate much of the remaining instruction decode. The portion of the instruction specified by the normal A-address is sensed character-by-character as it passes through the A-register. The % character in the hundreds position sets a UNIT SELECT latch to indicate the use of the tape I-O controls. The tens position denotes the type of operation. For a normal tape operation, the U-character is not used. Other characters

specified for optional tape features or the I-O attachment set individual latches. The unit digit of the A-address designates the tape unit selected. Each digit sets its respective latch. The d-characters of the instructions set individual latches to gate their assigned functions.

The tape I-O controls contains a clock control system, which synchronizes the 1401 processing unit to the TAU controls. When the read-in of the instruction is complete and the 1401 has completed its setup function, the clock is stopped by preventing a set of the first trigger. As a data transfer or program advance is required by TAU, a START CLOCK signal is developed, which allows setting the clock trigger. In most operations the clock advance is one cycle at a time.

The A-register in the processing unit serves as the receiving register. The validation of the character on both read and write, however, is a function of the control area. The bit redundancy must be changed and word-marks controlled depending on the type of operation.

Operation Execution

Move Tape — M (% UX) (BBB) d

A move tape instruction starts the tape unit specified by the normal (AAA) address. Data from the selected tape unit enters core storage beginning at the location specified by the (BBB) address. The d-portion of this instruction determines whether the function is a tape-read or tape-write operation. Word-marks are not affected by a move tape instruction.

MOVE TAPE AND READ

Example: M (% U2) (419) R

Read an even-redundancy record from Tape Unit 2 to 1401 core storage in a tape move operation. The first tape record character read is moved to storage location 419, the next character to storage location 420, etc., until transmission is stopped by an inter-record gap on tape, or by a group-mark with a word-mark in core storage.

OBJECTIVES (MOVE TAPE AND READ)

The following objectives are required to perform the Move Tape and Read operation:

A. Set Move status.

B. Select the desired tape unit.

C. Select the starting storage location.

D. Enable read call.

E. Develop TAU Read Controls.

F. Start 1401 clock.

G. End data operation.

Objectives A, B, C, and D comprise the instruction phase of the operation, which is nine I-cycles long. Objectives E, F, and G comprise the execution phase of the instruction, which consists of a B-cycle only, and varies according to tape record length. During the instruction phase, the 1-character Op code is transferred from storage through the B-register to the Op register and remains there until the start of a new operation. The (AAA) and (BBB) addresses are transferred from core storage, pass through the B- and A-registers, and enter the A-address and B-address registers, respectively. The d-character is transferred from core storage to the B-register and then is entered into the A-register where it remains until data is read from tape and moved into the A-register. During the execution phase which is of variable length, the B-address register controls the read-out of core storage each cycle (being modified after each cycle) until the process is stopped by an inter-record gap on tape, or by a B-register group-mark with word-mark in core storage.

LOGIC DETAIL (MOVE TAPE AND READ)

A. Set Move Status (Figure 48): During I-Op time, a (WM) B4 character is read from core storage into the B- and Op registers. The Op register contents are then decoded as M (move) to set move status.

ILD 15	Op Register (WM) B 4 (M)	Time 060-090 I-Ring Op B-Register (WM) B4
ILD 80 A1	Move Operation	Op Decode (WM) B4

The move operation signal, which prevents the transfer of word-marks from storage to tape, is used in circuits external to the tape I/O adapter.

B. Select the Desired Tape Unit (Figure 48): During I-Ring-1 time, the high-order position of the (AAA) address sets the UNIT SELECT latch. This latch output develops tape operate, +1-tape mode, I/O operate, and unit select gate signals. The I/O operate signal develops an A-cycle elimination pulse which gates-off the A-CYCLE latch to prevent A-cycles while in tape operation. During I-Ring-2 time, the next order position of the (AAA) address selects the REDUNDANCY latch while during I-Ring-3 time, the low-order position selects the tape unit to be operated on.

ILD 80 B2	Set Unit Select Latch	Time 090-000 I-Ring-1 A-Register A-84 (%) Not Manual Op
ILD 80 C2	Select Redundancy Latch	Time 030-045 I-Ring-2 A-Register BA 2 (B) Unit Select

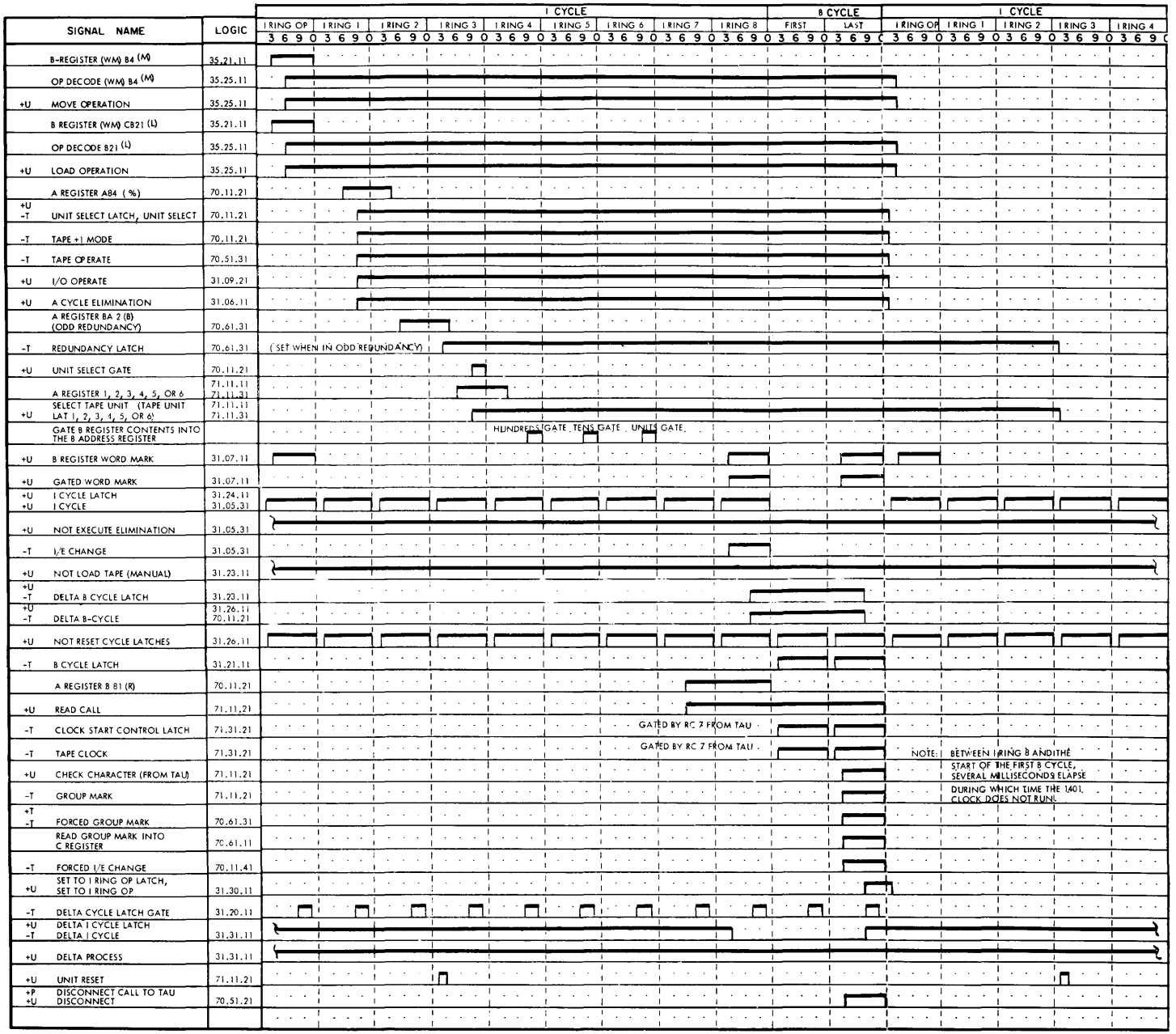


Figure 48. Timing for Move (Load) Tape Read Operation

If the A-register contains (B), the REDUNDANCY latch is *set* and the redundancy is *odd*. If the A-register does not contain (B), the REDUNDANCY latch is *not set* and redundancy is *even*.

ILD 80 B2	Enable Unit Select Gate	Time 090-000 I-Ring-3 Unit Select
ILD 80 A3-B3	Select Tape Unit Latch	Unit Select Gate A Register 1, 2, 3, 4, 5, or 6 Not Manual Op

The TAPE UNIT latch outputs are connected by cable to their respective tape units.

C. Select the Storage Location (Figure 48): During I-Ring-4 through I-Ring-6 time, the (BBB) address selects the storage location at which the operation begins.

ILD 17 C3	Gate B-register Contents into Hundreds-Thousands Position of B-Address Register	Time 090-000 I-Ring-4 B-Register Contents
ILD 17 C5	Gate B-Register Contents into Tens Position of B-Address Register	Time 090-000 I-Ring-5 B-Register Contents
ILD 17 C6	Gate B-Register Contents into Units Position of B-Address Register	Time 090-000 I-Ring-6 B-Register Contents

D. Enable Read Call (Figure 48): At I-7 time, the d-character of the instruction (which indicates the specific function under the operation) enters the B-register and the A-register. At I-8 time, the next Op code with its word-mark enters the B-register. The presence of a word-mark in the B-register prevents reset and read-in of the A-register, and generates an I/E Change. With A-cycle Elimination, Delta B-cycle is set and a Read Call to TAU is initiated.

ILD 14 A5	Gated Word Mark	Not I-Ring Op B-Register Word-Mark
ILD 13 C3	I/E Change	I-Cycle Gated Word Mark Not E-Phase Elimination
ILD 13 C4	Set Delta B-Cycle Latch	I/E Change I-Cycle Latch A-Cycle Elimination Not Load Tape (Manual)
ILD 13 C5	Set B-Cycle Latch	Time 000-060 Delta B-Cycle Not Reset Cycle Latches Address Register Keys Off
ILD 80 C2-C3	Enable Read Call	Unit Select Delta-B or B-Cycle A-Register B-81 (R)

When Read Call is enabled, a Read Call to TAU is initiated. Read Call remains enabled until reset by the fall of the UNIT SELECT latch on the next I-Op.

E. Develop TAU Read Controls: A Read Call to TAU activates the Read-Only, Busy, and Read-Delay triggers. The Read-Only trigger output enters the tape unit where read status is set. The tape unit then returns Select and Read to TAU which turns ON the Go trigger to start the tape moving. At the same time, the Read-Delay trigger starts the Delay counter. When the delay operation is complete, the Read-Condition trigger comes ON, conditioning the read circuits for operation.

F. Start 1401 Clock (Figure 48): With each TAU read clock cycle, an RC-7 pulse enters the tape control circuits and is switched with the Read Call output to set the Clock Start Control trigger. The output of this trigger is then switched with a Delta B-cycle signal to start the 1401 clock.

ILD 80 C4	Set the Clock Start Control Latch	Time 015-000 RC-7 Read Call
ILD 80 B5	Enable Tape Clock	Clock Start Control Latch Delta B-Cycle

The TAPE CLOCK signal enables CLOCK CONTROL (70.-11.41.2) which in turn enables START CLOCK (31.10.11.2). This operation continues with each character read from tape.

During Move Tape and Read, the data flow path (Figure 46) is:

1. From tape to TAU (read-write register).
2. To the TAU I/O Adapter (through read gating circuits and the A-register).
3. To the Inhibit drivers.
4. To core storage.

Just prior to character entry from the A-register to a core storage location, the old contents of that location are transferred to the B-register, tested for the presence of a group-mark with word-mark (see heading G), and then lost except for the WM bit. The new contents are then read into storage from the A-register. Following each storage read-in, the address register modifier augments the contents of the B-address register by plus-one. Thus each tape character is stored in the next higher location than was the preceding tape character.

G. End Data Operation (Figure 48): Data transmission ends when an inter-record gap is sensed on tape, or a group-mark with word-mark is sensed in the B-register. An inter-record gap on tape initiates control circuits in TAU which will terminate the read operation (see the *Tape Adapter Unit Manual,* Form 223-6847). A group-mark with word-mark contained in

the B-register starts the 1401 clock and enables FORCED GROUP-MARK which forces an I/E Change, and conditions the 1401 for a new instruction. However, the tape continues to move until an inter-record gap is sensed.

An inter-record gap ends the Move Tape and Read operation as follows:

ILD 80 C4	Enable Group-Mark	Read Call Check Character (from TAU)
ILD 80 D5	Forced Group-Mark	Group-Mark
ILD 80 C6-D6	Read Group-Mark into A-Register	Forced Group-Mark Read Call
ILD 80 D4	Forced I/E Change Clock Control	Forced Group-Mark
ILD 13 C3	I/E Change	Forced I/E Change
ILD 80 B1	Reset Unit Select Latch	I-Ring Op
ILD 80 C4	Disable Read Call	Not Unit Select
ILD 80 B2	Enable Unit Reset	Time 030-060 Unit Select Latch (From Next Tape Instruction) I-Ring 3
ILD 80 C2	Reset Redundancy Latch	Time 030-060 I-Ring 2 Unit Select (From Next Tape Instruction)

A group-mark with word-mark transferred from storage to the B-register ends the Move Tape and Read operation as follows:

ILD 80 D2	Forced Group-Mark Word-Mark	Disconnect

Then the same sequence follows as that described for an inter-record gap.

MOVE TAPE AND WRITE

Example: M (% B4) (525) W

Write an odd-redundancy record in tape unit 4 beginning with the contents of storage location 525, then 526, etc. End at the first storage location containing a group-mark with word-mark. This results in an inter-record gap on the tape.

OBJECTIVES (MOVE TAPE AND WRITE)

The following objectives are required to perform the Move Tape and Write operations:

A. Set move status.

B. Select the desired tape unit.

C. Select the starting storage location.

D. Enable Write Call.

E. Develop TAU write controls.

F. Start 1401 clock.

G. End data operation.

Objectives A, B, C, and D comprise the instruction phase of the operation; objectives E, F, and G comprise the execution phase of the operation (in a manner identical to that described for Move Tape and Read).

LOGIC DETAIL (MOVE TAPE AND WRITE)

Objectives A through C: These objectives follow the procedure and timing described for Move Tape and Read, headings A through C.

D. Enable Write Call (Figure 49): The procedure and timing is the same as that described for Move Tape and Read, heading D, except that a Write Call to TAU, rather than a Read Call to TAU, is initiated.

ILD 80 B4	Enable Write Call	Unit Select Latch Delta B-Cycle or B-Cycle A-Register A-42 (W)

When Write Call is enabled, a Write Call to TAU is initiated. Then, a Write Gate signal is enabled.

ILD 80 B4	Enable Unit Write Gate	Write Call Not Forced Group-Mark Not Tape Readdress

When both the TAU write circuits and the 1401 clock have been conditioned, writing begins.

E. Develop TAU Write Controls: A Write Call to TAU activates the Write, Busy, and Write Delay triggers. The Write trigger output enters the tape unit where write status is set. The tape unit then returns Select and write to TAU which turns on the Go trigger to start the tape moving. At the same time, the Write-Delay trigger starts the Delay counter. When the delay operation is complete, the Write-Condition trigger comes ON. The write clock now starts, and continues to turn in repetitive cycles under control of the Write-Condition trigger. At each WC-9 time, a write pulse is sent to the tape unit to initiate the writing action. For more detailed information on TAU write-controls, and an analysis of timing sequence, refer to the *Tape Adapter Unit Manual* (Form 223-6847).

F. Start 1401 Clock (Figure 49): A Write Call signal, switched with the I-CYCLE latch output, starts the 1401 clock under tape control. When the I-CYCLE latch is reset because of an I/E phase change, the CLOCK START CONTROL latch output is switched with a Delta B-cycle pulse to take control of the 1401 clock. On a write

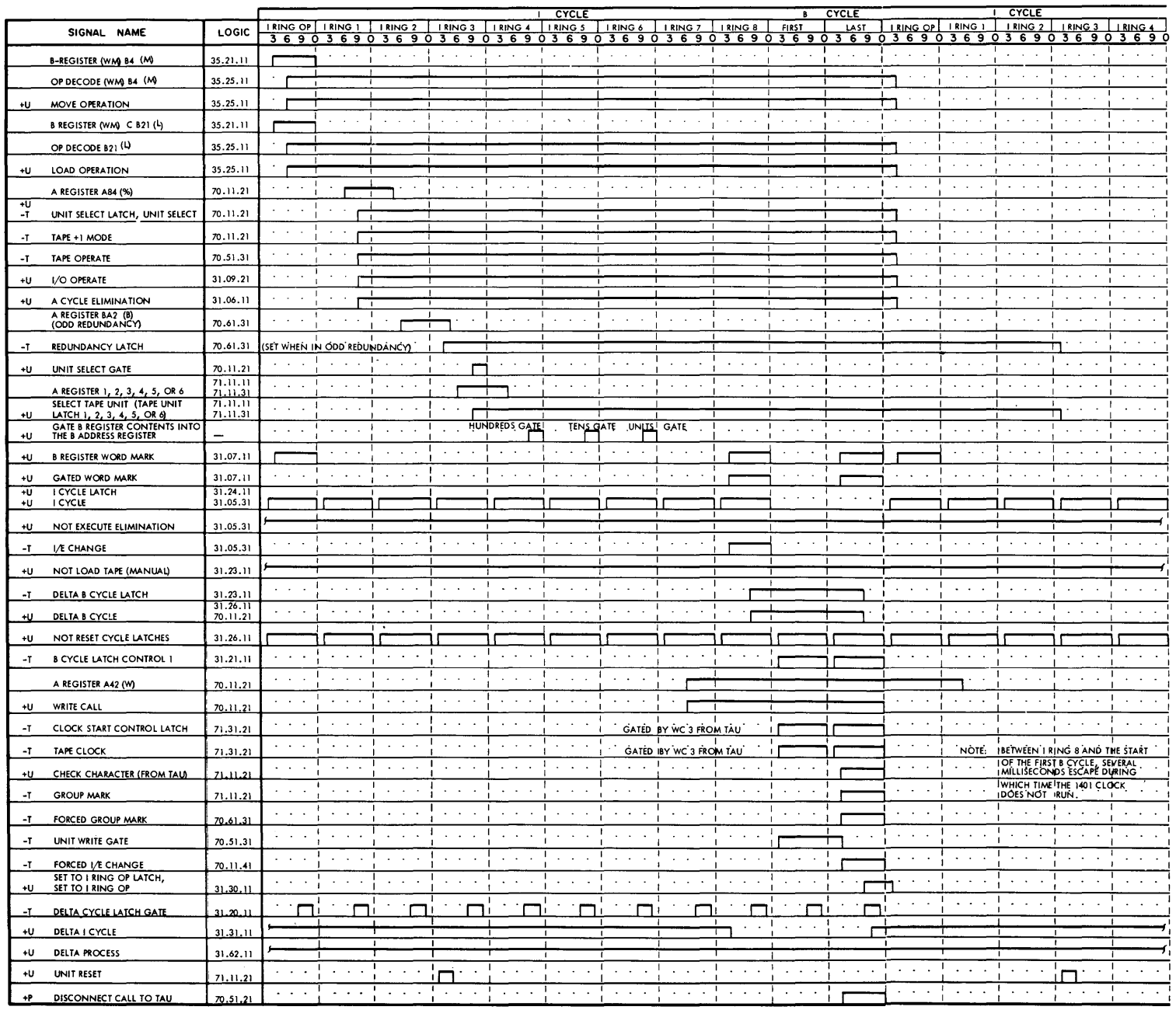


Figure 49. Timing for Move (Load) Tape and Write Operation

Tape Character No	1	Word Separator	2	3
Tape Code (Even Redundancy)	CBA 1	A 841	CBA 2	BA 21
Tape Meaning	A	Word Separator	B	C
1401 Code	BA 1	Bit in the Eighth Plane	BA 2	CBA 21
1401 Storage Location	100	101		102
1401 Meaning	A	Word Mark with B		C

Figure 50. Read Word Separator from Tape

operation the clock is allowed to run one B-cycle before stop to fill the B-register. The CLOCK START CONTROL latch is set at WC-3 time and reset at time 000-015.

ILD 80 B4	Enable Tape Clock	Write Call I-Cycle latch Not I/O Select

The tape clock signal enables CLOCK CONTROL (70.-11.41.2) which in turn enables START CLOCK (31.10.11.2). This operation continues until an I/E phase change occurs. The 1401 clock is then controlled as follows:

ILD 80 B4	Set the Clock Start Control Latch	WC 3
ILD 80 B5	Enable Tape Clock	Clock Start Delta B-Cycle

During Move Tape and Write, the data flow path (Figure 46) is:

1. From core storage to the B-register
2. To the TAU I/O Adapter (through write gating circuits) and also to the inhibit drive circuits (for regeneration in storage).
3. To the selected tape unit (data lines are direct, control lines go through TAU).

Following each storage read-out and regeneration, the address register modifier augments the contents of the B-address register by one. Thus each character written on tape comes from the next higher storage location than the preceding character written.

G. End Data Operation (Figure 49): Data transmission ends whenever a group-mark with word-mark is sensed in the B-register. Procedure and timing is the same as that described for Move Tape and Read, heading G, except that the Write Call signal, rather than the Read Call signal, is disabled. Also, upon receiving Disconnect Call, TAU writes a check character, reads tape for checking, stops the tape, and resets the write circuits. (See the *Tape Adapter Unit Manual,* Form 223-6847.)

The write call and write gate are disabled as follows:

ILD 80 D2	Enable Disconnect Call to TAU	Unit Select B-Register Group-Mark
ILD 80 B4	Disable Unit Write Gate	Disconnect
ILD 80 B1	Disable Unit Write Gate	I-Op

Load Tape — L (% UX) (BBB) d

A Load Tape instruction is basically the same as a move tape instruction; however, the L-Op code affects word-mark identification of data. That is, this instruction must be used when word-marks are needed for identification of each new word on tape. If data is written on tape by means of a Load instruction, it must also be read from tape by means of a Load instruction, to insure proper translation of data from tape to storage.

LOAD TAPE AND READ

Example: L (% U3) (T 32) R

Read an even-redundancy record from Tape-Unit-3 to 1401 core storage in a tape load operation. The first tape record character is loaded into storage location 1332, the next into 1333, etc. until transmission is stopped by an inter-record gap on tape, or by a group-mark with word-mark in core storage. During transmission of data, the special characters which separate words on tape are translated to word-marks in core storage. Each word-separator read from tape causes a word-mark to be associated with the next tape character entered into core storage (Figure 50).

OBJECTIVES (LOAD TAPE AND READ)

The following objectives are required to perform the Load Tape and Read operations:

A. Set load status.
B. Select the desired tape unit.
C. Select the starting storage location.
D. Enable read call.
E. Develop TAU read controls.
F. Start 1401 clock.
G. End data operation.

Objectives A, B, C, and D comprise the instruction phase of the operation; objectives E, F, and G comprise the execution phase of the operation (in a manner identical to that described for Move Tape and Read).

LOGIC DETAIL (LOAD TAPE AND READ)

A. Set Load Status (Figure 48): During I-Op time, a (WM) CB-21 character is read from core storage into the B- and Op registers. The Op register contents are then decoded as L (Load) to set Load status.

ILD 15	Op Register B 21 (L)	Time 060-090 I-Ring Op B-Register (WM) CB 21
ILD 81 A1	Load Operation	Op Decode B 21

The L-Op code signal enables the transfer of word separator characters on tape to word-marks in core storage during a Load Tape and Read operation.

Objectives B through F (Figure 48): These objectives are identical to those described for Move Tape and Read, headings B through F, except for the transfer of word-marks.

Conversion of a word separator character on tape to a word-mark in core storage requires even-redundancy operation. The word separator character enters the A-register from tape, and enables a WS read signal which sets the TAPE READDRESS latch at time 045-030. The ON-OUTPUT of the TAPE READDRESS latch sets the NOT READDRESS latch at 000-030. The tape readdress signal serves to block the reset of the address register and prevents the gate-out of the B-address register. The ON-OUTPUT of the LAST CYCLE latch develops an inhibit-word-mark signal which generates a word-mark in core storage.

The addressing and data flow scheme is as follows (Figure 51):

1. Address a core storage location from the address register and read in a word separator character plus a word-mark.
2. Augment the contents of the address register by one via the address register modifier and gate the modified contents, serially, into the B-address register; also, block the address register reset.
3. Block the B-address register gate-out; then readdress the core storage location from the address register and read in the next tape character plus another word-mark (the word separator character plus word-mark read into storage in 1 above, is obliterated).

ILD 81 D1	Sense Word Separator	A Register-A841 (WD Char) Read Call
ILD 81 D2	Tape Readdress	Word Separator
ILD 81 D2	Not Readdress Latch	Time 000-030 Tape Readdress
ILD 81 D3	Inhibit Word Mark	Not Readdress Latch Read Call

During a word-separator character to word-mark conversion, operation of the address register, B-address register, and address register modifier is normal with this exception. The address register reset and the B-address register gate-out are blocked by a tape-readdress signal to allow a readdress of the core storage location containing the word separator character.

ILD 26 C1	Set the Readdress Latch (Cycle Control)	Tape Readdress
ILD 26 D2	Disable the Address Register Reset	Readdress Latch On (Cycle Control)
ILD 17 C1	Disable the B-Address Register Gate-Out	Readdress Latch On (Cycle Control)

LOAD TAPE AND WRITE

Example: L (% U 5) (414) W

Write an even redundancy record on Tape Unit 5 beginning with the contents of storage location 414. Terminate the write operation at the first storage location containing a group-mark with word-mark. This will result in an inter-record gap on the tape. During transmission of data, the word-marks which separate words in core storage are translated to word separator characters on tape. A word-mark associated with any

position in storage causes a word separator character to be written automatically on tape one character ahead of the character containing the word-mark (Figure 52).

OBJECTIVES (LOAD TAPE AND WRITE)

The following objectives are required to perform the Load Tape and Write operation:

A. Set Load status.
B. Select the Desired Tape Unit.
C. Select the Starting Storage Location.
D. Enable Write Call.
E. Develop TAU Write Controls.
F. Start 1401 Clock.
G. End Data Operation.

Objectives A, B, C, and D comprise the instruction phase of the operation; objectives E, F, and G comprise the execution phase of the operation; in a manner identical to that described for Move Tape and Read.

LOGIC DETAIL (LOAD TAPE AND WRITE)

A. Set Load Status (Figure 49): Follow the procedure and timing described for Load Tape and Read, heading A.

The L-Op code signal enables the transfer of word-marks in core storage to word separator characters on tape during a Load Tape and Write operation.

Objectives B through F (Figure 49:) These objectives are identical to those described for Move Tape and Write, headings B through F, except for the transfer of word-marks.

Conversion from a word-mark in core storage to a word separator character on tape requires even-redundancy operation. Each character read from storage enters the B-register. The B-register contents are switched with a Unit-Write-Gate signal and transferred to TAU write control lines, and are also regenerated in storage. Whenever a character containing a word-mark enters the B-register, the ON-OUTPUT of the WORD-MARK latch sets the READDRESS latch at time 045-030.

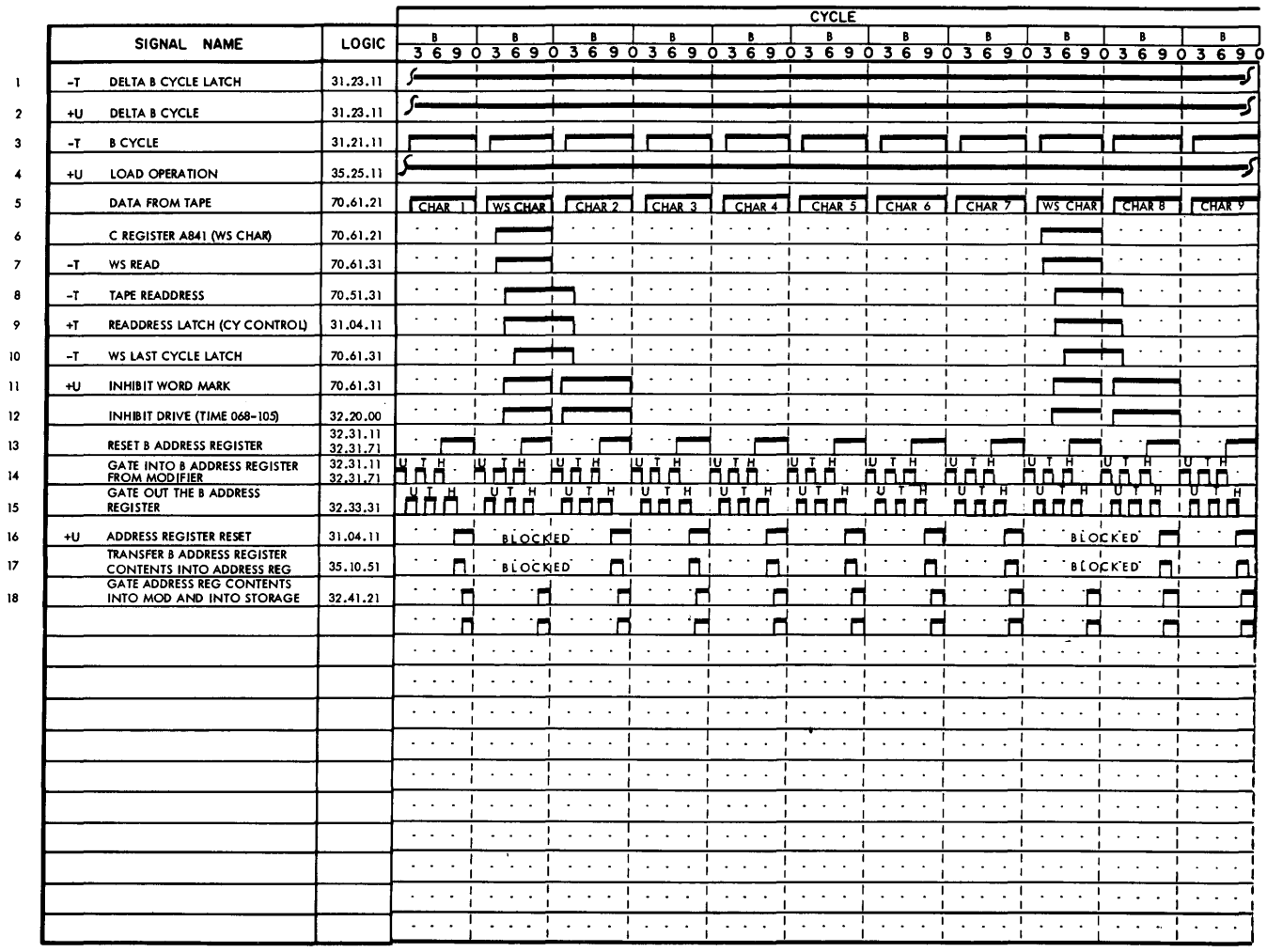


Figure 51. Timing for Load Tape and Read a Word Separator Character

The ON-OUTPUT of the READDRESS latch is switched with the OFF-OUTPUT of the NOT READDRESS latch to develop Tape Readdress, WS on Tape, and Not Unit Write Gate signals. The tape readdress signal blocks the reset of the address register and prevents the normal gating of the B-register contents to the TAU write-control lines.

At time 015-030 the ON-OUTPUT of the READDRESS latch sets the NOT READDRESS latch. This disables the Tape Readdress, WS On Tape, and Not Unit Write Gate signals, and prevents another immediate readdress from taking place.

The addressing and data flow scheme is as follows (Figure 53):

1. Address a core storage location from the address register and read a character plus word-mark from storage into the B-register.
2. Block the normal entry to the TAU write control lines from the B-register, and force a word separator (A-841) on the lines; also, regenerate the B-register contents into storage.
3. Augment the contents of the address register by plus-one via the address register modifier, and gate the modified contents, serially, into the B-address register; also, block the address register reset.
4. Block the B-address register gate-out; then re-address the core location from the address register and re-read the character plus word-mark (as described in 1 above) from storage into the B-register. Release the block of the normal entry to the TAU write control lines.

ILD 81 D2	Readdress Latch (Tape)	Time 045-075 Not Readdress Latch Off Write Call Not Disconnect B-Register Word Mark L-Op Code
ILD 81 D3	Tape Readdress	Readdress Latch On Not Readdress Latch Off
ILD 81 D3	WS On Tape	Readdress Latch On
ILD 81 D3	Unit Write Gate	Readdress Latch Off Not Disconnect Write Call
ILD 80 B5	Force Lines, A, 8, 4, 1, to TAU	WS On Tape

During a word-mark to word separator conversion, operation of the address register is normal with this exception. The address register reset and the B-address register gate-out are blocked by a tape readdress signal to allow a readdress of the core storage location containing the word-mark.

ILD 26 C1	Readdress Latch (Cycle Control)	Time 060-030 Tape Readdress

1401 Character	A	Word Mark with B		C
1401 Location	100	101		102
1401 Code	BA 1	Bit in the Eighth Plane	BA 2	CBA 21
Tape Meaning	A	Word Separator	B	C
Tape Code (Even Redundancy)	CBA 1	A 841	CBA 2	BA 21
Tape Character No	1	Word Separator	2	3

Figure 52. Write Word Separator from Tape

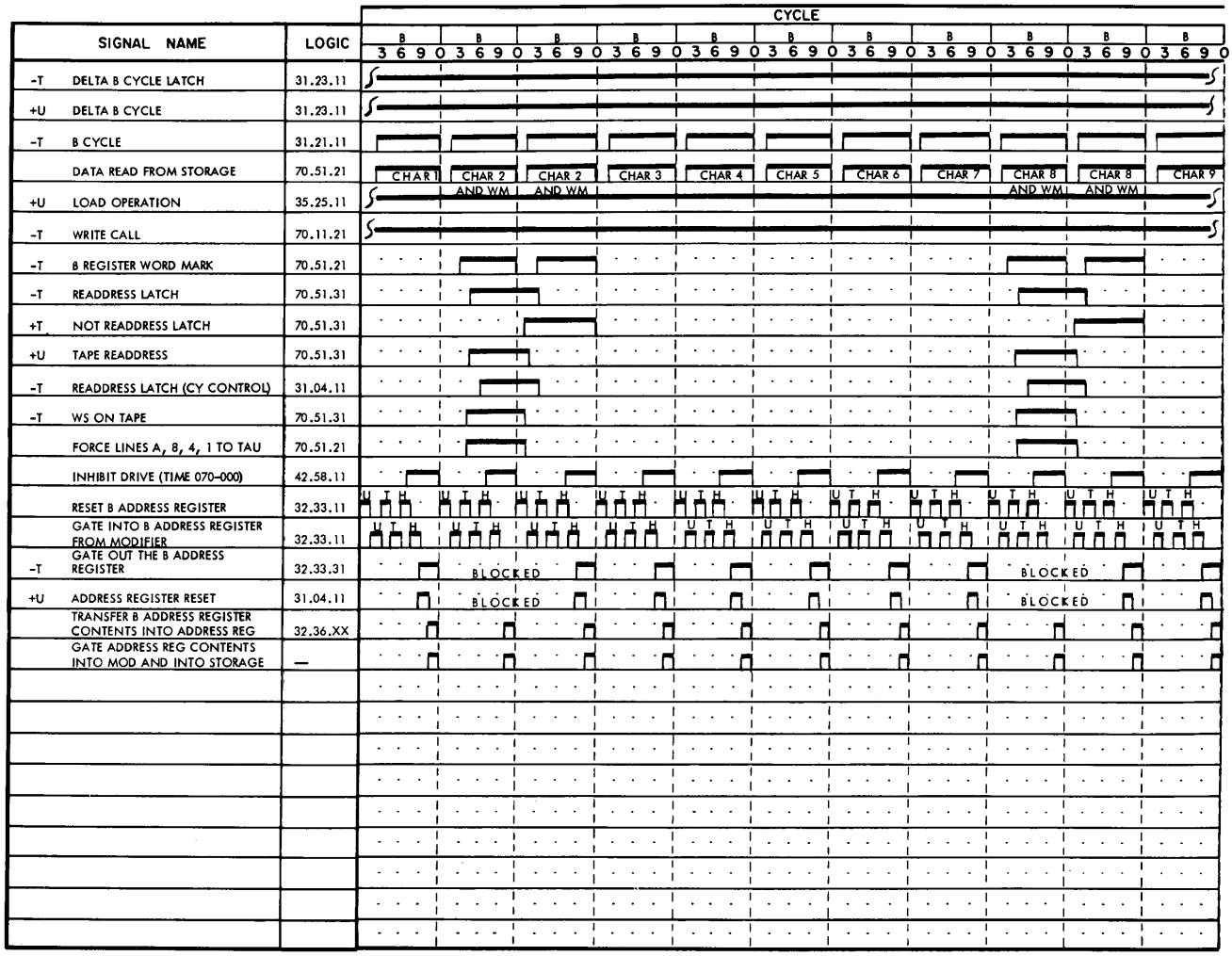


Figure 53. Timing for Load Tape and Write a Word Separator Character

ILD 26 D2	Disable the Address Register Reset	Readdress Latch On (Cycle Control)
ILD 17 C1	Disable the B-Address Register Gate-Out	Readdress Latch On (Cycle Control)

After a readdress operation has been generated, the NOT READDRESS latch is set, and additional readdressing is prevented until a word-mark from a new location enters the B-register.

ILD 81 D2	Not Readdress Latch	Time 015-000 Readdress Latch On

Unit Control — U (% UX) d

A unit control instruction starts the tape unit specified by the normal (AAA) address. The d-character portion of this instruction determines the specified function under the operation—either backspace, erase forward, write tape mark, rewind tape, diagnostic read, or rewind and unload tape. Since no (BBB) address exists with a unit control instruction, no data in core storage is affected.

BACKSPACE TAPE

This instruction causes the specified tape unit to backspace over one tape record. The backspace operation stops when an inter-record gap is sensed.

Example: U (% U4) B

Backspace Tape Unit 4 until an inter-record gap is sensed on tape.

OBJECTIVES (BACKSPACE TAPE)

The following objectives are required to perform a backspace operation:

A. Set unit control status.

B. Select the desired tape unit.

C. Enable backspace call.

D. Develop TAU backspace controls.

E. Start 1401 clock.

F. End data operation.

The instruction phase of a backspace operation is six I-cycles long. The execution phase which is of

variable length, is controlled by TAU, but the 1401 is released for new operations. During the instruction phase, the 1-character Op code is transferred from core storage to the Op register. The (AAA) address is transferred from core storage, passes through the B- and A-registers, and enters the A-address register. The d-character is transferred from core storage to the B-register and then enters the A-register. Since the execution phase is eliminated within the 1401 (at I-Ring Op time the Op and A-registers are reset) the new operation can begin immediately. With no execution phase, the contents of the A-address register remain unchanged.

LOGIC DETAIL (BACKSPACE TAPE)

A. Set Unit Control Status (Figure 54): During I-Op time, a (WM) A4 character is read from core storage into the B- and Op registers. The Op register contents are then decoded as U (Unit Control) to set unit control status and enable the U-Op code signal.

ILD 15 2B	Op Register (WM) A 4 (U)	Time 060-090 I-Ring Op B Register (WM) A4

ILD 82 A1	U Op Code	Op Decode (WM) A4

The U-Op code signal develops an execute-eliminate signal which blocks the execution phase of the operation within the 1401.

ILD 13 D3	Execute Eliminate	U Op Code

B. Select the Desired Tape Unit (Figure 54): Follow the procedure and timing as described for any of the previous operations; e.g., Move Tape and Read, heading B.

C. Enable Backspace Call (Figure 54): At I-Ring-4 time, the d-character (which defines the specific function under the operation) enters the B-register and transfers to the A-register. At I-ring-5 time the next Op code with its word-mark enters the B-register. The B-register word-mark output is switched with the unit-select and U-Op code signals, producing U-Op code gate which in turn is switched with the contents of the A-register to initiate Backspace, and Backspace Call to TAU.

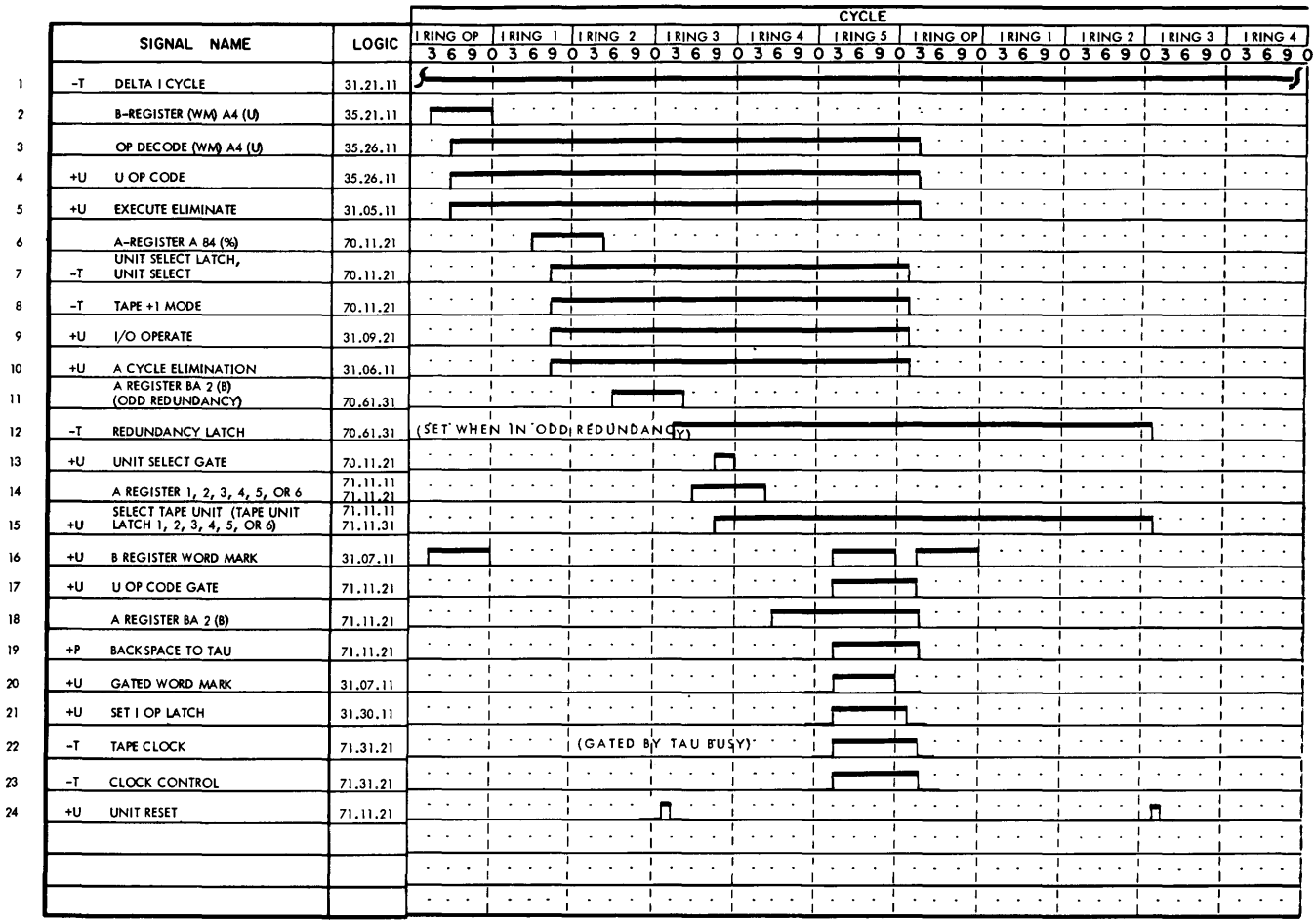


Figure 54. Timing for Backspace Tape Operation

ILD 82 B3	U-Op Code Gate	B-Register Word Mark Unit Select U-Op Code
ILD 82 D4	Backspace Call (to TAU)	U-Op Code Gate A-Register BA 2 (B)

The 1401 is released from the execution of the operation as follows. During I-Ring-5 time, the presence of a word-mark in the B-register generates a gated word-mark signal. This signal is switched with execute-eliminate to set the SET-1-OP latch, which enables SET-TO-I-RING OP. TAU has now taken control of the backspace operation, and the d-character contained in the A-register is no longer needed; thus at I-Ring Op time a new instruction can be processed.

ILD 14 A5	Gated Word Mark	Not I-Ring Op Not Prevent Gated Word Mark B-Register Word Mark
ILD 14 B4	Set-to-I Ring Op	Gated Word Mark Execute Eliminate
ILD 14 A6	I-Ring Op Latch	Time 015-030 Delta I-Cycle Set to I-Ring Op

D. Develop TAU Backspace Controls: Backspace is a read operation in the backward direction. The data being read is not sent to the 1401 A-register. A backspace call to TAU activates the backspace trigger (unless the tape is at load point in which case TAU takes no action). The Backspace trigger output turns on Busy and enables the Delay counter.

All tape movement in a backward direction is done with the tape unit in read status. If the tape unit is in write status, TAU must set it to read status. Because a write-to-read status change deposits noise on tape, the tape is first moved forward to insure that the noise deposition is far enough out on the tape to be erased in a subsequent write operation.

With the tape unit in write status, the Delay counter output turns on the Go trigger to start the tape moving in the forward direction. At a predetermined time later, the Delay counter output resets the Go trigger, stops the tape, and then resets and restarts the Delay counter for the backward operation.

If the tape unit is already in read status at the start of the operation, no read-write status change is necessary, and the tape does not move forward. The Delay counter output turns on the Backward trigger to set backward status.

When Select, Ready, and Read are returned to TAU, a Delay counter output turns on the Go trigger. Because of backward status, the tape starts moving backward. When the delay operation is complete, the Read-Condition trigger comes ON, conditioning the read circuits for operation. When the read head senses the first bit of the first tape character (in this case the check character), the TAU read clock starts, and a read cycle is taken. With the sensing of the first bit of each subsequent character, a read clock cycle is taken. At each RC7 time the Read Disconnect Relay (RDD) trigger is turned ON. The RDD trigger output conditions the Delay counter to start. If left running, the Delay counter output resets the backspace operation; however, as long as first bits are being read at timed intervals, the RDD trigger will be reset before it can accomplish ending the operation. When the first bit is not read for a longer interval of time, the RDD trigger remains ON and the output of the Delay counter resets the backspace operation. This longer interval of time is the time between the beginning of a record to the check character of the previous record. For additional and more detailed information on TAU backspace controls and an analysis of timing sequence, refer to the *Tape Adapter Unit Manual* (Form 223-6847).

E. Start 1401 Clock (Figure 54): With BACKSPACE CALL to TAU enabled, the TAU Busy signal comes ON and is switched with Backspace to develop a Tape Clock signal. The Tape Clock signal develops Clock Control which in turn enables Start Clock. With this Start Clock signal, Clock Control is returned to the 1401.

ILD 82 D4	Enable Tape Clock	TAU Busy U Op Code Gate
ILD 80 D4	Enable Clock Control	Tape Clock

F. End Data Operation (Figure 54): The Backspace operation ends with an inter-record gap on tape (See the *Tape Adapter Unit Manual* – Form 223-6847.)

The UNIT SELECT, REDUNDANCY, and TAPE UNIT latches must be reset before another tape operation can be performed. Reset is as follows:

ILD 82 B2	Reset Unit Select Latch	I-Ring Op
ILD 82 B3	Enable Unit Reset	Time 030-060 Unit Select Latch (from next Tape Instruction) I-Ring-3
ILD 82 A5	Reset Tape Unit Latch	Unit Reset
ILD 82 C3	Reset Redundancy Latch	Time 030-060 Unit Select (from next Tape Instruction) I-Ring-2

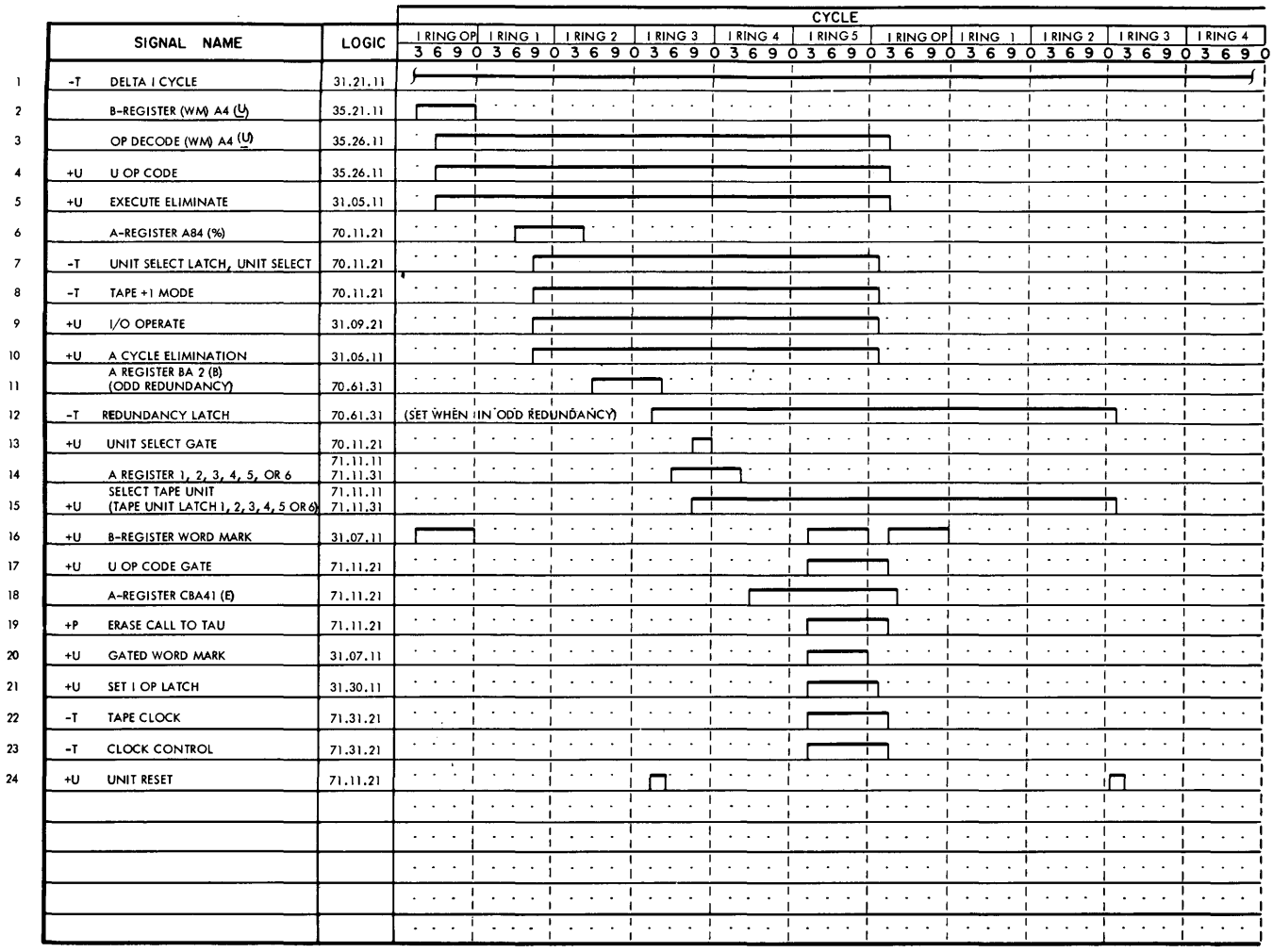


Figure 55. Timing for Erase Tape Operation

ERASE TAPE

Dust or damage to the magnetic tape is the most frequent source of errors detected during write operations. Since such imperfections are usually isolated, the 1401 contains an Erase Tape instruction which causes the specified tape unit to space forward 3½ inches, bypassing defective tape areas. As the tape is passed, this short length is erased so that extraneous data is not sensed when the tape is later read. The skip does not occur until the start of a subsequent tape write operation.

Example: U (% U5) E

Tape unit 5 erases 3½ inches of tape at the start of the next M (% U5) (BBB) W or L (% U5) (BBB) W instruction.

OBJECTIVES (ERASE TAPE)

The following objectives are required to perform an Erase Tape operation:

A. Set erase tape status.
B. Select the desired tape unit.
C. Enable Erase Call.
D. Develop TAU erase controls.
E. Start 1401 clock.
F. End operation.

The instruction phase of an erase tape operation is identical to that described for a backspace tape operation. The execution is carried out as part of the next tape write instruction.

LOGIC DETAIL (ERASE TAPE)

Objectives A and B (Figure 55): These objectives follow the procedure and timing as described for a Backspace Tape operation, headings A and B.

C. Enable Erase Call (Figure 55): Procedure here is identical to that described for Backspace Tape, except that the d-character contained in the A-register at I-Ring-5 time is CBA41 (E) rather than BA2 (B). The contents of the A-register are switched with the U-Op code gate to initiate Erase and Erase Call to TAU.

ILD 82 C4	Erase Call (to TAU)	U Op Code Gate A Register CBA41 (E)

D. Develop TAU Erase Controls: An Erase Call to TAU turns on the Erase trigger. The trigger output conditions the Delay counter for longer than normal write delay by duplicating load point status. Regardless of tape position, at the start of the next tape write operation a load point write delay is forced. This allows TAU to skip over a 3½ inch section of tape before writing begins. The Erase trigger is reset at the end of the subsequent tape operation. For additional information on TAU Erase Controls, refer to the *Tape Adapter Unit Manual* (Form 233-6847).

E. *Start 1401 Clock (Figure 55):* The Erase signal enables Tape Clock which in turn enables Clock Control. The Clock Control output develops Start Clock (31.-10.11.2). With this Start Clock signal, Clock Control is returned to the 1401.

ILD 82 D4	Tape Clock	Erase
ILD 80 D4	Clock Control	Tape Clock

F. End Operation (Figure 55): The Erase Tape operation is completed at the beginning of the next tape write operation. After the Erase Tape operation has been performed, the tape write operation continues.

The UNIT SELECT, REDUNDANCY, and TAPE UNIT latches are reset as described in the Backspace Tape operation, heading F.

WRITE TAPE MARK

A tape mark is a special character (8421) recorded following the last record on tape, to indicate an end-of-file condition. When the tape mark character is read back from tape, the end-of-file indicator is turned ON.

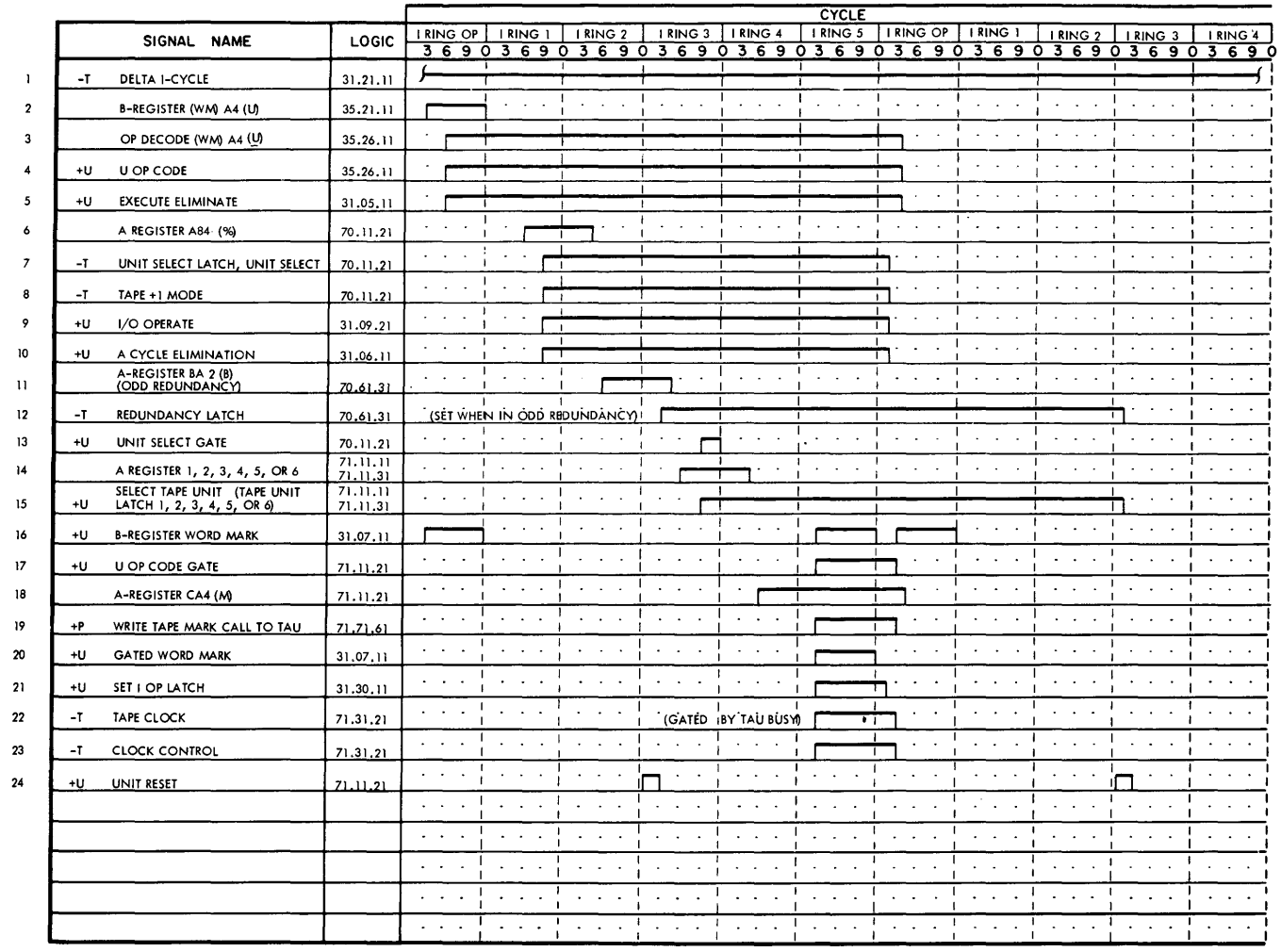


Figure 56. Timing for Write Tape Mark Operation

This signals the 1401 program that the end of the used portion of tape has been reached.
Example: U (% U1) M

Insert a tape mark following the last record written on tape 1.

OBJECTIVES (WRITE TAPE MARK)

The following objectives are required to perform a Write Tape Mark operation:

A. Set unit control status.
B. Select the desired tape unit.
C. Enable Write Tape Mark Call.
D. Develop TAU write tape mark controls.
E. Start 1401 clock.
F. End operation.

The instruction phase of a Write Tape Mark operation is identical to that described for a Backspace Tape operation, Execution. The execution phase is carried out as part of a 1-character Move Tape operation.

LOGIC DETAIL (WRITE TAPE MARK)

Objectives A and B (Figure 56): These objectives follow the procedure and timing as described for a Backspace Tape operation, headings A and B.

C. Enable Write Tape Mark Call (Figure 56): Procedure here is identical to that described for Backspace Tape, except that the d-character contained in the A-register at I-Ring-5 time is CB4 (M) rather than BA2 (B). The contents of the A-register are switched with the U-Op Code Gate to initiate Write Tape Mark and Write Tape Mark Call to TAU.

ILD 82 C4	Write Tape Mark Call (to TAU)	U-Op Code Gate A-Register CB4 (M)

D. Develop TAU Write Tape Mark Controls: On receiving Write Tape Mark Call, TAU activates the Write Tape Mark trigger, the normal write operation circuits, and the Write Disconnect Delay (WDD) trigger. The Write Tape Mark trigger output blocks the normal input data lines. Instead, it forces data lines 8, 4, 2, and 1 to the tape unit. These lines are switched with a write pulse (WC9) to write a tape mark on tape. The WDD trigger output conditions the end of the operation which stops in the normal manner after one write cycle. A check character is also written on the tape.

During normal read operation, the recognition of a tape mark in TAU is conditioned by the first-character trigger which is turned on by the read-condition trigger. If the first character is a tape mark, the First Character Tape Mark signal becomes available to the system. At RC7 time, the first-character trigger is reset, deconditioning the tape mark recognition circuits and disabling the First Character Tape Mark signal. The read operation concludes in the normal manner. For more detailed information on TAU Write Tape Mark Controls, refer to the *Tape Adapter Unit Manual* (Form 223-6847).

E. Start 1401 Clock (Figure 56): The Write Tape Mark signal enables Tape Clock which in turn enables Clock Control.

The Clock Control output develops Start Clock (31.10.11.2). With this Start Clock signal, clock control is returned to the 1401.

ILD 80 C4	Enable Tape Clock	U-Op Code Gate TAU Busy
ILD 80 D4	Enable Clock Control	Tape Clock

F. End Operation (Figure 56): The Write Tape Mark operation ends when the Write Tape Mark trigger in TAU has been reset (See the *Tape Adapter Unit Manual,* Form 223-6847).

The UNIT SELECT, REDUNDANCY, and TAPE UNIT latches are reset as described in the Backspace Tape operation, heading F.

REWIND TAPE

This instruction usually follows an end-of-reel condition, and causes the selected tape unit to rewind its tape. When a Rewind Tape Operation is initiated, the specified tape unit is disconnected from the system. When the tape reaches its load point, the tape unit ready light turns ON and the unit is again available to the 1401 program.
Example: U (% U3) R

Rewind the tape in Tape Unit 3.

OBJECTIVES (REWIND TAPE)

The following objectives are required to perform a Rewind Tape operation:

A. Set unit control status.
B. Select the desired tape unit.
C. Enable Rewind Call.
D. Develop TAU rewind controls.
E. Start 1401 clock.
F. End operation.

The instruction phase of a Rewind Tape operation is identical to that described for a Backspace Tape operation. The execution is the actual movement of tape itself, and is TAU controlled.

LOGIC DETAIL (REWIND TAPE)

Objectives A and B (Figure 57): These objectives follow the procedure and timing as described for a Backspace Tape operation, headings A and B.

C. Enable Rewind Call (Figure 57): Procedure here is identical to that described for Backspace Tape, except that the d-character contained in the A-register at I-Ring-5 time is B81 (R) rather than BA2 (B). The contents of the A-register are switched with the U-Op Code Gate to initiate Rewind and Rewind Call to TAU.

ILD 82 C4	Rewind, Call (to TAU)	U-Op Code Gate A-Register B81 (R)

D. Develop TAU Rewind Controls: For information on TAU Rewind Controls, refer to the *Tape Adapter Unit Manual* (Form 223-6847).

E. Start 1401 Clock (Figure 57): The Rewind signal is switched with a Select and Rewind Signal from TAU to enable Tape Clock, which in turn enables Clock Control. The Clock Control output develops Start Clock ILD 1. With this Start Clock Signal, Clock Control is returned to the 1401.

ILD 82 C4	Enable Tape Clock	Rewind Select and Rewind (TAU)
ILD 80 D4	Enable Clock Control	Tape Clock

F. End Operation (Figure 57): The Rewind Tape operation is completed when the tape reaches its load point. The UNIT SELECT, REDUNDANCY, and TAPE UNIT latches are reset as described in the Backspace Tape operation, heading F.

REWIND AND UNLOAD TAPE

This instruction operates in the same manner as Rewind Tape. However, in addition to rewinding the tape, it causes an unload condition to occur at the end

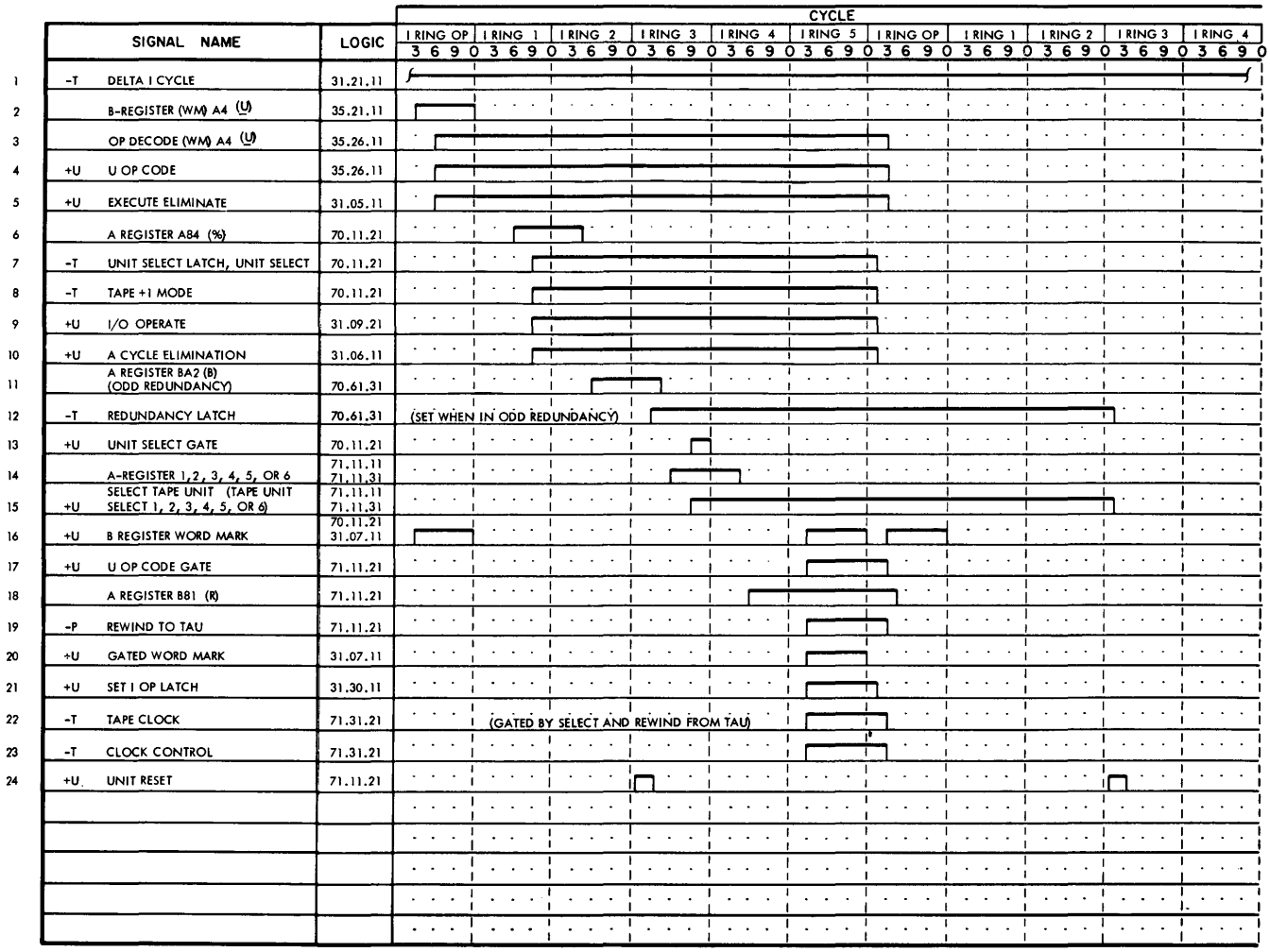


Figure 57. Timing for Rewind Tape Operation

of the rewind. At completion of the Rewind and Unload Tape operation, the specified tape unit is no longer available to the 1401 program.

Example: U (% U2) U

Rewind and unload the tape in Tape Unit 2.

OBJECTIVES (REWIND AND UNLOAD TAPE)

The following objectives are required to perform a Rewind and Unload Tape Operation:

A. Set unit control status.

B. Select the desired tape unit.

C. Enable Rewind Unload Call.

D. Develop TAU rewind unload controls.

E. Start 1401 clock.

F. End operation.

The instruction phase of a Rewind and Unload operation is identical to that described for a Backspace Tape operation. The execution phase is the actual movement of the tape itself, and is TAU controlled.

LOGIC DETAIL (REWIND AND UNLOAD TAPE)

Objectives A and B (Figure 58): These objectives follow the procedure and timing as described for a Backspace Tape operation, headings A and B.

C. Enable Rewind Unload Call (Figure 58): Procedure here is identical to that described for Backspace Tape, except that the d-character contained in the A-register at I-Ring-5 time is CA4 (U) rather than BA2 (B). The contents of the A-register are switched with U-Op Code Gate to initiate Rewind and Unload and Rewind Unload Call to TAU.

ILD 82 D4	Rewind and Unload (to TAU)	U-Op Code Gate A-Register CA4 (U)

D. Develop TAU Rewind Unload Controls: For information on TAU Rewind Unload Controls, refer to the *Tape Adapter Unit Manual* (Form 223-6847).

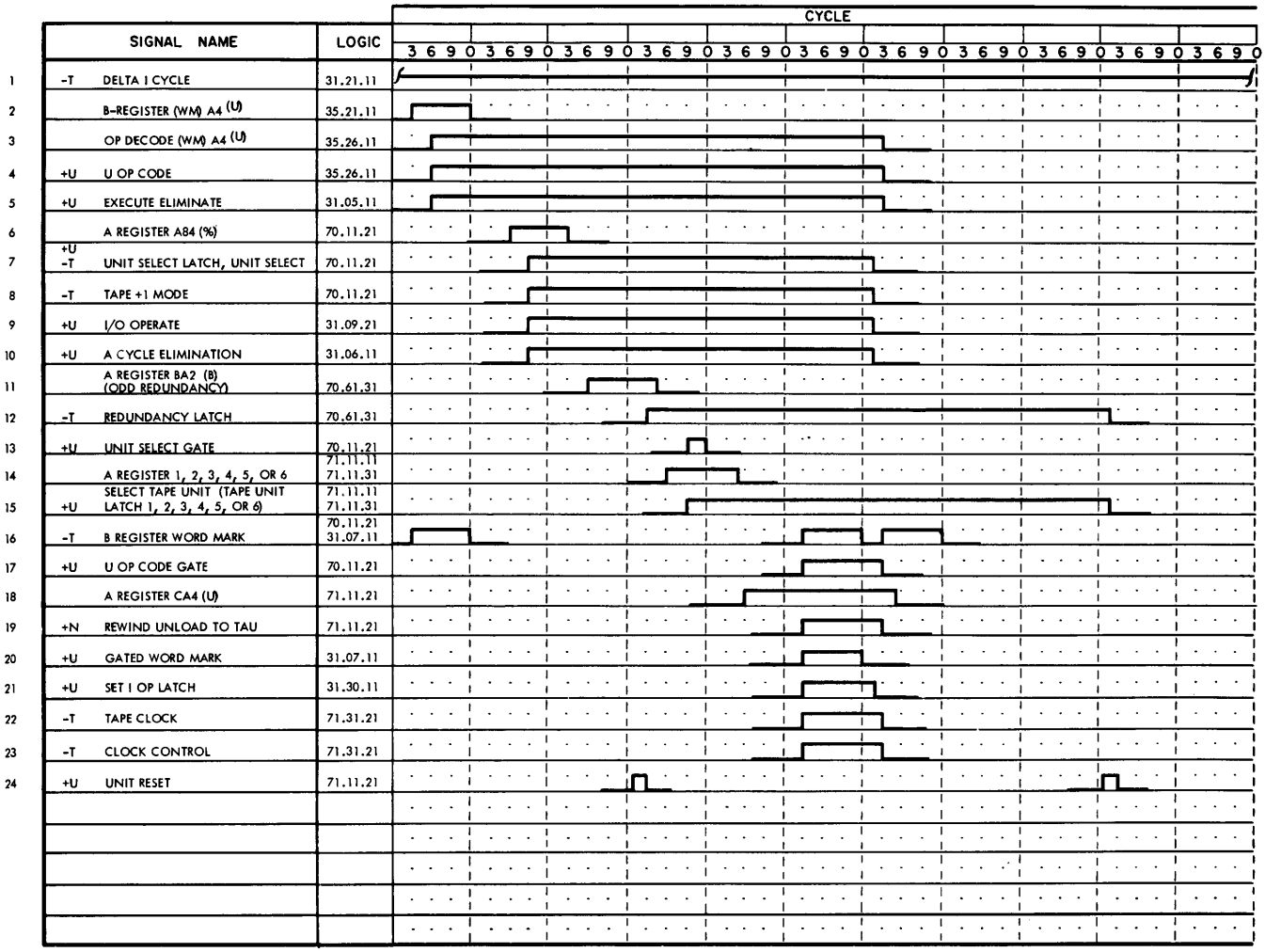


Figure 58. Timing for Rewind and Unload Tape Operation

E. Start 1401 Clock (Figure 58): The Rewind and Unload signal which is switched with a Select and Rewind signal from TAU to enable Tape Clock which in turn enables Clock Control. The Clock Control output develops Start Clock. With this Start Clock Signal, clock control is returned to the 1401.

ILD 82 D4	Enable Tape Clock	Rewind and Unload Select and Rewind (TAU)
ILD 80 D4	Enable Clock Control	Tape Clock

F. End Operation (Figure 58): The Rewind and Unload Tape operation is completed when the tape unit reaches load point and is unloaded.

The UNIT SELECT, REDUNDANCY, and TAPE UNIT latches are reset as described in the Backspace Tape operation, heading F.

Tape Test Instructions

Two tape instructions of the form B(AAA)d provide tests for end-of-file on tape and tape error transmission. The d-character determines which condition is to be tested, as follows:

d-character	*Test Condition*
K	End-of-file
L	Tape Transmission Error

If the condition tested-for exists, the 1401 program branches to the next instruction, which begins at the location specified by the (AAA) address. If the condition tested for does not exist, the 1401 program continues in normal sequence.

The End-of-File and Tape Transmission Error tests must be made immediately following a tape-read or tape-write operation to insure correct results. Otherwise, any new operation on any of the tape units resets the Tape Error Light on the 1401 console. Also, the End-of-File indicator cannot be tested if another tape unit is selected.

TAPE TRANSMISSION ERROR TEST

When an error occurs in transmission between a tape unit and the 1401 during a tape-read or tape-write operation, an error indicator in the 1401 turns ON, and a tape error light on the console lights. The B(AAA)L instruction tests the error indicator for a set condition. If the test is positive, the 1401 program branches to the location specified by the (AAA) address.

Example: B (399) L

Branch to storage location 399 if a Tape Transmission Error condition exists.

OBJECTIVES (TAPE TRANSMISSION ERROR TEST)

The following objectives are required to perform a Tape Transmission Error test:

A. Set Branch status.

B. Read-in the location of the next instruction.

C. Develop tape transmission error test controls.

D. End operation.

The instruction phase of the operation is six I-cycles long. During this phase, the 1-character Op code is transferred from storage to the Op register and remains there until the start of a new operation. The (AAA) address is transferred from core storage, passes through the B- and into the A-register, and enters the A-address register. The d-character is transferred from core storage to the B-register and then enters the A-register.

During the instruction phase of the next operation, storage is addressed from the A-address register at I-Op time, then with the modified A-address from the I-address register during succeeding I-cycles. Refer to the *1401 Data Processing System Manual, Logic Circuits,* for more detailed information on branch operations.

LOGIC DETAIL (TAPE TRANSMISSION ERROR TEST)

A. Set Branch Status (Figure 59): During I-Op time, A (WM) CBA2 character is read from core storage into the B- and Op-registers. The Op register contents are then decoded as B (branch) to set branch status.

ILD 15	Op Register (WM) CBA2 (B)	Time 060-090 I-Ring Op B-Register (WM) CBA2
ILD 83 B1	Branch Operation	Op Decode (WM) CBA2

B. Read-in the Location of the Next Instruction (Figure 59): During I-Ring-1 through I-Ring-3 time, the (AAA) address is developed. The address is transferred from storage to the A-address register serially by digit first the hundreds-thousands digit, next the tens digit, and finally the units digit.

ILD 17 C3	Gate B-Reg Contents into H-T Position of A-Addr Reg	Time 090-000 I-Ring-1 B-Register Contents
ILD 17 C5	Gate B-Reg Contents into Tens Position of A-Addr Reg	Time 090-000 I-Ring-2 B-Register Contents
ILD 17 C6	Gate B-Reg Contents into Units Position of A-Addr Reg	Time 090-000 I-Ring-3 B-Register Contents

C. Develop Tape Transmission Error Test Controls (Figure 59): At I-Ring-4 time, the d-character of the instruction (which indicates the type of test to be

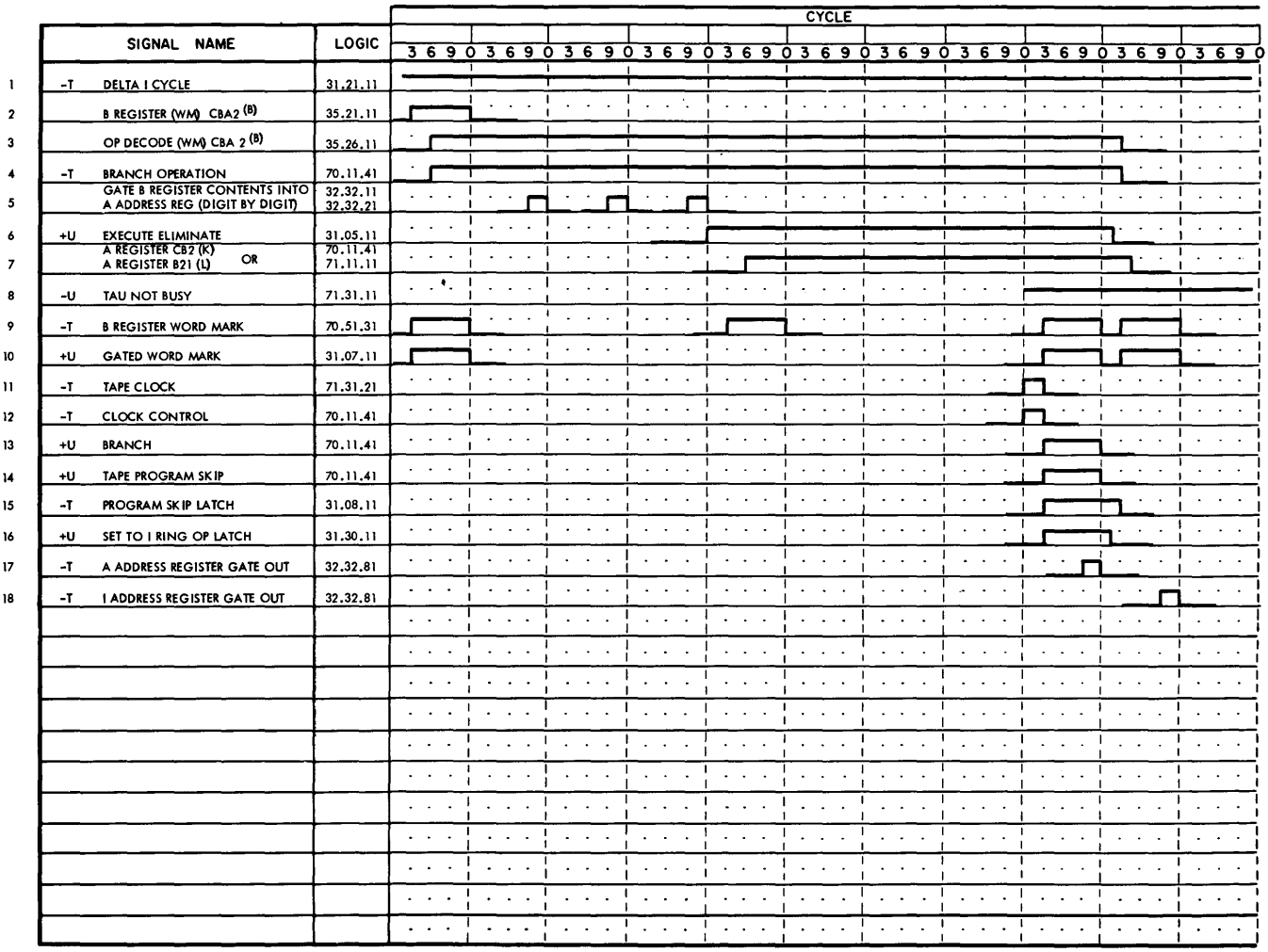


Figure 59. Timing for Tape Indicator and Tape Error Tests

made) enters the B-register and transfers to the A-register. Also, branch operation is switched with I-Ring-4 to develop an execute-eliminate signal which blocks an I-E phase change. Execute Eliminate remains enabled through I-Ring-5 time.

At I-Ring-5 time, the next Op code with its word-mark enters the B-register. Also, during I-Ring-5 time, the A-register contents are switched with Branch operation, Not Manual operation, TAU Busy. The resultant output generates a *Turn Off Delta Process* signal to stop the system. If no error exists, the program continues at the I-address when TAU Busy goes OFF.

ILD 13 D1	Execute Eliminate	I-Ring-4 and -5 Branch Operation
ILD 83 B2	Turn-Off Delta Process	Run A-Register B21 (L) Not Manual Op Branch Operation TAU Busy
ILD 83 B3	Delta Process On	Time 060-075 Run A-Register B21 (L) Not Manual Op Branch Operation Not TAU Busy Not Proc Chk Stop
ILD 83 C2	Branch	I-Ring-5 Gated Word-Mark Branch Opr
ILD 83 C2	Tape Program Skip	Branch A-Register B21 (1) TAU Error

D. End Operation (Figure 59): The Tape Program Skip signal turns on the PROGRAM SKIP and SET TO I-RING OP latches. The PROGRAM SKIP latch output develops an I-A Address Register Gate, while the SET TO I-RING OP latch output develops a Set to I-Ring Op signal, which gates on the I-Ring Op trigger. At time 090-000 of I-Ring-5, the A-address register contents are gated into

the storage address register which (1) reads out core storage, and, (2) transfers the address, digit-by-digit, through the address register modifier and into the I-address register.

At time 015-030, the I-Ring Op trigger is set. Then at 090-000 time, the contents of the I-address register are gated into the address register, and the normal program sequence for the new instruction begins.

ILD 14 B3-B4	Program Skip	Tape Program Skip
ILD 14 B4	Set to I-Ring Op	Program Skip
ILD 14 B4	Enable I-A Address Register Gate	Program Skip latch

END-OF-FILE INDICATOR TEST

When the 1401 senses a tape mark during a tape-read operation, or a reflective spot during a tape-write operation, the End-of-File indicator turns on. The End-of-File indicator is then tested with this instruction for a set condition. If the test is positive, the 1401 program branches to the location specified by the (AAA) address.

Example: B (501) K

Branch to storage location 501 if an End-of-File condition exists.

OBJECTIVES (END-OF-FILE INDICATOR TEST)

The following objectives are required to perform an End-of-File test operation.

A. Set Branch status.
B. Read-in the location of the next instruction.
C. Develop end-of-file test controls.
D. End operation.

The instruction phase of the operation is six I-cycles long. During this phase, the I-character Op code is transferred from storage to the Op register and remains there until the start of a new operation. The (AAA) address is transferred from core storage, passes through the B- and into the A-register, and enters the A-address register. The d-character transfers from core storage to the B-register, and then enters the A-register.

During the instruction phase of the next operation, storage is addressed from the A-address register at I-Op time, then with the modified A-address from the I-address register during succeeding I-cycles. Refer to the *1401 Data Processing System Manual, Logic Circuits*, for more detailed information on Branch operations.

LOGIC DETAILS (END-OF-FILE INDICATOR TEST)

A. Set Branch Status (Figure 59): During I-Op time, a (WM) CBA2 character is read from core storage into the B- and Op registers. The Op register contents are then decoded as B (branch) to set branch status.

ILD 15	Op Register (WM) CBA2 (B)	Time 060-090 I-Ring Op B-Register (WM) CBA2
ILD 83 B1	Branch Operation	Op Decode (WM) CBA2

B. Read-in the Location of the Next Instruction: During I-Ring-1 through I-Ring-3 time, the (AAA) address is developed. The address is transferred from storage to the A-address register serially by digit – first the hundreds-thousands digit, next the tens digit, and finally the units digit.

ILD 17 C3	Gate B-Reg Contents into H-T Position of A-Address Reg	Time 090-000 I-Ring-1 B-Register Contents
ILD 17 C5	Gate B-Reg Contents into Tens Position of A-Address Reg	Time 090-000 I-Ring-2 B-Register Contents
ILD 17 C6	Gate B-Reg Contents into Units Position of A-Address Reg	Time 090-000 I-Ring-3 B-Register Contents

C. Develop End-of-File Test Controls: At I-Ring-4 time, the d-character of the instruction, indicating the type of test, enters the B-register and is transferred to the A-register. Branch operation switches with I-Ring-4 and 5 to develop an execute-eliminate signal, which inhibits the normal I-E Change. The A-register K-character switches with the tape indicator signal at I-Ring-5. A positive result develops a TAPE PROGRAM SKIP, PROGRAM SKIP latch and I-A ADDRESS REGISTER GATE in sequence. A signal is sent to reset the tape indicator on the tape drive. The 1401 clock is stopped for one cycle to prevent advance of the program during the reset. The fall of the tape indicator signal allows the clock to restart, and the program to continue from the A-address. A negative test allows the program to continue without interruption from the next I-address.

ILD 13 D1	Execute Eliminate	I-Ring-4 and -5 Branch Operation
ILD 83 C2	End-of-File Test	Sel and TI on A-Reg A2 (K) Branch
ILD 83 C3	Tape Program Skip	End-of-File Test Time 045-075
ILD 83 C3-C5	Prevent Clock Control	End-of-File Test Time 090-030
ILD 83 C3	Turn Off TI	End-of-File Test Time 090-030

D. End Operation: At I-Ring-4 time, the d-character of the instruction (which indicates the type of test to be made) enters the B-register and is then transferred to the A-register. Also, Branch operation is switched with I-Ring-4 to develop an execute-eliminate signal which blocks an I-E phase change. Execute-eliminate remains enabled through I-Ring-5 time.

At I-Ring-5 time, the next Op code with its word-mark enters the B-register. The B-register word-mark enables Gated Word-Mark which in turn develops a Branch signal.

ILD 14 B3-B4	Program Skip	Tape Program Skip
ILD 14 B4	Set to I-Ring Op	Program Skip
ILD 14 B4	Enable I-A Address Register Gate	Program Skip latch

Error Conditions

Errors can arise from two areas in the tape configuration for the IBM 1401. One group includes the normal errors detected in the basic system. In the second group are those sensed in the TAU. With the exception of operation decode errors and address errors, the operation is completed before a stop. Errors in operation and address may cause destruction of good information. In the case of data transmission errors, an invalid character is available both at its source and as it was written.

Processing Unit Errors

Data transfer errors within the basic system are detected in the usual manner. When an invalid character is detected in a register or on the inhibit drive, a check latch is set for each area. These signals are mixed to produce a process-check signal to stop the system. As with card I/O operations the check signal is not gated until after the I-E Change. Then the complete record is transferred. If the error is in programming or addressing, the stop in the processing unit is immediate. The tape unit continues until the end of record on a read operation or the end of the tape reel on a write operation.

One new condition is introduced with a tape read error in TAU. Having detected an error in bit count in TAU, it is assumed that the A-register check is also set. To prevent a process-check, the TAU error signal is used to reset the A-register check. Characters leaving the A-register for the inhibit drive are always validated by adjusting the C-bit. Further transfer of the character always checks valid. Analysis of the stored information and the original tape indicates the type of failure. No provision is made on a tape write operation to prevent both system and TAU errors from being set.

TAU Errors

The Tape Adapter Unit checks character- and record-validity during both read and write operations. Detected errors set the TAU ERROR latch but do not stop the operation. In order that an error in tape operation may be evaluated, it is necessary to perform a test and branch operation B (III) L immediately following the tape operation. The TAU ERROR latch is reset at the start of the next tape operation.

Five check conditions are used to test for data transfer errors, and result in setting the TAU ERROR latch:

Vertical Redundancy Read-Write Register: A bit count is made on all characters passing through the register. This provides a check of entry on write operations and of exit on read operations. The bit count is checked either even or odd depending on the setting of the redundancy trigger (normally even for BCD and odd for binary).

Vertical Redundancy Read Register-A: A bit count is made on all characters passing through the register. This resgister serves both read operations and read-after-write operations. The bit count is checked either even or odd depending on the setting of the redundancy trigger.

Write Compare: During the read-after-write check, the two read registers are compared bit-by-bit for a matching condition. The vertical redundancy check on Skew Register-A serves to prove that both registers are valid, when the compare is equal.

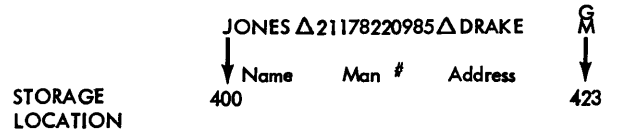
Write Echo: A check is made at the write amplifiers in the tape unit to determine that the write gating was performed. A trigger is set in TAU to indicate *no-echo*. A signal from any write driver resets the trigger to prevent the error signal.

Longitudinal Redundancy Check: A seven bit register of binary connected triggers is used to store the count of recorded bits. In each position the trigger sets ON for the first bit and OFF for the second bit. On a write operation the bits are entered from the read register after write check. The result after any number of characters should be even. After the last character of the record is written, the reset of the write triggers writes a check character. When the check character reads back on write check, it adds to the bit count, and normally resets all positions. During a read operation, all the characters of the record plus the check character are read into the check register, and should result in all triggers OFF. Any trigger left ON during either a read or write operation causes the TAU ERROR latch to set.

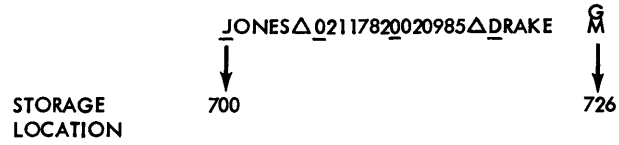
Expand Compressed Tape

This is an optional routine used to expand 7070 tape records written with zero elimination. Zeros are eliminated in the positions to the left of the last significant digit of numerical fields. Alphabetic fields are not compressed on 7070 tape, and therefore require no expansion.

COMPRESSED TAPE RECORD



EXPANDED RECORD



PROGRAM FOR EXPANDING THE COMPRESSED RECORD

STORAGE LOCATIONS	INSTRUCTION	OPERATION
001-008	M % C 1 400 R	Read Compressed Tape
009-012	H 034	Store B Address
* 013-019	M XXX 399	Set GM
* 020-026	⌐ 399 706	Set WM
* 027-030	⌐ 713	Set WM
031-037	X 423 726	Expand Operation

* These instructions may be performed in advance

TAPE DRIVE 1 ADDRESSED BY M % C 1 400 R INSTRUCTION

Figure 60. Example of Compressed Tape Expansion

Expanding a record requires the insertion of zeros in the high-order positions of numerical fields. This brings them to the desired word length for processing by the 1401 system.

The expand routine requires three optional feature instructions (Figure 60):

1. M (% CX) (XXX) R
2. H (AAA)
3. X (AAA) (BBB)

The first of these instructions is basically the same as other tape read instructions. The difference is the C-character of the A-address, which specifies tape written in the compressed mode. The record is transferred directly to storage locations specified by the B-address of the instruction by the use of B-cycles and +1-modification of the B-address. All numerical field signs and delta, (mode-change characters) are stored. As delta characters are read from tape the mode of the next field is signified. The mode is normally alpha. A delta character will cause a change to numerical mode signifying a numerical field. The next delta will occur when an alpha field is to follow. The mode of the last field of the tape record is retained for control of the actual expand operation. The delta characters stored are used to control the expansion of numerical fields of the stored record. End-of-record gap initiates a forced group-mark without word-mark, which is read into storage and causes an E-to-I Change. This read operation is not terminated by a group-mark with word-mark in storage.

The H (AAA) instruction designates: store the B-address register in the storage location selected by the A-address portion of the instruction. The purpose of this instruction, which immediately follows the read compressed tape instruction, is to store the last address of the tape record. The location contains a group-mark without word-mark, forced by the end-of-record gap on the tape. This address is stored in the A-address portion of the expand instruction.

The X (AAA) (BBB) instruction is the actual expand instruction. It is a move with zero insertion operation. The move portion is similar to all other move operations, except that zeros are inserted in numerical fields, under control of the mode change and sign of the numerical field.

Read Compressed Tape—M (%CX) (XXX) R (Figures 61 and 62)

The object of this instruction is to select the desired tape unit, and read from the tape into storage. The

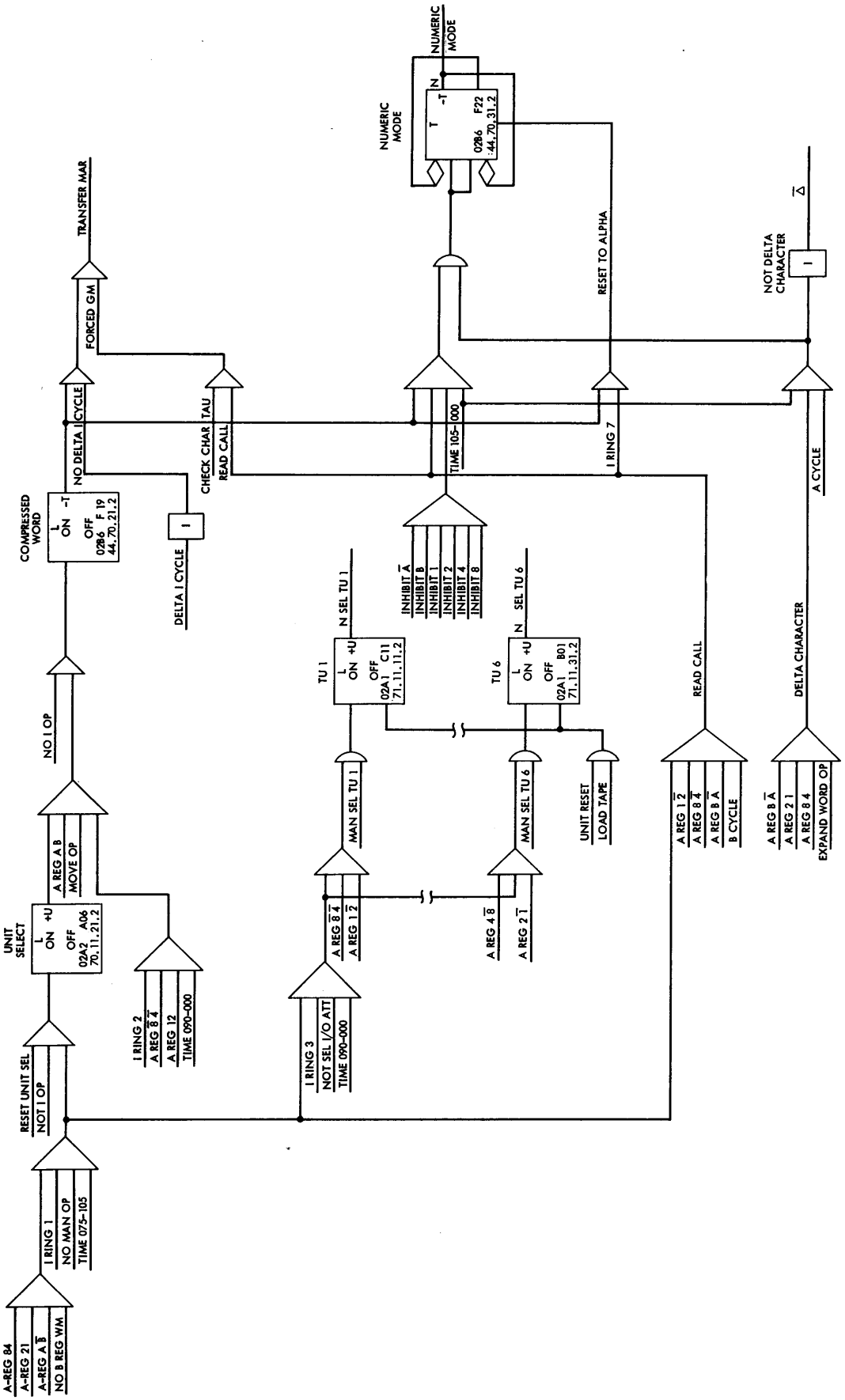


Figure 61. Read Compressed Tape Load M(%CX)(BBB)R

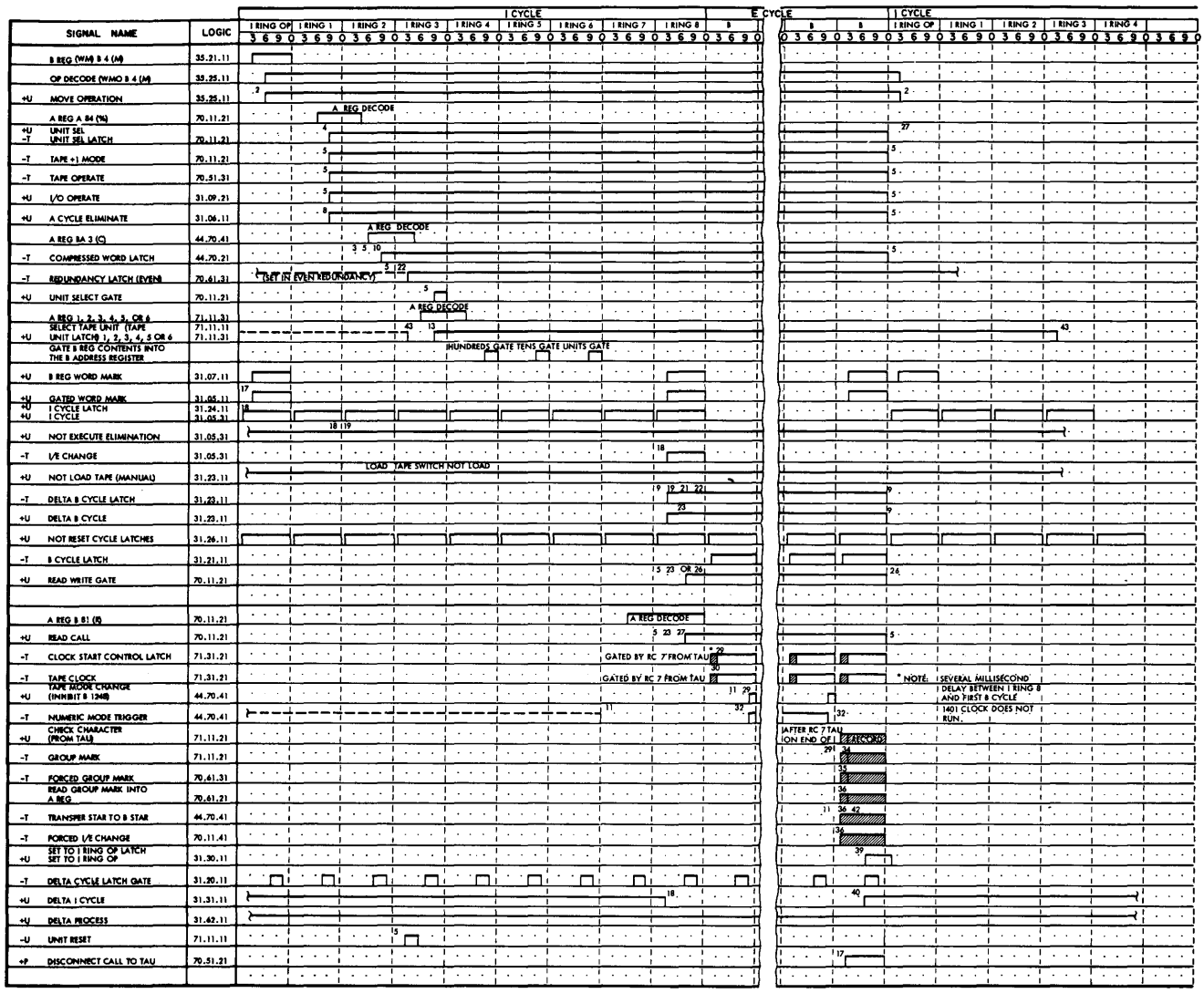


Figure 62. Move Compressed Tape Read Operation

tape unit selection is determined by the units position of the A-address portion of the instruction. The storage location to receive the tape record is specified by the B-address.

The operation is accomplished as follows:

1. At I-Op time the M Op code is transferred to the Op reg and sets the MOVE OP latch.
2. At I-1 time the % sign is transferred to the A-reg and sets up tape unit selection.
3. At I-2 time the C is transferred to the A-reg and turns on the COMPRESSED WORD latch.
4. At I-3 time the address of the tape drive is transferred to the A-reg, which brings up the unit select gate and TAPE UNIT latch of the tape unit addressed.
5. At I-4, I-5, and I-6 time the B-address is transferred to the B-STAR.
6. At I-7 time the R-modifier is transferred to the A-reg. This turns ON the READ CALL latch. The alpha-merical trigger is reset to alpha.
7. At I-8 time the word-mark of the next OP is sensed in the B-reg and the B-STAR is transferred to the STAR. An I-to-E change also occurs.

During the E-phase, the tape record is read into storage starting with the location specified by the B-address of the instruction. Read-in is accomplished with B-cycles only and +1-address modification.

Each time a delta character is sensed in the B-reg the status of the alpha-numerical trigger changes from alpha mode to numerical or numerical to alpha.

When an inter-record gap is sensed, a group-mark without word-mark is forced into storage. Forced group-mark initiates a transfer of the STAR to the B-STAR, and an E-to-I change.

Store B-Address Register—H (AAA)

The purpose of this instruction is to store the address in the B-STAR, which was transferred from the STAR on the read compressed tape operation.

A description of this operation is available in section under *Store B-Star Operation.*

Before the expand operation is undertaken, all field defining word-marks must be set. Also, a group-mark with a word-mark must be set in storage to the left of the high-order position of the compressed tape record. This group-mark with word-mark will be used to control the end of the expand operation.

Move with Zero Insertion — X(AAA)(BBB) Figures 63 and 64

The purpose of this instruction is to move the tape record stored on the tape read operation to the B-address. Zeros are inserted to the left of the last significant digit in numerical fields up to the defining word-mark.

Zero inserts are controlled by the first and second B-bit triggers when the alpha-numerical trigger is set in the numerical mode.

The *Move with Zero Insert* operation is accomplished as follows:

1. At I-Op time the X Op code is transferred to the Op reg. This initiates the expand operation and resets the first and second B-bit triggers OFF.
2. At I-1, I-2, and I-3 time the A-address is transferred to the B-STAR.
3. At I-4, I-5, and I-6 time the B-address is transferred to the B-STAR.
4. At I-7 time, the word-mark of the next Op code is sensed in the B-reg and initiates an I-to-E change.

FIRST A-CYCLE OF E-PHASE

1. The A-address is gated into the STAR.
2. The A-field group-mark is read into the A- and B-registers
3. The A-register group-mark prevents triggering the first or second B-bit triggers.
4. The A-field group-mark is read back into storage.
5. The A-address, in the STAR, is modified by a −1 and gated back into the A-STAR.

FIRST B-CYCLE OF E-PHASE

1. The B-address is gated into the STAR.
2. The B-field character is read into the B-reg.
3. The group-mark in the A-reg is transferred into the B-field storage location.

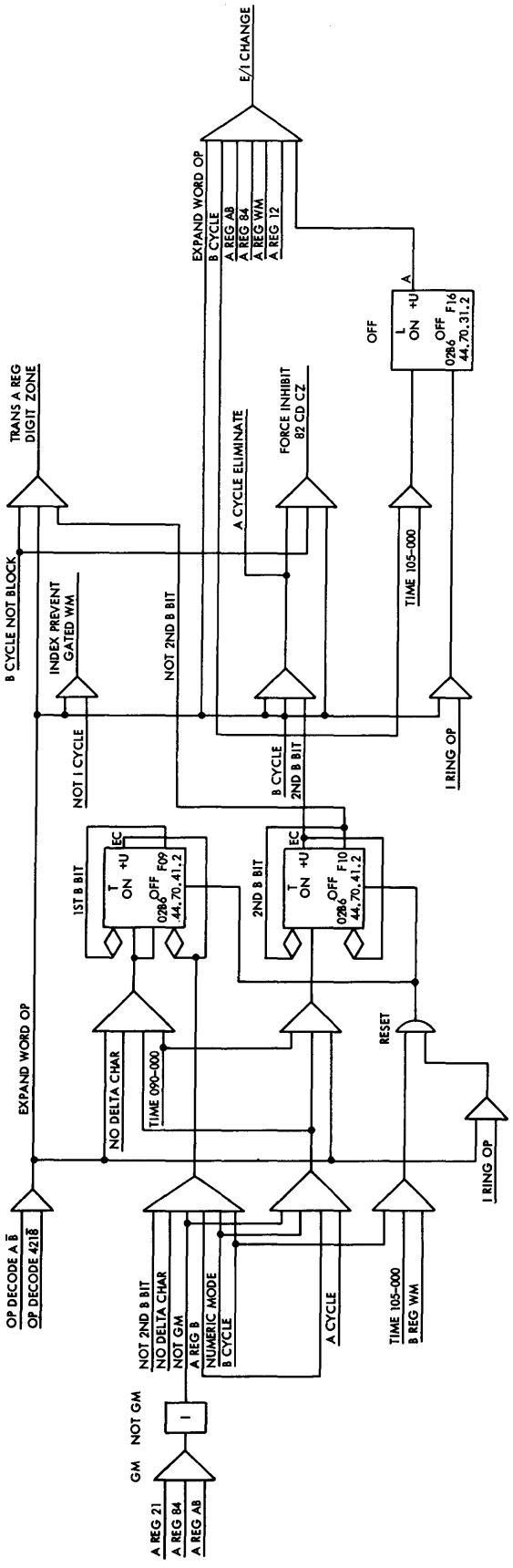


Figure 63. Expand Operation Logic X(AAA)(BBB)

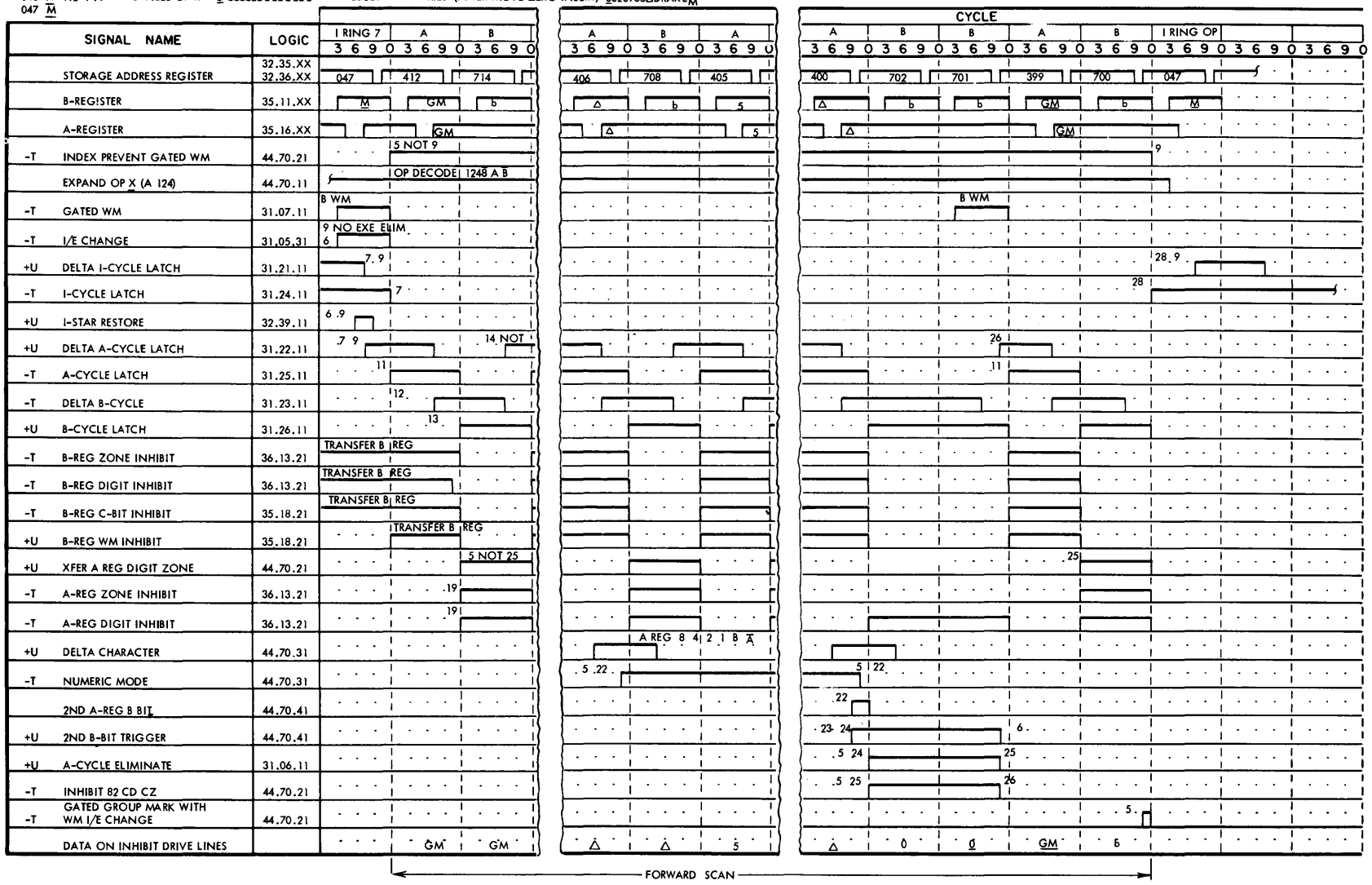
Figure 64. Move with Zero Insertion

399 GM
040 X 412 714
047 M

A-FIELD DATA GM Δ20985*ΔDRAKEGM
B-FIELD DATA b bbbbbbbbbbbbb

RESULTANT A-FIELD (AFTER MOVE ZERO INSERT) Δ20985ΔDRAKEGM
RESULTANT B-FIELD (AFTER MOVE ZERO INSERT) 0020985ΔDRAKEGM

		SIGNAL NAME	LOGIC
1		STORAGE ADDRESS REGISTER	32.35.XX 32.36.XX
2		B-REGISTER	35.11.XX
3		A-REGISTER	35.16.XX
4	-T	INDEX PREVENT GATED WM	44.70.21
5		EXPAND OP X (A 124)	44.70.11
6	-T	GATED WM	31.07.11
7	-T	I/E CHANGE	31.05.31
8	+U	DELTA I-CYCLE LATCH	31.21.11
9	-T	I-CYCLE LATCH	31.24.11
10	+U	I-STAR RESTORE	32.39.11
11	+U	DELTA A-CYCLE LATCH	31.22.11
12	-T	A-CYCLE LATCH	31.25.11
13	-T	DELTA B-CYCLE	31.23.11
14	+U	B-CYCLE LATCH	31.26.11
15	-T	B-REG ZONE INHIBIT	36.13.21
16	-T	B-REG DIGIT INHIBIT	36.13.21
17	-T	B-REG C-BIT INHIBIT	35.18.21
18	+U	B-REG WM INHIBIT	35.18.21
19	+U	XFER A REG DIGIT ZONE	44.70.21
20	-T	A-REG ZONE INHIBIT	36.13.21
21	-T	A-REG DIGIT INHIBIT	36.13.21
22	+U	DELTA CHARACTER	44.70.31
23	-T	NUMERIC MODE	44.70.31
24		2ND A-REG B BIT	44.70.41
25	+U	2ND B-BIT TRIGGER	44.70.41
26	+U	A-CYCLE ELIMINATE	31.06.11
27	-T	INHIBIT 82 CD CZ	44.70.21
28	-T	GATED GROUP MARK WITH WM I/E CHANGE	44.70.21
29		DATA ON INHIBIT DRIVE LINES	



4. The B-field address is modified by a −7 and gated back into the B-STAR.

This process is repeated. If the next A-field character is the units position of a numerical field, designated by the alpha-numerical trigger being in numerical mode, the B-bit in the sign turns ON the first B-bit trigger. The move operation continues until a second B-bit is sensed. This can be from one of two sources. It is either from 1) a delta character, or 2) the sign of the next numerical field.

The second B-bit turns on the second B-bit trigger. When a delta character is sensed, the status of the alphamerical trigger is reversed also. When the second B-bit trigger goes on, A-cycles are eliminated, and zeros are forced into the B-field storage locations. When the B-field word-mark is read into the B-reg, A-cycles are restored and the B-bit triggers are reset. When the second B-bit is in the sign of a numerical field, the first B-bit trigger is turned back ON during the next A-cycle.

When the mode change to alpha occurs, the operation continues as a normal move until another delta character is sensed. Then the operation is again altered by B-bit control to cause zero insertion.

When the group-mark with word-mark is read into the A-reg and sensed on a B-cycle, the EXPAND OFF latch is turned ON ending the operation and initiating an E-to-I change.

Manual Tape Controls

Operator's Console

One indicator lamp and three switches are added to the operator's console for tape control. These additions allow the necessary controls for tape loading and checking-out error conditions. The tape unit select control serves to activate the CE manual controls.

Tape: This lamp is lighted when the TAU error trigger is set. It is extinguished automatically when the next tape instruction is sent to the TAU.

Tape Unit Select Switch: This is a rotary switch used for manual selection of tape units. A normal (N), a diagnostic (D), and six select positions are provided. In the normal position the control is effectively out of the circuit. With switch set to the D-position, the operation is similar to the N-position with the following exceptions:

1. A tape-read error is placed in storage exactly as read from tape. No correction is made for validity.
2. In writing, with the error-stop switch ON, the write error *freezes* in the Read-A and Read-B registers.

   If the Read/Write register is in error, it will also *freeze.*

In any select position the unit-select and tape-branch circuits cannot be set by program instructions. The manual control switches on the CE panel are activated in any select position. The program can proceed on non-tape operations, while CE operations are being performed on tape.

Load Tape: This is a key type switch used to read-in one record from Tape Unit-1 without a programmed instruction. The system must be stopped before the key is used. The Load instruction is initiated and the system is started in an execute cycle (B-cycle). The record is stored starting with address 001 and continued until an inter-record gap is sensed on the tape. At the end of the record, the 001 address is again forced as the start of the next instruction.

Backspace: This is a key type switch used to backspace the selected tape unit by one record. The Tape Unit Select switch must be set to select a tape unit. It serves in manual evaluation or correction of tape records.

CE Panel

A CE test panel for the tape area is located in 02A1 of the 1401. Lamp indicators and control switches allow forced operation and evaluation of the Tape Adapter Unit (TAU). Either of two CE test panels may be used depending on the TAU model installed. TAU-2 is used in earlier systems using only the 729 Tape Drives. TAU-9 used in the later systems has, in addition, facilities for the 7330 Tape Drive. The basic concept of the two units is the same. Their operation is detailed in the *TAU Instruction Manual,* Form 223-6847. The two panels are discussed together, with their differences noted (Figures 65 and 66).

INDICATOR LAMPS

The upper portion of the CE panel provides lamp indication of the status of various TAU triggers. A lighted lamp indicates the associated trigger is in the ON status. The lamps function during normal operation, but many of them do not glow because of very short pulsing. Their major service value comes from being left ON when the system stops under error conditions. The lamps are divided into related groups as follows:

Read-Write Control Indicators: Eight lamps show the status of read and write clock control triggers.

READ—Indicates the basic read operation trigger.
WRITE—Indicates the basic write operation trigger.
READ DELAY—Indicates the starting delays for a read operation.
WRITE DELAY—Indicates the starting delays for a write operation.
READ CONDITION—Indicates actual read operation.

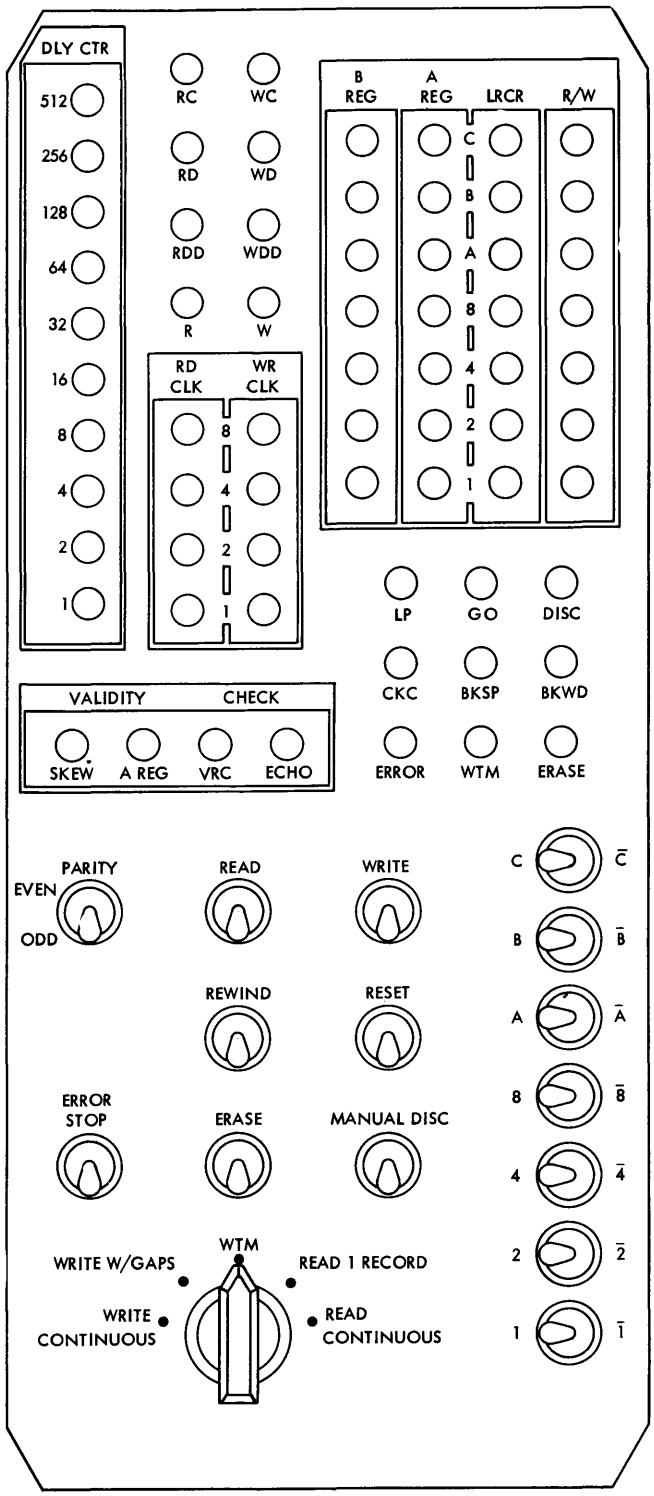


Figure 65. TAU-2 Tape CE Panel

WRITE CONDITION—Indicates actual write operation.

READ DISCONNECT DELAY—Indicates the stopping delay for a read operation.

WRITE DISCONNECT DELAY—Indicates the stopping delay for a write operation.

These indicators plus the clock or delay counter indication give the point in the cycle at which the hang-up occurred.

Read-Write Clocking: Four lamps on each clock show the status of the clock in binary notation.

READ CLOCK—Indicates the position of the clock used for read operations (also used for read-after-write).

WRITE CLOCK—Indicates the position of the clock used for write operations.

Delay Counter Indicators: Lamps show each trigger in the counter ring, which progresses in binary notation. TAU-2 panel has ten lamps (1 to 512), while TAU-9 panel has twelve lamps (1 to 2048).

Register Bit Indicators: Seven lamps show the bit status in BCD code for each register .

B-REGISTER (READ)—Indicates the bits in the register during operation. Normally it is blank, when stopped. With the error-stop switch ON, the bits *freeze* at the error condition on stop.

A-REGISTER (READ)—Indicates the bits in the register the same as B-Register above.

LONGITUDINAL REDUNDANCY CHECK REGISTER—Indicates the bits left in check register at the end of the operation. The check character normally returns all triggers to the OFF status.

READ/WRITE REGISTER—Indicates bits in the register. The register normally retains the bit combination of the last character written when stopped. With a CE error stop condition, the register freezes with the bit combination at the time of error.

Function and Status Indicators: In general this group of lamps reflects operations other than read or write. It also includes several status gates to show the progression of the cycle.

BACKSPACE—Indicates the backspace operation trigger is set to back the tape over a record.

BACKWARD—Indicates the backward motion trigger is set for reverse operation of the tape drive.

GO—Indicates the tape movement trigger is set for either forward or backward motion.

LOAD POINT—Indicates the selected tape drive has not been advanced since being placed in Load status. The tape starts to move a short distance before starting the read or write operation to insure clean tape.

DISCONNECT—Indicates the basic operation is completed and the controls are cycling for a stop.

REWIND (TAU-9 only)—Indicates the tape is in a rewind operation. The drive is left in Load point status on completion. For the 7330 this is a slow-speed rewind.

REWIND & UNLOAD (TAU-9 only)—Indicates the tape is in a rewind operation. The drive is left in the unloaded status on completion. For the 7330 this is a high-speed rewind.

FIRST CHARACTER (TAU-9 only)—Indicates the first read character of either a read-only or read-after-write operation.

NO ECHO (TAU-9 only)—Indicates the trigger has not been reset as the result of the write echo from the tape drive.

FORWARD STOP DELAY (TAU-9 only)—Indicates the trigger is set for the 7330 stop delay controls which differ from those for 729.

WRITE TAPE MARK—Indicates a write tape mark operation is in progress.

ERASE—Indicates an erase call has been made. Lamp stays lit until the end of the following write operation.

CHECK CHARACTER—Indicates check character read time on a disconnect cycle.

ODD REDUNDANCY (TAU-9 only)—Indicates the trigger is set for odd redundancy call as the result of either operation control or CE panel parity switch.

WRITE RELEASE (TAU-9 only)—Indicates the WRITE TRIGGER RELEASE latch is set ON to allow functioning of the write triggers in the tape drive.

ERROR (TAU-2 only)—Indicates one of the check circuits has detected an error and set the TAU ERROR latch. This lamp is located in the check area of TAU-9.

Check Indicators: These lamps indicate errors detected by the validity-check circuits in TAU.

ERROR (TAU-9 only)—Indicates one or more of the errors have been detected to set the TAU ERROR. The lamp is included as a status indicator in TAU-2.

A-REGISTER (READ)—Indicates a vertical redundancy check of the register bits.

READ/WRITE VERTICAL REDUNDANCY CHECK—Indicates a redundancy check of the register bits.

COMPARE (TAU-9 only)—Indicates a difference in bits read into the A- and B-registers.

SKEW—Indicates a bit or bits sensed after the read sample gate. The bits are angled or skewed on the tape to an angle greater than permissible, or the characters are written too closely together *(packed)*.

CONTROL SWITCHES

The lower portion of the CE panel provides switches to force various operation calls to TAU. These switches, except Stop-on-Error, are not effective unless the console Tape Drive Select switch is set to a drive.

Early Sample (TAU-9 only): This is a toggle-action switch, which causes the read-sample pulse to occur one clock-pulse earlier. This shortens the time allowed to read before the skew gate is set.

Amplifier Bias (TAU-9 only): This is a toggle-action switch, which reverses the read-only and read-after-write clipping levels. The clipping levels for read-after-write are tighter to insure good records. Use of these levels for read-only aids in detecting reading problems.

Error Stop: This is a toggle-action switch used to set a stop-on-error condition for CE tape operation.

With TAU-2 it is effective only on write operations. Any error condition setting the TAU error causes the read registers A and B to *freeze* with the existing bit combination. The R/W register does not freeze.

With TAU-9 the switch is effective on both read and write operations. On a write operation the R/W freezes on an R/W validity check. The read registers A

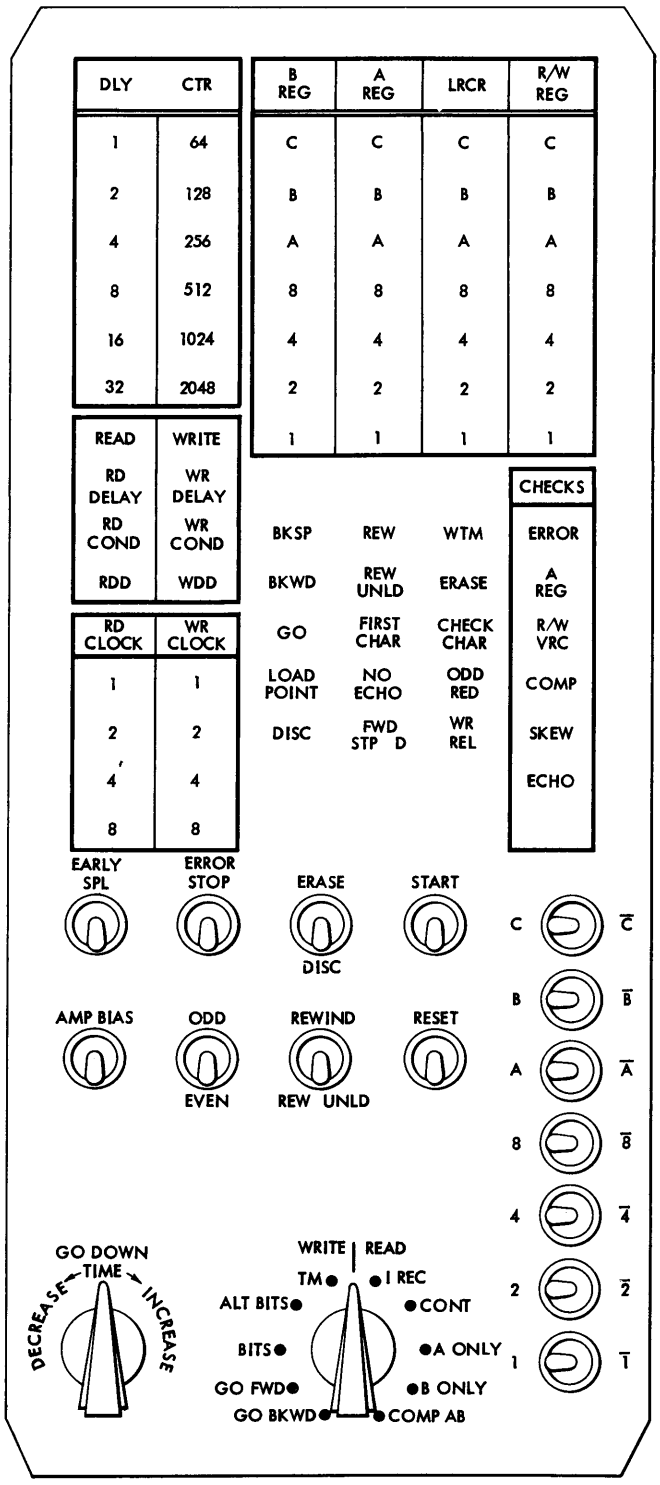


Figure 66. TAU-9 Tape CE Panel

and B *freeze* with any error condition. An analysis of the register bits determines which type of error was sensed. On a read operation the read register errors do not set the TAU error. An error detected in the R/W Register *freezes* all registers. The character in the read registers is the following character from tape.

Parity Odd-Even: This is a toggle-action switch, which sets the redundancy call to TAU for manual operations.

Erase: This is a momentary type switch, which causes an erase call to TAU. The erase trigger is set in the usual manner, but no operation is performed until the write call following is initiated. This switch on the TAU-9 panel shares a center-off type switch.

Manual Disconnect: This is a momentary type switch, which causes a TAU disconnect on a write operation and blocks the continuous restart controls for other operations. This switch on the TAU-9 panel shares a center-off type switch.

Rewind: This switch is of the momentary action type. When operated, it causes the selected tape drive to start a rewind operation. The operation ends with the tape in Load point status. For the 7330 it causes a slow speed rewind. This switch on the TAU-9 panel shares a center-off type switch.

Rewind and Unload (TAU-9 only): This operation shares a center-off momentary type switch. When operated, it causes the selected tape drive to start a rewind operation. The operation ends with the tape in the unloaded status. For the 7330 it causes a high speed rewind.

Read (TAU-2 only): This is a momentary type switch used to start a read-only operation. The function control switch must be set to a read operation.

Write (TAU-2 only): This is a momentary type switch used to start a write operation. The function control switch must be set to a write operation.

Start (TAU-9 only): This is a momentary type switch used to start various tape operations. Selection of the operation is determined by the setting of the function switch.

Reset: This is a momentary type switch used to force a TAU reset. The error check latches, the function latches, the register latches, and the clocking triggers are reset. Use during an operation causes immediate stop of the operation.

Bit Switches: Seven toggle-action switches provide selection of the bits to be written on a write operation. The selection and the parity setting must produce a valid combination. The same character is written in every position. TAU-9 has an alternate character control described under the *TAU-9 Function Switch—Alternate Bits.*

Function Switch: A dial switch is used to provide various read and write operating conditions. The switch layout and features differ between TAU-2 and TAU-9. They are described separately to prevent confusion.

TAU-2 Function Switch:

A. Write Continuous—causes the tape unit to write a continuous record of the preset character until manually stopped, or the tape reflective strip is sensed.

B. Write with Gaps—causes the tape unit to write multiple records of the preset character. The length of the individual record is controlled by pulses from the write delay control. MANUAL DISCONNECT of the tape reflective strip can stop the operation only at the end of a record.

C. Write Tape Mark—causes the tape unit to write a tape mark character and stop, when the write switch is operated.

D. Read 1-Record—causes the tape unit to read one tape record and stop at the inter-record gap. MANUAL DISCONNECT has no effect if operated.

E. Read Continuous—causes the tape unit to read record-by-record with automatic restart between records. The operation stops at the end of a record with MANUAL DISCONNECT or at a tape indicator.

TAU-9 Function Switch:

A. Write Tape Mark—causes the tape unit to write a tape mark character and stop, when the start switch is operated.

B. Write Alternate Bits—causes the tape unit to write a record alternating between the preset character and one with all-bits. Because all-bits is an odd-bit count, the parity and preset character must be set odd. This operation may write a continuous record or multiple records with gaps as described under *TAU-2 Logic Control.*

C. Write Bits—causes the same type of operation as alternate bits described above. The record is composed entirely of the preset character and may be of either odd- or even-parity.

D. Go Forward — causes forward tape movement without either read or write controls. The 1401 controls cause intermittent movement to check clutch operation. This operation causes tape erasure if it follows a write operation.

E. Go Backward—causes backward tape movement without other tape controls. The 1401 controls cause intermittent movement to check clutch operation. The control is described under *TAU-2 Logic Control.*

F. Read 1-Record — causes the selected tape drive to read a single record. Stop is automatic at the inter-record gap. MANUAL DISCONNECT has no effect on the operation.

G. Read Continuous — causes the selected tape drive to read consecutive records until stopped. The TAU controls cause a stop at each inter-record gap, but the 1401 controls initiate another start. The operation terminates at an inter-record gap with either a MANUAL DISCONNECT or a tape indicator.

H. Read A-Only — causes a read-continuous operation as above. The A-register is always transferred to the read/write register and the B-register is ignored. This allows observation of the A-register value on invalid characters.

I. Read B-Only—causes a read-continuous operation as above. The B-register is always transferred to the read/write register and the A-register is ignored. This allows checking the B-register for invalid characters, when the A-register is correct.

J. Read-Compare AB — causes a read-continuous operation as above. The A- and B-registers are compared in the same manner as for read-after-write. This allows effectively checking the validity of both registers.

Go Down Time Control (TAU-9 only): This is a variable resistor control, which is part of the timing control for the inter-operation delay single-shot. Its normal use is with the go-forward and go-backward operations.

TAU-2 LOGIC CONTROL (ILD 85)

The logic controls for TAU-2 deal with the continuous restart operation. Operating either the read or write switches with a corresponding setting of the function switch starts the timing of a ten millisecond single-shot. With the fall of the single-shot timing, a second shingle-shot starts timing a two millisecond pulse. This pulse serves as the timing for TAU call signals. The pulse is gated with the operation indicated by the function switch to produce a specific call. A MANUAL INTERLOCK latch is set preventing a second switch control.

On a continuous type operation, the TAU call must be repeated after each disconnect. The TAU sends a check-character signal soon after the start of the disconnect cycle. This signal is used to start the same ten millisecond single-shot previously used by the start circuit. The delay serves to allow completion of the disconnect cycle in the TAU. With the fall of the single-shot, the two millisecond timing is again started for a new TAU call signal.

The operation continues until either a tape-indicator or a manual-disconnect signal occurs to block the new TAU call. Single record read operations stop after one cycle because the check character is blocked from restarting the single-shot. A write-continuous operation is stopped with a TAPE INDICATOR or MANUAL DISCONNECT forcing a single-shot start. The output of the single-shot, gated by the starting pulses, develops a write manual disconnect signal to TAU.

A variable timing is provided for the length of each record of a write-with-gaps operation. Three write-delay timings from TAU are terminated at 02A1-A26. A26A has a timing of WD 52, A26B of WD80, and A26C of WD 768 for forcing a write disconnect. The selection is made by jumper on the terminal block.

TAU-9 LOGIC CONTROL (FIGURE 67)

Continuous Restart Control: Operating the start switch with the function switch in a read or write position initiates a TAU call signal. The initial start does not use the single-shot control for the call signal. The rise of TAU BUSY sets the MANUAL INTERLOCK latch to

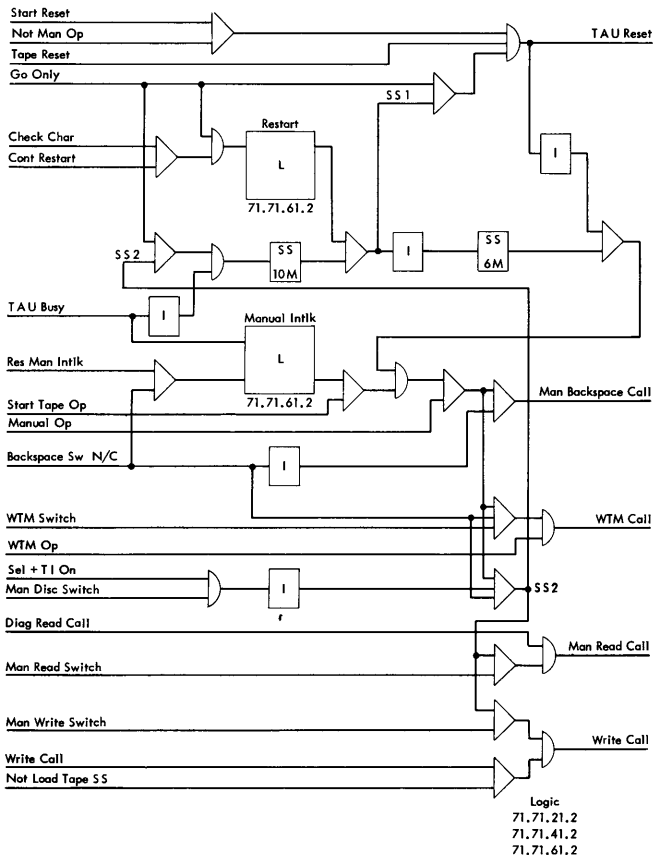


Figure 67. CE Panel Control Logic (TAU-9)

prevent a second control. The fall of TAU BUSY at the end of the operation starts a ten millisecond single-shot for disconnect delay.

The check-character signal from TAU switches with continuous operation to set the single-shot START latch. The latch output and the output of the delay single-shot provide the negative swing necessary to reset the six millisecond single-shot. When the delay single-shot falls, the six millisecond single-shot starts timing. The output provides the TAU call signal for the next operation. The START latch is reset with the start of the timing (6 millesecond) to prevent a second timing until the next check-character occurs. A continuous operation stops if either the check-character fails or the TAU remains busy.

Write Disconnect Control: The write operations are continuous restart of indefinite length record. A variable timing is provided to control the length of record. Three write-delay timings from TAU are terminated at 02A1-A26. A26B provides a timing of WD 52, A26C of WD 80, and A26D or WD 768. A26A provides a voltage level for continuous operation, while A26J is an open pin causing immediate disconnect. Selection

is made by jumper on the terminal block. A full disconnect is effected by TAPE INDICATOR or MANUAL DISCONNECT.

Go Operation Control: The first operation is initiated with the start switch to produce a *manual-go* signal to TAU. With the function switch set for *forward* operation, this is the only signal required to start movement in the selected tape drive. It should be observed that if the tape drive is in write status, the tape is erased with forward operation. For backward operation, the function switch adds the backward signal to TAU to shift the tape drive clutch. The backward signal always turns the tape drive to read status.

The initial operation length is determined by the time the start switch is held. When the switch is released, the fall of the manual-go signal starts the ten millisecond single-shot. The exact timing of the single-shot is variable below this timing with the go-down-time control. This is used to evaluate the mechanical start and stop delays of the tape drive. The use of the single-shot output forces a TAU RESET to drop the GO and BACKWARD latches to stop the tape movement.

The fall of the ten millisecond single-shot starts the six millisecond single-shot for the next movement. For the GO operations the six millisecond timing is changed to sixty milliseconds by switching in the ten microfarad capacitor. The shorter period used for read/write call is too short for the go-signal. The operation is stopped by a TAPE INDICATOR OR MANUAL DISCONNECT blocking a restart.

General Information

The Serial Input/Output (I/O) Attachment allows connection of one external device to the IBM 1401 system. The device can be a data transmission control, an input/output converter, or another data processing system. The data handling speed can vary from very low speeds up to speeds approaching the 1401 character rate (11.5 μs/char). At lower speeds the operation is normally controlled on a character-by-character basis. At higher speeds the control is by record.

Because of the variation in operation from one attachment to another, the exact system of operation cannot be detailed. Signals and interlocks are developed for any external condition. These lines are taken to the I/O connector, where they are cross-connected within the attachment plug (inter-face plugging) to activate the features desired. Operation controls and data transfer are similar to those for a tape operation. Additional controls are provided for communication between the attachment and the system. The I/O adapter contains registers, clocking, and controls that function similar to the Tape Adapter Unit (TAU).

The serial I/O feature shares the unit control circuits with the tape feature. The controls are switched to the I/O feature by a character change in the instruction. The unit select, read, and write circuits are set in the normal manner. Only one feature can be used at a time. The controls can be alternated, however, to service the normal complement of tape units.

Machine Language

Data is transferred to and from the I/O Adapter in Binary Coded Decimal (BCD) notation. The redundancy check in the I/O adapter may be either odd or even under control of the interface plugging. The redundancy checks within the 1401 remain odd-bit count. Word-marks are only considered on a Load operation. Conversion is normally made from the system word-mark to the word separator for the I/O adapter. Provisions are also made, through interface plugging, to enter word-marks directly on a load operation.

Transfer is parallel by bit, serial by character. Either forward or reverse scan is available by interface plugging. If other codes or transmitting systems are required by the external device, the I/O adapter provides the translation and buffering.

Data Flow

The data flow can only be positively described within the 1401 processing unit. For a transmission (write) operation, the flow is from core storage to the B-register with normal regeneration. From the B-register data is read through the tape I/O control to adjust the redundancy check if required. The data is then gated out through the I/O attachment connector to the buffering register in the I/O adapter.

The data is picked up for a receive (read) operation at the I/O adapter buffer register and sent to the system through the I/O attachment connector. The data passes through the tape I/O controls, where the redundancy check is adjusted if required, to set the A-register. During the next B-cycle, the view data in the A-register enters core storage to replace existing data. B-register word-marks are retained for a Move operation and dropped for a Load operation.

Serial Input/Output Instructions

Five operation codes are used to control the I/O attachment. The basic Move (M) and Load (L) operations are used to control read and write in the same manner as for tapes. The Unit Control (U) code is used to provide two major control signals for the I/O attachment. The Select Stacker (K) code is used to send secondary controls or conditions to the I/O attachment, while the operation is in progress. Additional modifier characters are used with the Branch (B) code to allow testing for conditions and signals sent to the system by the I/O attachment.

M(% x 1) (BBB) d — MOVE I/O ATTACHMENT

The M (Move) operation is used to send or receive data to/from the I/O atachment without reference to word-marks. The B-field word-marks are retained in core storage. The normal (AAA) address is used to indicate the I/O attachment. The % sign designates tape I/O control and the X indicates the I/O attachment. The character assigned to the X-position is dependent on other feature asignments, and is controlled by interface plugging. The units position is always the digit-1, because this is the only control setup for I/O control. The d-character of R or W designates a read (receive) or write (transmit) operation respectively.

The write operation starts by transferring the (BBB) address from core storage to the I/O adapter. The

address is modified by −1 or +1 depending on the interface plugging. The operation may stop after one character has been transferred, if the single character transfer line is ON. Otherwise the operation continues until a group-mark with a word-mark is sensed as the B-field character. The group-mark is not transmitted, but is used to force a disconnect signal to the I/O adapter.

The read operation starts by transferring the first I/O character to the (BBB) address in core storage. The address is modified by −1 or +1 depending on the interface plugging. The operation may end with a single character transfer if the I/O indicates. If a continuous record is being transferred, the operation continues until either a group-mark with word-mark is read from the B-field, or an end of transmission is signalled from the I/O adapter. The end-of-transmission signal forces a group-mark into the following position of the B-field in core storage.

Operations calling for single character transfer require reservicing at intervals. The operation must be reinitiated for each transfer. The B-address of the original instruction must be updated either by an index operation or an arithmetic process. The remaining time may be used for other programming.

L (% x 1) (BBB) d — LOAD I/O ATTACHMENT

The L (Load) operation is used to send or receive data to/from the I/O attachment with word-marks. A word-mark in core storage transmits as a word separator character (A 841). The word separator positioning is correct only with a +1 modifier progression. If the reverse progression is required, the positioning must be controlled within the I/O adapter.

The breakdown of the (AAA) address and the d-character are the same as for the M-operation. The data flow and control differs only in the handling of the word-mark. Both the word separator and the character are written from one storage location. On a read operation the word separator is first stored, and then replaced, by the next character with a word-mark. The start and end of record controls are identical to the M-operation.

U (% x 1) d — UNIT CONTROL

The U (Unit Control) operation code is used to establish major controls for engaging or conditioning the I/O attachment. The (AAA) is used to designate that the I/O attachment is to be controlled. The same (% X 1) used in the Move and Load instructions determines the selection. The X-character becomes that assigned to the I/O attachment by interface plugging.

No (BBB) address is used, since data is not being transferred. The d-character designates the control to be set as follows:

d-character	*Control*
E	I/O Unit-1 Control provides a signal to the I/O Attachment to engage, receive, etc.
D	I/O Unit-2 Control provides a signal to the I/O Attachment to disengage, transmit, etc.

K d — I/O SELECT CONTROL

The basic Select Stacker (K) code is used with additional modifiers to provide I/O select controls. The d-characters are assigned as follows:

d-character	*Control*
A	I/O 1-Select provides a signal line to the I/O attachment. By interface plugging this signal can be retained until the I/O Transfer 2 is brought up by the I/O Adapter.
B through N	I/O 2-Select through I/O 14-Select respectively provide pulse signal lines to the I/O attachment. These signals are of about 6 μs duration.

B (*III*) d TEST AND BRANCH

The Test and Branch (B) operation is identical to the basic system tests. The I/O controls are provided with eight I/O transfer lines to receive status conditions from the I/O attachment. These are eight new d-character modifiers assigned as follows:

d-character	*Test*
1	I/O Transfer—is normally used to indicate an error condition in the I/O attachment. A latch is provided by interface plugging to retain the signal When a test is made the latch is reset and a reset signal is sent to the I/O adapter.
2	I/O Transfer 2 resets the latch used to retain the I/O Select-1 line. It may be used for any transfer signal, if the reset feature is not used.
3	I/O Transfer 3 is normally used to indicate that the I/O attachment is ready for a read operation. A latch is provided by interface plugging to retain the signal. The latch is reset with the rise of I/O Read Call.
4	I/O Transfer 4 is normally used to indicate that the I/O Attachment is ready for a write operation. A latch is provided by interface plugging to retain the signal. The latch is reset with the rise of I/O Write Call.
5	I/O Transfer 5 resets the latch used to retain the I/O disconnect line. It may be used for any transfer signal, if the reset feature is not used.
6	I/O Transfer 6 has no normal assignment.
7	I/O Transfer 7 has no normal assignment.
8	I/O Transfer 8 has no normal assignment.

I/O LOAD CONTROL

When the 1401 has stopped, a load operation can be forced from the I/O Attachment. This signal sets the DELTA PROCESS and DELTA B-CYCLE latches ON and all other cycle latches OFF. Simultaneously, it forces latches and conditions to indicate the instruction L (% X 1) (001) R. The transfer of data proceeds in the normal manner. Data starts entry at address 001 and continues with a +1 modifier. The operation continues until the I/O attachment signals an end-of-record or end-of-transmission. B-field group-marks are ignored. A new group-mark is written at the end of the transmission. Following the transmission, an instruction is forced starting with address 001. The processing continues in this subroutine until stopped.

I/O Attachment Connections

The I/O attachment is connected to the system through a 200 pin pluggable connector. It is intended to accept any one of several types of I/O attachments. Signal lines are provided to cover all contingencies. The control lines fall into three groups. One group consists of signals being sent from the 1401 to the I/O attachment, and they are classed as output lines. A second group consists of signals from the I/O attachment to the 1401, and they are classed input lines. These two groups are shielded lines using a line driver and terminator combination. The third group is used for interface plugging. These lines are at logic levels and cannot be carried through the cable. They are intended to control features needed with one attachment but not with another. By connecting these on the interface of the plug, attachments can be interchanged as required.

Output Lines

Each of the output lines from the I/O control is powered by line drivers for normal twisted pair or coax. line with a C-level. If the lines are not used, they are left open.

Select I/O Attachment: This signal is available on any operation during which the SELECT I/O ATTACHMENT latch is set. It is used to indicate that the 1401 is ready to execute an I/O operation.

I/O Attachment Read Call: This signal is available when both the I/O ATTACHMENT and READ CALL latches are set. It indicates a read operation has been programmed.

I/O Attachment Write Call: This signal is available when both the I/O ATTACHMENT latch and the WRITE CALL signal are on. It indicates a write operation has been programmed.

I/O Output Lines (8-4-2-1-A-B-C): These seven lines feed data from the 1401 to the I/O attachment. The data comes from the B-register when the service response trigger is set.

I/O Unit Control 1 and I/O Unit Control 2: These two signals are identical except for instruction decode. They result from E- and D-modifiers with the Unit Control (U) operation respectively. Their use in the I/O attachment varies. In general they produce the interlock or engaging conditions for the read and write operations. The signal is sent out prior to the actual data transfer. In some cases the #2 signal is used after data transfer for disengaging the I/O attachment.

I/O Select Lines (1 through 14): This group of lines is used to send conditions or control signals to the I/O attachment. They are developed from the modifier of the Stacker Select (K) operation code. They supply a 6 μs pulse for control. The #1 position has provisions for a DC latch output by interface plugging. The latch is reset by the I/O Transfer 2 input.

Service Response: This signal is sent to the I/O attachment to indicate that the 1401 is in operation to process data. The signal rises during the 1401 B-cycle. Its fall time depends on the mode of I/O operation. Interface plugging allows control from either the rise or fall of the service request input line. For continuous transfer, the service response trigger need not be reset until the next request for a character transfer. The service response gated with write call controls the read-out to the I/O attachment.

I/O Disconnect: Signals the I/O attachment that a group-mark has been sensed in the B-field. The 1401 ends its operation and starts the next instruction. It indicates the end of record to the I/O attachment on a write operation. On a read operation, it indicates that the remainder of the transmission is not being stored.

I/O Process Check: Signals the I/O attachment that the 1401 has detected an error. It can be used in the I/O attachment to show an invalid transmission. The 1401 stops at the end of the operation and requires a START RESET.

Reset Transfer Error Latch: Provides a means of resetting the I/O ATTACHMENT ERROR latch. The signal occurs when a test and branch operation is made on I/O Transfer 1. If the latch is used in the 1401, it is reset also.

1401 Reset: Provides the I/O attachment with a reset line to manually reset error conditions or latches which may have been left. Interface plugging allows selection of either a positive or negative swing from an in-phase or inverted start-reset signal.

1401 is Stopped: Provides the indication to the I/O attachment that the 1401 has stopped and is not testing for I/O conditions. The I/O attachment may then force an I/O Load operation to start the 1401 on a sub-routine.

Clock Timings (000-030) (030-060) (060-090) (090-000): The four basic 1401 clock pulses are made available to the I/O attachments to allow synchronizing controls during transfer.

Input Lines

Each of the input lines to the I/O control is terminated for normal twisted pair or coax. line with a C-level. If the lines are not used, they are left open.

Service Request: Sets the service request trigger to start the 1401 clock if in either read or write operation. Starting the clock causes the 1401 to read in or read out. The input signal may be either a pulse or a DC level.

I/O Selecting Unit: Serves as a gate on the service request trigger. When OFF, the service request line is not recognized.

I/O Input Lines (8-4-2-1-A-B-C-WM): These eight lines feed data into the 1401 on a read operation. The data is entered into the A-register when the clock is running and the I/O read call is up.

End of Transmission: This line serves to terminate a read or write operation in the 1401. It is brought up following the service response signal for the last character. It must be up for at least 12 μs for a read operation and at least 18 μs for write operation.

Single Character Transmission: When operating at low character rates, the 1401 can be returned to its regular program. Holding this line in its plus level forces an I-E Change after one character has been transferred. On a Load operation, with word-marks, the transmission is held for two cycles to accommodate the word separator.

I/O Readdress: This signal forces the READDRESS latch to cause the next transfer to occur at the same storage address. It can be used by the I/O attachment to control word-mark entry or in self-correcting error controls.

I/O Transfer Lines (1 through 8): These are signals from the I/O attachment indicating conditions. Their prime function is to allow the 1401 to Test and Branch if the condition has been reached. The first five of these lines have tentative condition assignments under control of the interface plugging. If they are not required for this purpose, one or all can be used for another purpose.

I/O Transfer 1	A. Error B. End of Record
I/O Transfer 2	A. Reset I/O Select 1 B. End of Record
I/O Transfer 3	A. Ready to Read
I/O Transfer 4	A. Ready to Write
A-Reg A not B A-Reg B not A A-Reg AB	A-Reg Select 1
A-Reg not 12 A-Reg 1 not 2 A-Reg 2 not 1 A-Reg 12	A-Reg Select 2
A-Reg not 84 A-Reg 4 not 8 A-Reg 8 not 4	A-Reg Select 3

These signals are plugged to assign an alphabetic character to the I/O attachment. The three inputs gate the set of the I/O ATTACHMENT latch.

Set Error Latch Out—Set Error Latch In
Set Read Latch Out—Set Read Latch In
Set Write Latch Out—Set Write Latch In

These three pairs of signals function in the same manner. They provide the latchback for a logic latch and thus extend the basic signal. The original signals come from the I/O Transfer 1, 3, and 4 lines respectively.

Select-1 Out – Select-1 In: The out terminal of this pair is the I/O-1 select line before powering. The in-line is the collector output of a logic latch. Coupling the two extends the output until the latchback is broken by a signal on the I/O Transfer 2 line.

I/O Transfer 5	A. Reset I/O Disconnect
I/O Transfer 8	A. Force Odd Redundancy 2

I/O Load: This line can be used by the I/O attachment, when the 1401 is stopped without an error condition, to initiate a load operation. The following signals are developed to start the operation without an instruction cycle.

A. Delta Process latch set
B. Op decode forced to L (Load)
C. Unit Select latch set
D. I/O Attachment latch set
E. Generate address 001
F. Read Call latch set
G. Select TU 1 latch set
H. Set Delta B-Cycle latch

I/O Error Light: This signal is supplied by the I/O attachment when its ERROR latch is set. The external input/output indicator lamp is lighted on the console.

Interface Lines

The interface lines are treated in pairs or groups as they are normally associated for control. When a feature is not needed, it is left unplugged. In a few cases the +6 volt line may be used for continuous control, but this may result in forcing the same condition for a tape operation. The problem of other than normal pluggings is of no consequence with standard I/O attachments.

I/O DISCONNECT OUT 1
I/O DISCONNECT OUT 2 — I/O DISCONNECT IN

This group of terminals allows the selection of a 6 μs pulse on #1 or a latch output on #2 to feed the I/O disconnect output line. The latch is reset by the I/O Transfer 5 line.

START RESET OUT
NOT START RESET OUT — START RESET IN

This group of terminals allows the 1401 reset line to have either a positive or a negative swing to feed the I/O attachment.

SERVICE REQUEST OUT 1
SERVICE REQUEST OUT 2 — SERVICE REQUEST IN 1

These signals are used to control the reset of the service response trigger. Service Request Out-1 is a feed-through of the service request input, while #2 is the inverted output. Number 1 is used when the input is a short pulse and #2 when the input remains ON for the duration of the transfer.

I/O TRANSFER 1
I/O TRANSFER 2
+6 VOLTS — I/O TRANSFER EOR

These signals are used to select the end-of-record control from the I/O attachment. I/O Transfer 1 and 2 are feed-through of the input signals. If neither line is used the EOR must be connected to +6 volts.

FORCE ODD REDUNDANCY 1
FORCE ODD REDUNDANCY 2 — FORCE ODD REDUNDANCY

These signals are used to force an odd-bit redundancy check during an I/O operation. With no connection the output is checked for even-bit count. Force-Odd-Redundancy-1 forces the odd check on all transfers. The Force-Odd-Redundancy-2 comes from the I/O attachment over the I/O Transfer 8 line.

Inhibit Word-Mark 1 — Inhibit Word-Mark: When these terminals are connected, the I/O attachment can enter the word-mark bit directly on a Load operation.

Not I/O Read Call 1 — Not I/O Read Call: These terminals are connected to allow read-in from low-order to high-order (−1 modifier). The resulting signal prevents the normal +1 modifier.

Set Readdress Latch 1 — Set Readdress Latch: These terminals are connected when the I/O Attachment requires multiple access to a core storage position. Control is effected through the I/O readdress input.

Reset A-Reg Error 1 — Reset A-Reg Error: Connecting these two terminals causes the A-REGISTER CHECK latch to be reset if the I/O attachment already has detected the error.

Select I/O Out — Select I/O In: These two lines are normally connected to serve as an interlock. They control the set of the SELECT I/O ATTACHMENT latch. If the line were not brought through the plug, the latch would set continuously with the plug removed.

I/O Write Call Out — I/O Write Call In: These terminals are normally connected for a write operation. They cause a B-cycle to follow the instruction to fill the B-register with the first character to be transferred.

I/O Error Conditions

Like the tape system, errors are divided into two groups. One group involves the error checks within the basic system. The second group are those sensed in the external area. The method of error correction for an I/O attachment varies, depending on a number of factors. Two major items are involved, one being the possibility of a recycle operation, and the second is the communication problem. The latter includes long distance transmissions.

Processing Unit Errors

Data transfer errors within the basic system are detected in the normal manner. Being an I/O operation, the system is not stopped until an I-E Change. Errors resulting from operation decode or addressing cause the system to stop immediately. Both types of errors develop the process-check signal, which is available to the I/O attachment.

The processing unit errors affect the operation in different ways depending on the mode of I/O operation. At the higher transmission speeds, the system remains in operation for the full record. Except in the case where the storage area is too small, the transmission error is considered after the record is complete. At the slower transmission speeds when only a single character is transferred, the system stops before the

transmission is complete. These special conditions require controls within the I/O attachment to stop the transmission and/or the device.

I/O Attachment Errors

Errors within the attachment or its associated adapter may set a single latch, or they may use several. The I/O Transfer-1 is reserved for data transmission errors. When RESET A-REGISTER ERROR is connected on the interface, the character entering on a read operation does not set a PROCESS CHECK. Under this condition it may be possible for the attachment to recycle and transmit the corrected character. Use of the I/O readdress line causes the same storage address to hold for another cycle.

Errors resulting from other than data transmission usually require different correction routines. These can be detected by using separate I/O transfer lines for each condition. At the end of the transmission a series of test and branch instructions are used to determine the next steps to be taken. The instruction manual on the I/O attachment being used discusses the methods of error analysis.

Print Storage

To facilitate the description of machine logic later in this section, figures are used that employ six symbols. Conditions are listed to the left in the figure, and any symbol refers to the condition as stated on the same horizontal line. Figure 68 shows the symbols and an example of their use to indicate a logic condition.

Purpose of Print Storage

The purpose of print storage is to provide more 1401 processing time. This is accomplished by reading the information to be printed from 1401 storage into the auxiliary (print storage) unit. Once this has been done, the 1401 can resume processing. While processing is taking place, the print storage is scanned to cause printing.

To achieve the intended purpose of more processing time, the 1401 must not try to instruct either the printer or carriage while either unit is busy. Otherwise the 1401 will be interlocked, stopping processing. Two new branch codes are now available to test the status of the printer or carriage. These branch codes are:

1. B (III) P, branch on printer busy.
2. B (III) R, branch on carriage busy.

Operation

FUNCTIONS OF PRINT STORAGE

A 1401 with the print storage option will perform all the functions concerned with printing that a standard machine will perform. In addition it can do a storage scan of the print storage buffer. However, when doing any kind of print operation, the information is entered in print storage (buffer entry) and then printed. Also,

SYMBOL	MEANING
←	Condition is activated (trigger or latch comes on) due to the ANDING of conditions indicated in the same vertical column.
← Off	Trigger or latch goes off due to the ANDING of conditions indicated in the same vertical column.
X	Condition as stated active.
—	Condition as stated inactive.
↓	When condition as stated becomes active.
↑	When condition as stated becomes inactive.

EXAMPLE

Condition A	↑
Not Condition B	X
Condition C	
Condition D	—
Condition E	←
Not Condition F	—

Condition E becomes active due to the ANDING of (A becoming Active) (Not B) (Not D) (F)

Figure 68. Example of Logic Display with Symbols

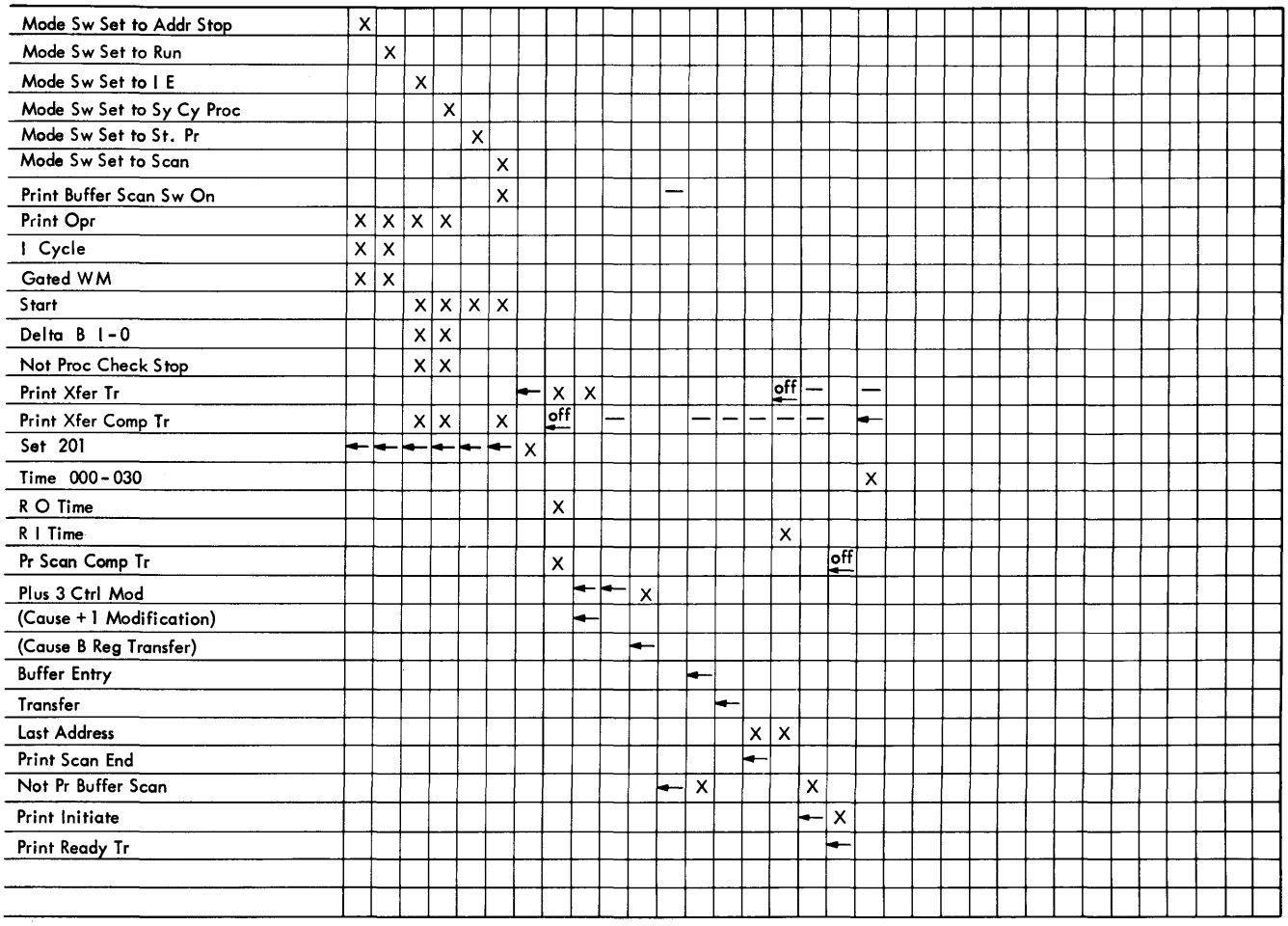


Figure 69. Print Storage Transfer and Entry

	Other Times	Entry	Buffer Scan	Print Scans		
				Op	WM	St. Pr.
Enter B C D Code (1) Into Buffer	No	Yes	No	No	No	No
Regenerate (1)	No	No	Yes	1 - 49	1 - 49	1 - 49
Set Hammer Fire Cores (1)	No	No	No	1 - 48	1 - 48	1 - 48
Regenerate (1)	No	No	No	No	No	No
Set Compare Equal Cores (0)	No	No	No	1 - 48	1 - 48	1 - 48
Regenerate (0)	No	No	No	No	No	No
Set P L C Cores (1)	No	No	No	1 - 49	1 - 49	1 - 49
Regenerate (1)	No	No	No	2 - 49	2 - 49	2 - 49
Set Error Cores (1)	No	No	No	1 - 49	1 - 49	1 - 49
Regenerate (1)	No	No	No	2 - 49	2 - 49	2 - 49

Figure 70. Set Print Storage Cores

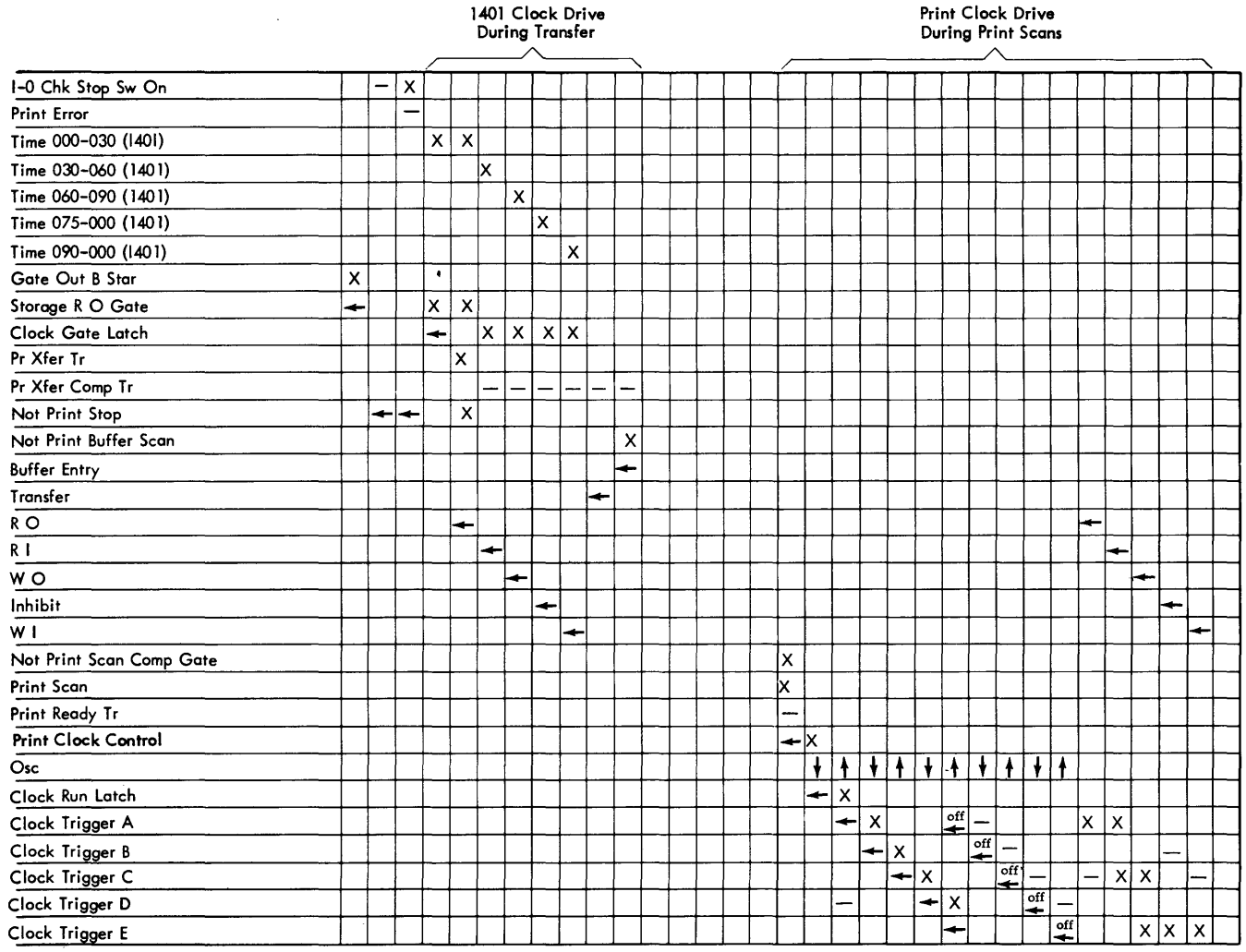


Figure 71. Print Storage Drive Pulses

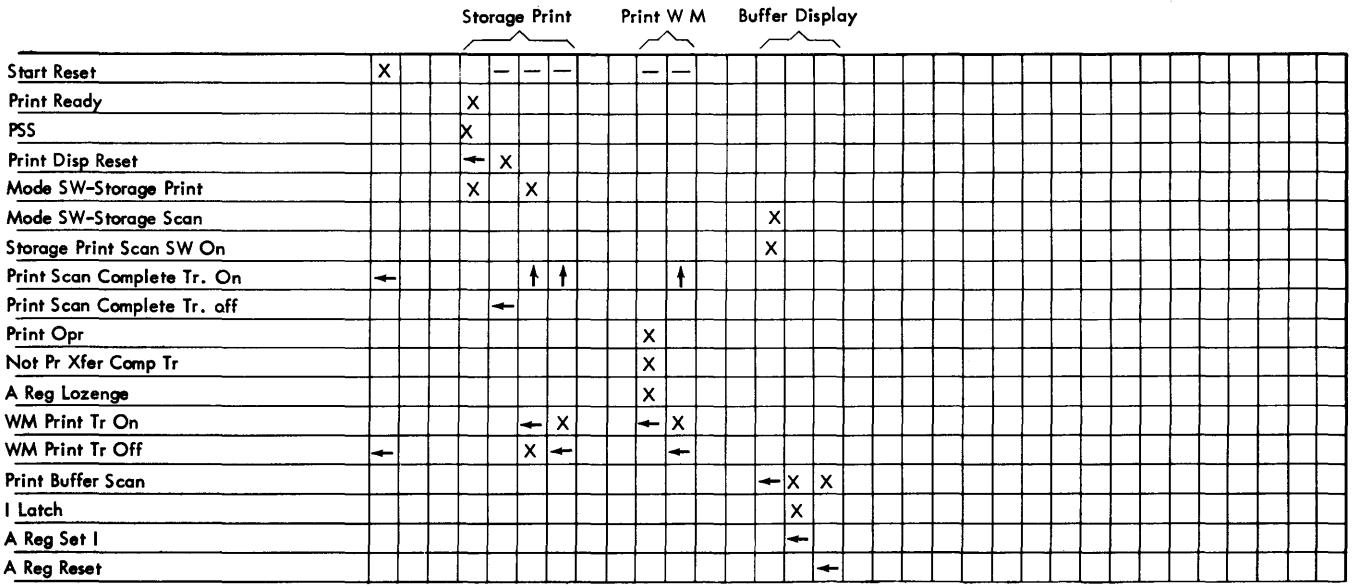


Figure 72. Storage Print, WM Print, and Buffer Display

it is possible to single-cycle through an entry. In I-E mode the 1401 will stop at the end of I-phase. Next pressing the start key will complete the entry before the 1401 stops; then the line will be printed. At the end of an entry, print-scan-end turns OFF the DELTA PROCESS (or D-CYCLE) latch. If a combined input-output operation is not set up, all-scans-complete is activated. All-scans-complete causes the DELTA PROCESS latch to turn ON when the Mode Switch is set to RUN or ADDRESS STOP, thereby allowing the 1401 to continue processing. After information has entered print storage, this information is printed independently of the 1401.

TRANSFER, ENTRY, AND PRINT (FIGURE 69)

The operation of the print storage buffer can be classified into three phases:

1. Transfer
2. Entry
3. Print

During a transfer the units and tens rings are advanced from the reset position 132 (T-13, U2) to the first position 001 (T-0, U-1), and thereafter are advanced by plus-one until position 132 is again reached. The transfer ends at this position. A transfer is defined

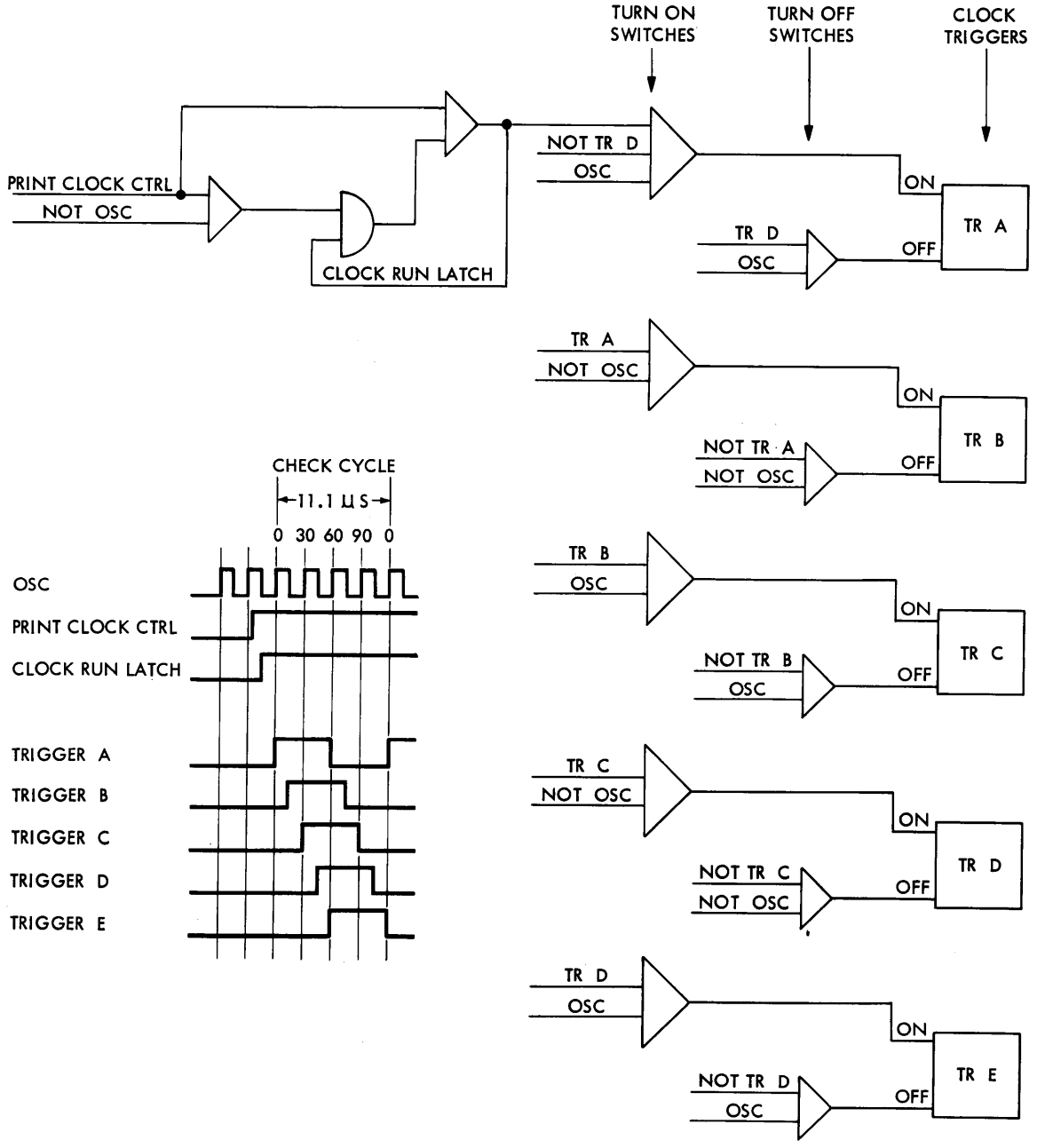


Figure 73. Print Storage Clock

as the advance of the print storage units and tens rings from the 1401 clock. It occurs when the print transfer complete trigger is OFF. Printing cannot occur during a transfer. The transfer is used either for buffer entry or for a print storage scan. During a transfer (except during storage print-out) the 1401 storage address register is set to 201 and is modified by plus-1 to address storage positions 201 through 332, while the rings address print storage positions 001 through 132.

Buffer entry is defined as entering information into the print storage buffer from the 1401. Buffer entry always occurs during a transfer except when a print storage scan (for errors) is taking place. Information enters the buffer from location 201 to 332 in the 1401 except during storage print-out. During storage print, information enters the buffer from the block of storage selected by the address switches in the 1401. The B-register output is switched with buffer-entry to allow information to read into the print storage character bit latches. Information in the bit latches is transferred to print storage (write ones) during time W-1 (090-000). Buffer entry also blocks the output from the sense amplifier so that previously stored information cannot read into the character bit latches and be regenerted. Regeneration is allowed when buffer entry is not active (print storage scan). During transfer, the equal-check plane is the only check plane in which one (1) is written (Figure 70).

Printing occurs after information is placed in the buffer during buffer entry. The print ready trigger is turned ON to set up the print operation. Print scans occur when the print ready trigger is turned OFF, causing the print scan trigger to turn ON. During print scans, the print clock runs, and the units and tens rings are advanced in the manner required to address print storage for printing (Figure 71). The print clock is stopped between print scans and also at the end of the last storage cycle in print scan 49. The rings are left sitting in *the last address* position 132 (T-13, U-2) at the end of print scan 49.

PRINT STORAGE SCAN SWITCH

When the mode switch is set to *Storage Scan*, and when the print storage scan switch is set to ON, pressing the start key causes the B-register to display positions 201 through 332 from 1401 storage and the A-register to display positions 001 through 132 from

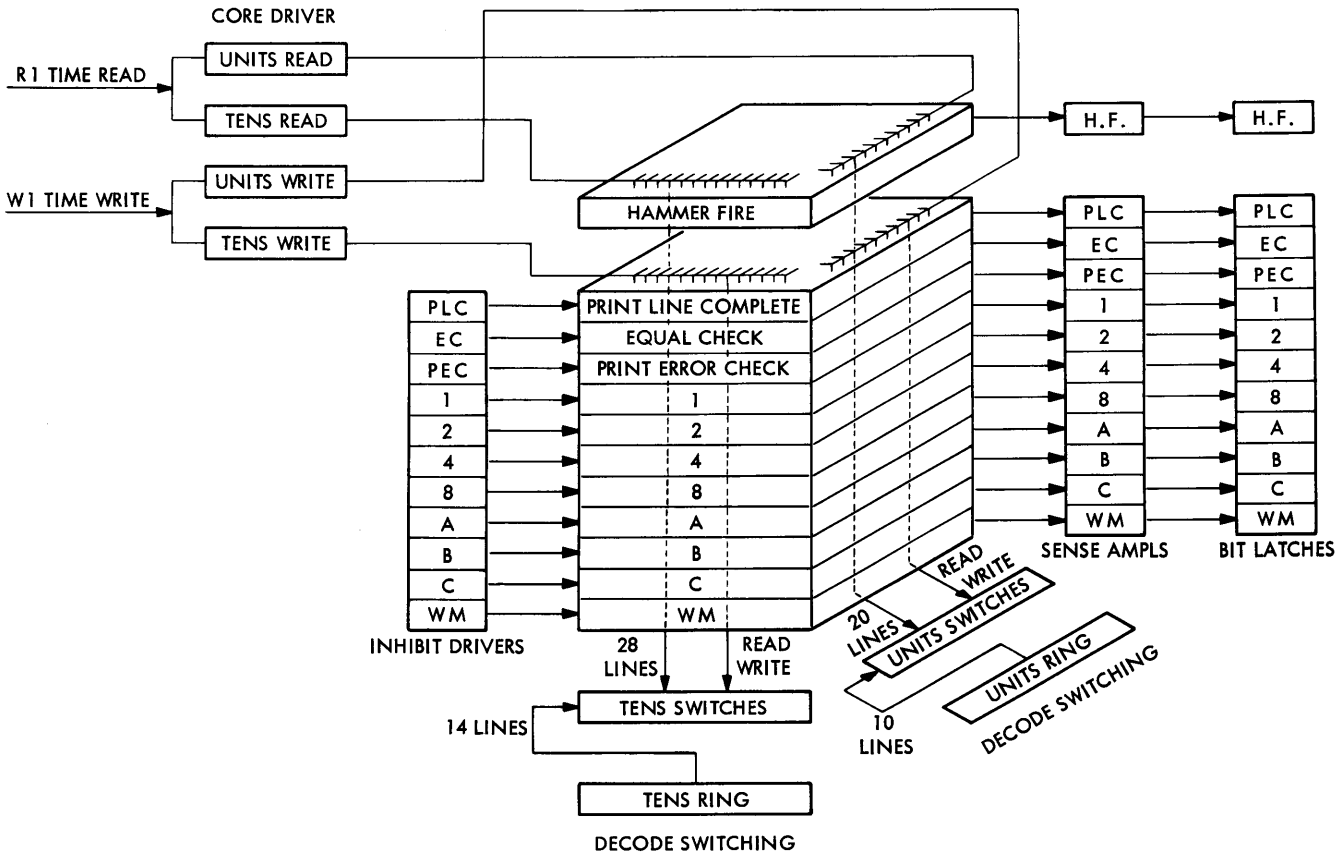


Figure 74. Print Storage

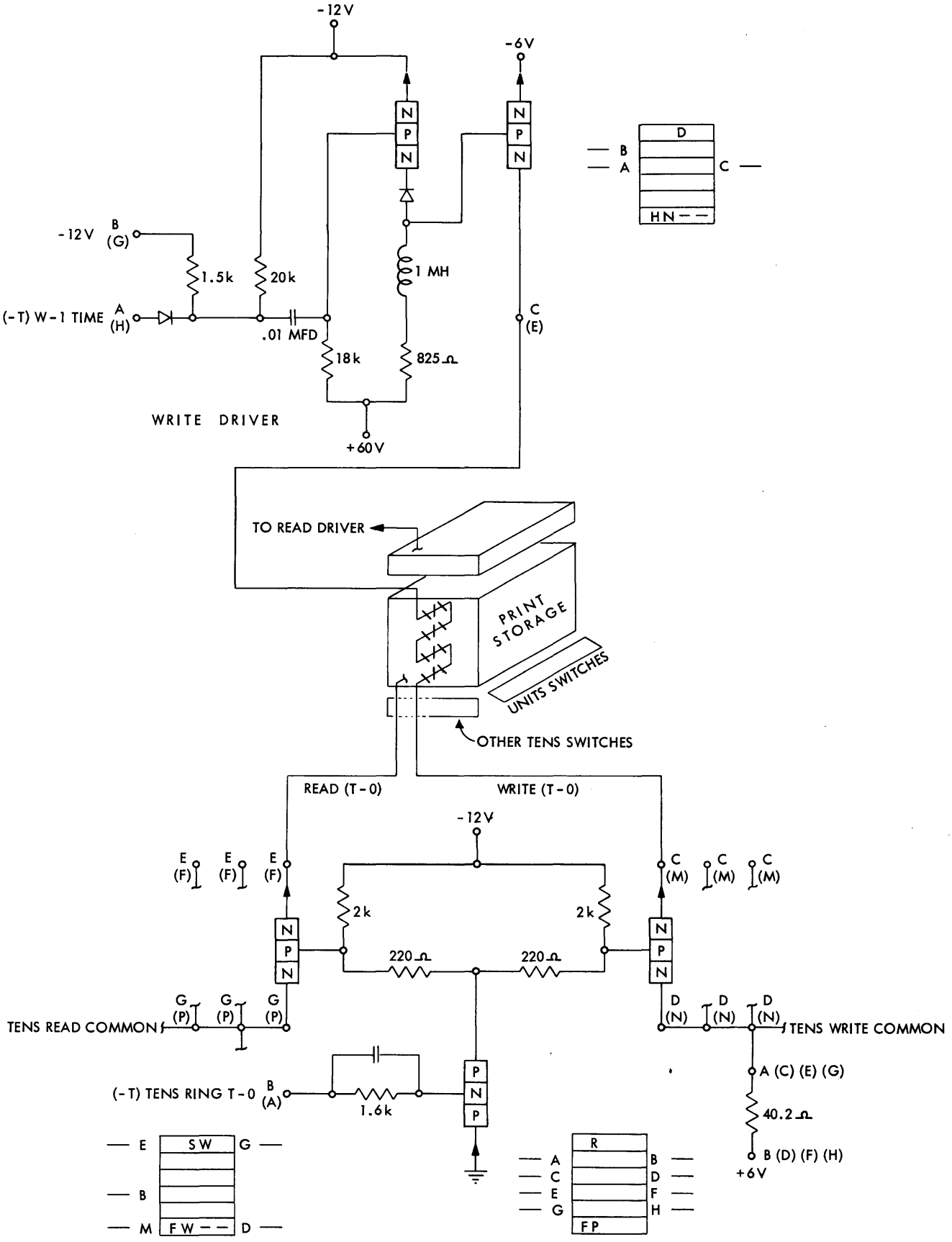


Figure 75. Tens-Zero Address Switch

the print storage buffer. The operation stops at the end of one transfer scan if no errors are detected. If an error is detected, the indication can be reset and the scan continued automatically from the next higher address.

When the print storage scan switch is set to OFF, a normal 1401 storage scan takes place when the start key is pressed.

STORAGE PRINT, PRINT WM, AND BUFFER DISPLAY (FIGURE 72)

The figure shows the switching used to turn OFF the print scan complete trigger for the second line of storage print-out. It also shows the operation of the WM trigger for both storage print and print WM Opr. In addition, the switching to display the print storage contents in the A-register lights during a print buffer scan is shown for a 1-bit. The A-register is continuously reset, the set to the A-register lights the display.

Clock and Timing Pulses (Figure 73)

The print-clock has a 11.1 μs cycle. The clock cycle is divided equally into four times:

1. RO (000-030), Ring Advance
2. R1 (030-060), Read
3. WO (060-090), Switching or logic
4. W1 (090-000), Write.

Inhibit time is 075-000. The print clock runs only during the 147 subscans of the 49 print scans associated with printing and checking. It does not run when information is entered in the print storage buffer from 1401 storage, or when a storage scan (for errors) is made of the print storage buffer.

The 1401 clock provides the RO, R1, WO, and W1 pulses used by the print storage buffer when information is entered into the buffer from 1401 storage, or when a print storage scan (for errors) is made (Figure 71). A clock gate latch is one condition necessary to gate-out the 1401 clock pulses. Once turned ON, the latch stays on for one 1401 clock cycle. It is turned ON by storage-read-out-gate (B-STAR GATE-OUT) and time 000-030.

Core Array (Figures 74 and 75)

Twelve bit planes are used for storage and checking. Of the 140 storage positions available, 132 are used.

Two read drivers supply current for reading, one for the ten (U-0 through U-9) units planes, the other for the fourteen (T-0 through T-13) tens planes. Likewise, two write drivers supply current for writing, one for the units planes, the other for the tens planes. The read windings and the write windings pass through the same cores in opposite directions. A position in storage is selected by conditioning one of the ten units switches and one of the fourteen tens switches. Selection of any one position lasts for one clock cycle and includes read and write time.

All bit-planes and check-planes have a sense amplifier (Figure 76), and all bit-planes and check-planes except the hammer fire check plane have an inhibit driver. The write lines do not pass through the hammer fire check plane. A one (1) is written in a core

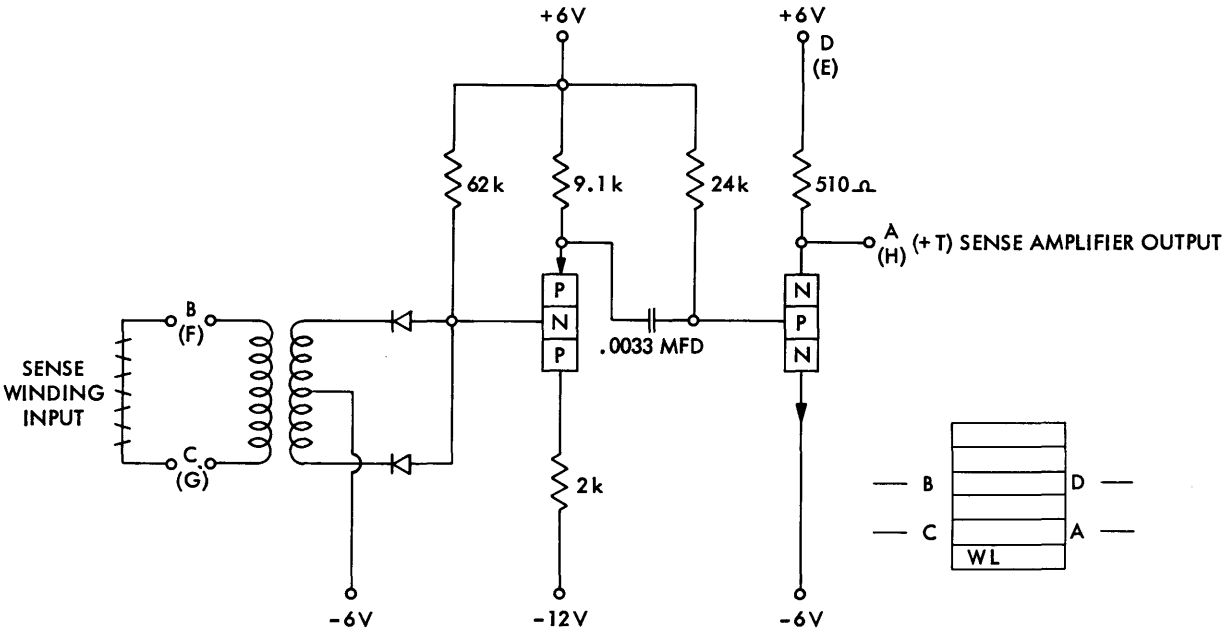


Figure 76. Print Storage Sense Amplifier

in this plane when the hammer driver corresponding to the storage position is fired. The one (1) is written while the hammer driver is ON and results from current flow through the hammer-response line and through the five-turns wound on the associated core in the hammer fire check plane.

NOTE: The word *inhibit*, when used with print storage, means *prevent the writing of a core (a bit).*

BIT LATCHES (FIGURE 74)

Twelve bit latches are used to retain the information read from print storage. These latches correspond to the I/O and B-register latches in a machine without the print storage option.

Print Storage Addressing

The output from the units and tens print storage rings address the print storage buffer. The buffer cannot be addressed by any other means (Figure 74). At the same time that print storage is addressed, the output from the rings addresses the particular hammer driver that corresponds both to the storage location and the print position (Figure 77). While it is addressed, a hammer driver can be selected to fire only during print scans when a print compare equal occurs. The tens-drive T-13 used in addressing hammer driver 132 (and also 130 and 131) is the switched output of tens ring T-13 and Transfer or not-print-scan-complete. Thus, while the rings are reset to position 132, there is no input to either pin of hammer driver 132. Tens-drive T-13 is used in developing last-address, a condition used both during transfer and during print scans.

PRINT STORAGE ADDRESS RINGS

The print storage feature utilizes a units address ring of ten positions and a tens address ring of fourteen positions. These rings are used to address the print storage buffer and the hammer matrix during the 49 print scans required for printing and checking. They are also used to address the print storage buffer during the entry of information from 1401 storage into the buffer. In addition, the rings are used to address the print storage buffer when it is being scanned for errors during a print storage scan.

The rings are always reset to T-13 (tens thirteen) U-2 (units two) and always advance from 132 to 001 (T-0, U-1). Other times, they advance by plus-1 at time 000-030 (RO) from the 1401 clock during information entry and during print storage scan; they advance by plus-3 at time 000-030 (RO) from the print clock during printing except when they are advanced from the last position in a subscan to the starting position of the next subscan (Figure 78).

PRINT STORAGE RING DRIVE (FIGURE 78)

The units ring is driven by RO pulses. These pulses are derived from the 1401 clock when information is being entered into print storage, and when print storage is being scanned for errors. These RO pulses are derived from the print clock during print scans.

The tens ring is driven by a tens drive trigger which will be on during RO time when the tens ring is to be advanced. During buffer entry and print storage scan for errors, this trigger is driven by the 1401 clock; during print scans, by the print clock.

Hammer Control

LATCH TYPE HAMMER DRIVERS (FIGURE 79)

The latch-type hammer drivers and the hammer matrix are the same regardless of whether or not the print storage option is installed. When the option is installed, the matrix is addressed from the units and

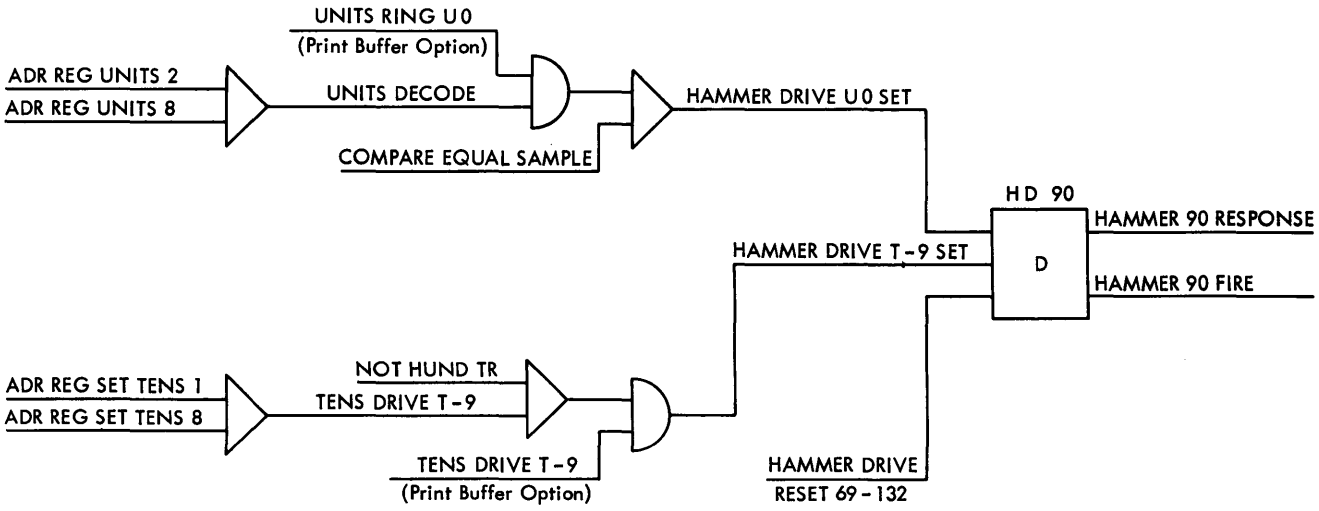


Figure 77. Decode, Firing, and Response for Latch Type Hammer Drive

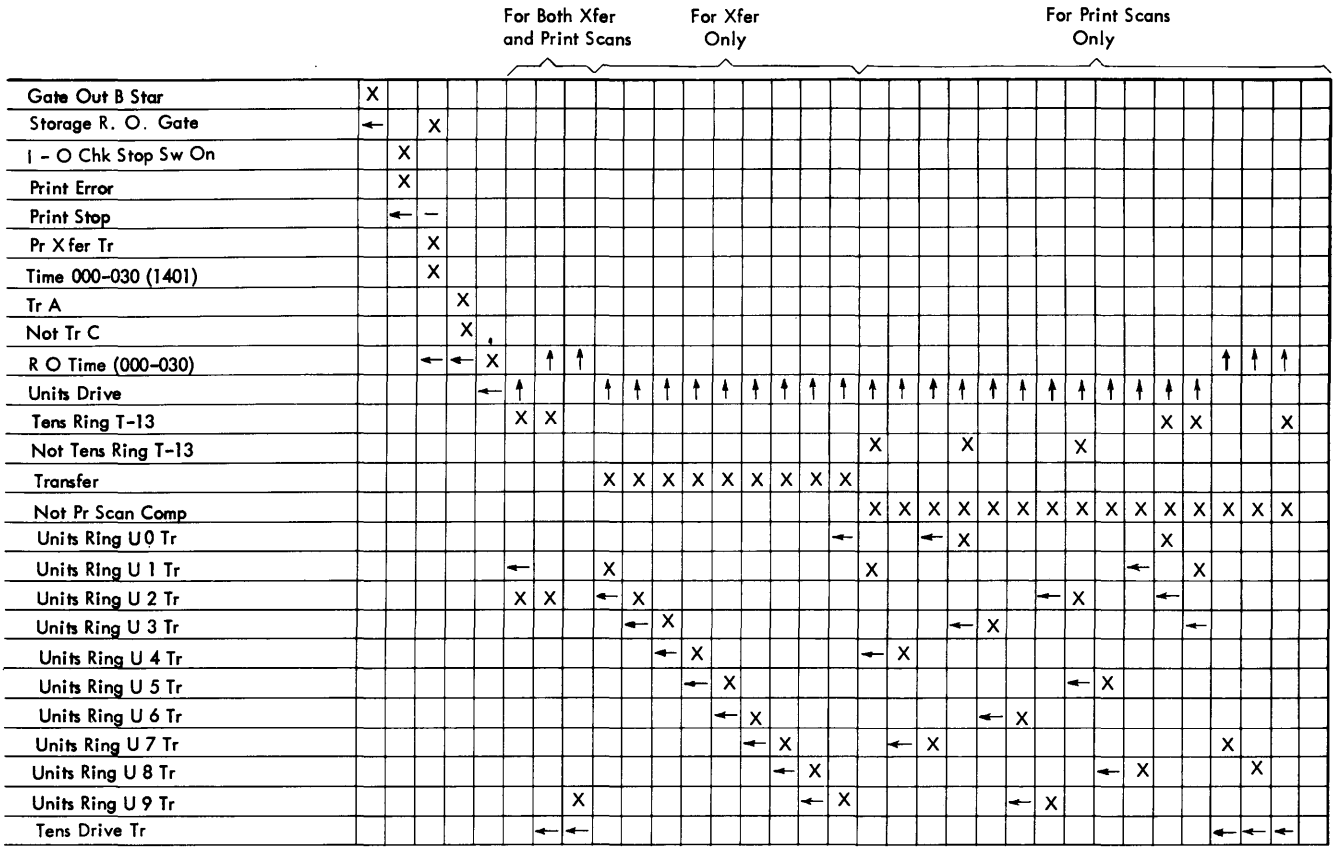


Figure 78. Print Storage Tens Drive and Units Sign Advance

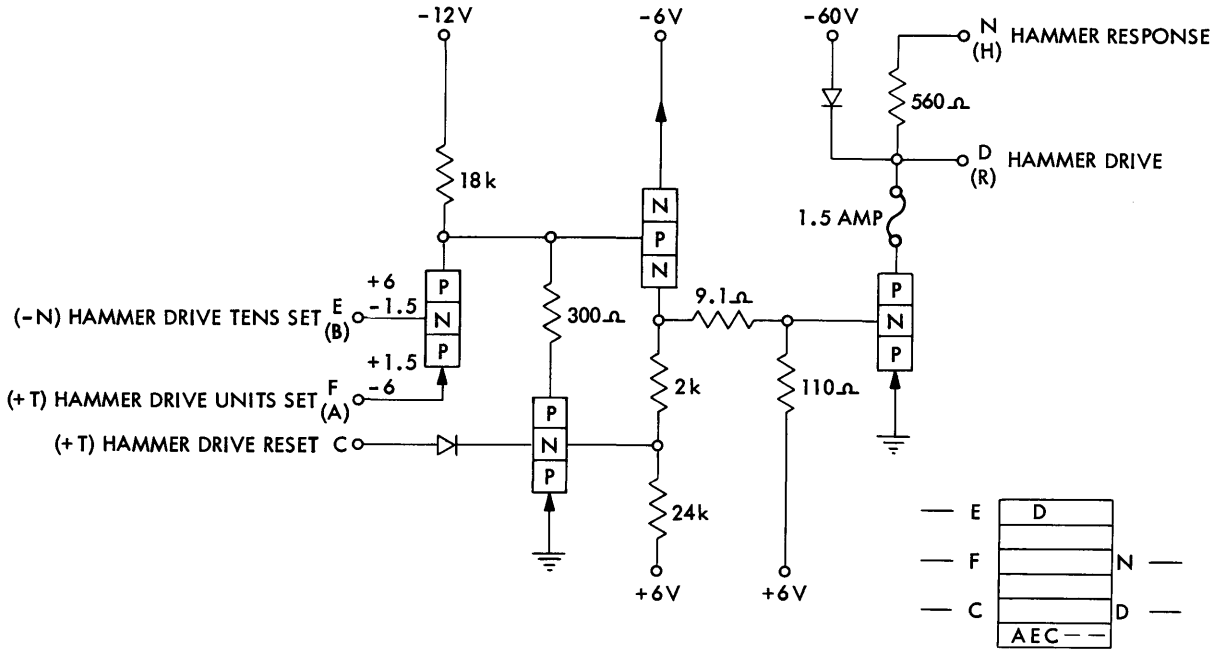


Figure 79. Latch Type Hammer Driver

	Other Times	Entry	Buffer Scan	Print Scans		
				Op	WM	St. Pr.
Print Check Latch	No	Yes	Yes	1 - 49	1 - 49	1 - 49
Parity Error	No	Yes	Yes	1 - 49	1 - 49	1 - 49
PLC Check	No	No	No	1 - 49	1 - 49	1 - 49
Hammer Fire Check	No	No	No	2 - 49	2 - 49	2 - 49
Home Error Trigger	Yes	Yes	Yes	1 - 49	1 - 49	1 - 49
Ring Check	No	No	No	1 - 49	1 - 49	1 - 49
Hammer Drive Lines	No	No	No	1 - 49	1 - 49	1 - 49

Figure 80. Print Storage Checks

tens rings rather than from the decoded output of the storage address register. A hammer driver is selected to fire when a compare equal ANDS with the units decode.

HAMMER MATRIX DECODE (FIGURE 77)

As with the switch-core type of driver, to cause printing, the latch-type driver must be reset, and it must be selected. When the driver is selected the latch is turned ON, causing current flow in the hammer magnet and in the hammer-fire check plane core. When the latch is reset, current flow stops.

Hammer drivers 1 through 66 (in three groups — one per subscan) are reset during tens drive T-9 and T-10 time. Hammer drivers 67 through 132 (in three groups — one per subscan) are reset during tens drive T-3 and T-4 time. Moreover, during printing, the reset for the drivers occurs within the half a subscan just previous to the time the drivers are optioned to print. (The drivers optioned to print in the last half of subscan-one are reset in the first half of subscan-one. The drivers optioned to print in the first half of subscan-one are reset in the last half of subscan-three.)

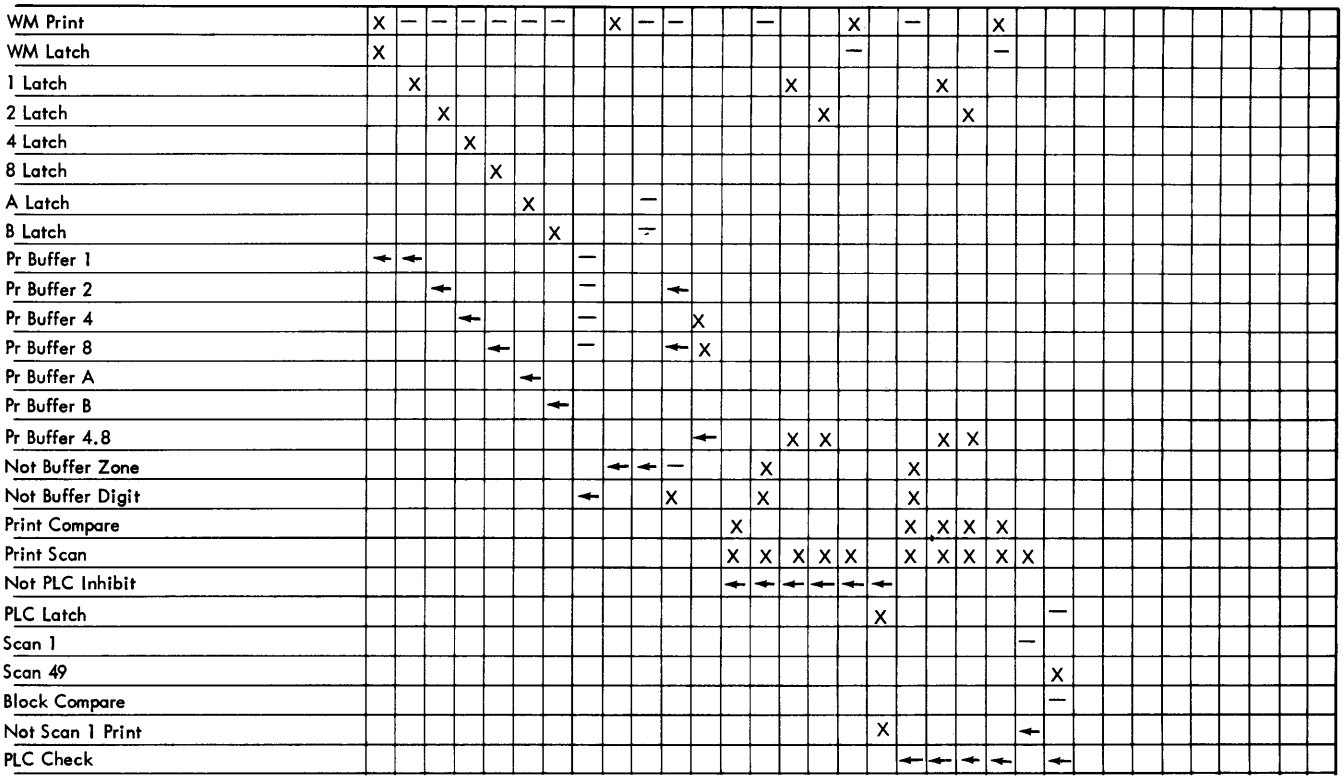


Figure 81. PLC Inhibit and Check

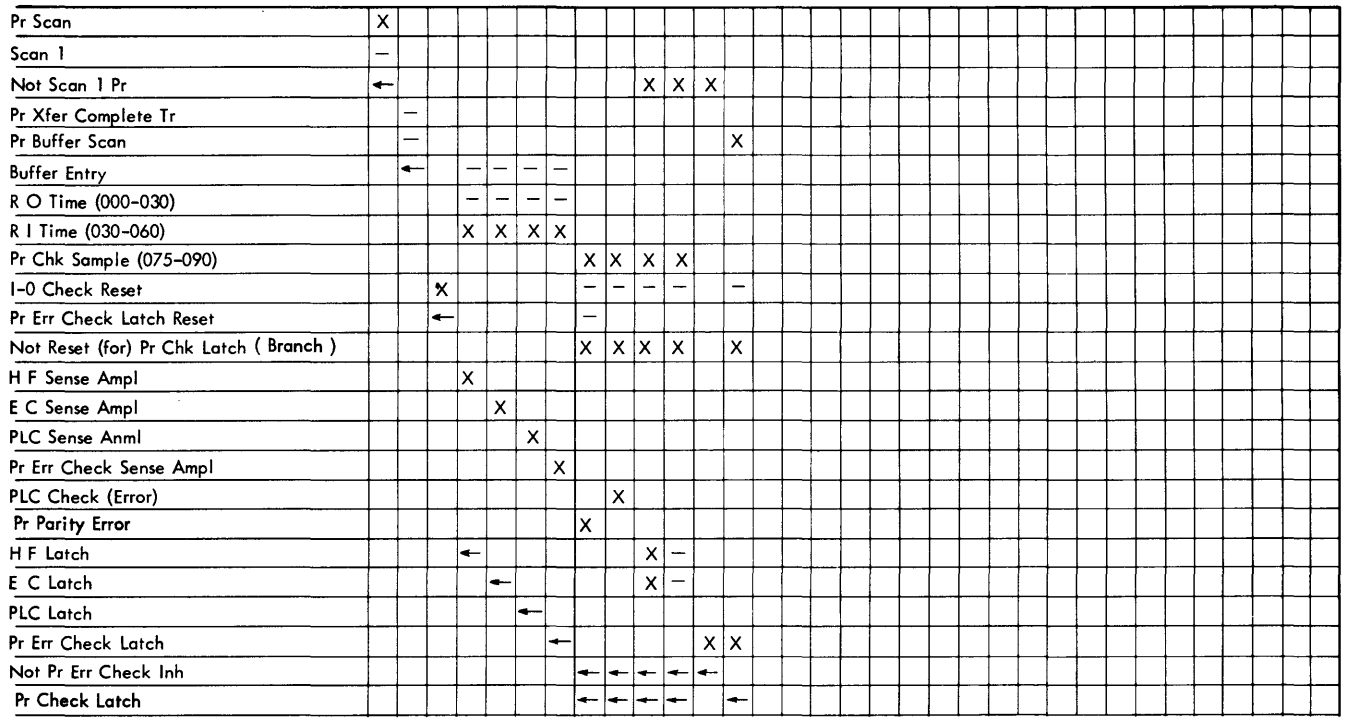


Figure 82. Print Storage Check (Logic)

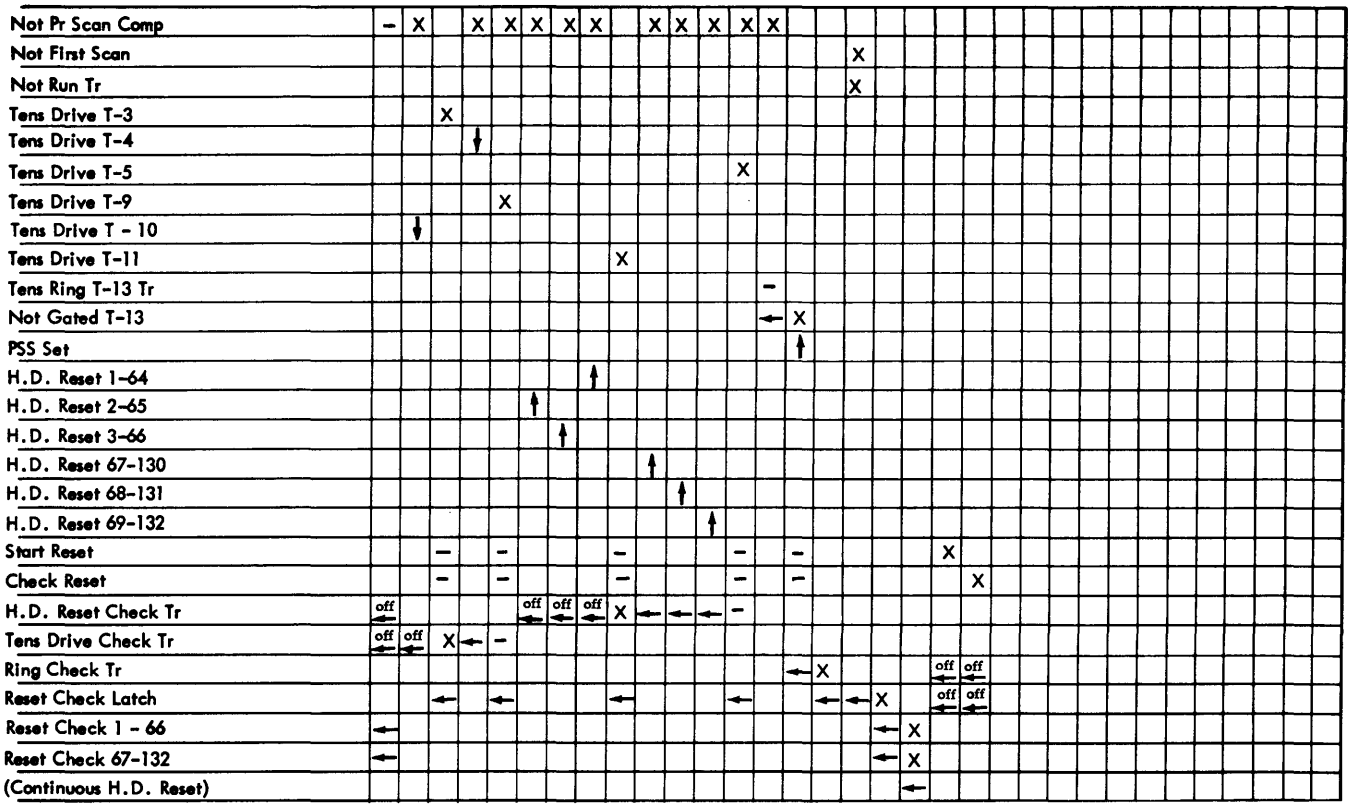


Figure 83. Hammer Drive Reset Checks

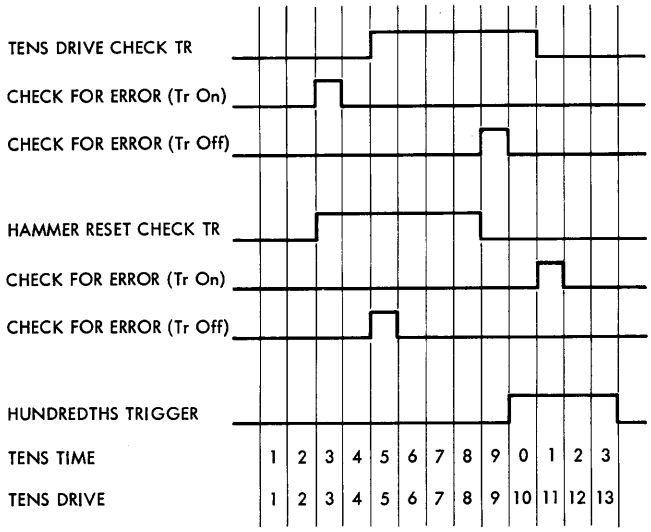


Figure 84. Check of Drive Lines for Hammer Reset

The hammer drivers are reset at the end of printing by print-scans-complete. They are also reset if the RESET CHECK latch comes ON (error conditions to be discussed later). The RESET CHECK latch can be turned OFF only by the start reset key or the check reset key.

In machines using the latch-type driver, 132 positions of storage are scanned regardless of the print positions (100-132) available in the 1403. For a 100 print position 1403, a block compare trigger is turned ON at 090 when the hundreds trigger is ON during subscan one. It comes ON with the hundreds trigger during subscan two and three. For a 132 print position 1403, the block compare trigger is not installed.

The hundreds trigger, used with the latch-type drivers, is turned ON when tens drive T-9 goes OFF and the print scan trigger is ON. The hundreds trigger is reset OFF by not-print scan.

Checking Circuits

PRINT STORAGE CHECKS (FIGURE 80)

The print error checks in a 1401 with the print storage feature have been expanded. They now include a parity check of the print storage bit latches which is made during transfer and during printing. They also include the printing of word-marks which affects the print line complete check (Figure 81). Moreover, with storage print, a hammer fire check plane core is now set to one (1) instead of zero (0) when a hammer driver fires. Consequently, the compare equal check plane core is now set to zero (0) instead of one (1) when the hammer driver is signalled to fire (Figure 82).

HAMMER DRIVER RESET CHECKS (FIGURE 83)

All of the hammer driver reset checks are incorporated into a 1401 with the print buffer feature. However, to replace the Star error check, a ring check trigger is used. If the tens ring is not sitting at T-13 at the end of a print scan while scans are not complete, all hammer drivers are continuously reset for the rest of the print operation.

A check is made of storage and hammer selection during printing. The detection of an error turns ON the RESET CHECK latch. These error conditions are (Figure 84):

1. Ring check Trigger ON.
2. T-11 AND (hammer reset check trigger ON).
3. T-5 AND (hammer reset check trigger OFF) AND Not-print-scan-complete.
4. T-3 AND (tens drive check trigger ON).
5. T-9 AND (tens drive check trigger OFF) AND Not-print-scan-complete.

The hammer reset check trigger is turned ON at the start of tens drive T-3 time; OFF, at the start of tens drive T-9 time. The binary input to the trigger are the six hammer drive reset lines (for the early and late resets in the three subscans). Therefore, this trigger will be turned ON and OFF during each subscan. The tens drive check trigger is turned ON at the start of T-5 time; OFF, at the start of T-11 time. The AC Inputs are Tens Drive T-5 and T-11. The hammer reset check trigger and the tens drive check trigger are reset OFF by print-scan-complete.

The tens drive lines T-3, T-4, T-9, and T-10 are used to reset the hammer drivers. The T-3 and T-9 drive lines also operate the hammer reset check trigger. The T-5 and T-11 drive lines also operate the tens drive check trigger. Correct operation of these triggers in conjunction with the ring check trigger indicates that the drive lines are activated correctly.

Interlocks

PRINT STORAGE READER-PUNCH INTERLOCKS

The reader and the punch cannot start a feed cycle until transfer is completed during a combined operation.

1401 INTERLOCKS (FIGURES 85 AND 86)

The 1401 clock is stopped if the printer is not ready to operate because of:

1. the run trigger being off.
2. a home error.
3. carriage out of forms.
4. print operation called for before last print operation is completed.

The clock is allowed to start when the run trigger comes ON or when a home error is reset. The clock is allowed to start if forms are placed in the carriage. However, when forms have run out and the single-cycle key on the printer is pressed, the clock will be allowed to run until a line of information is entered in print storage. After entry, the line is printed. The single-cycle key on the printer will be effective as described until a hole in channel one is sensed by the carriage stop brushes.

If a print operation is set up before the previous print operation is completed, the clock will start as soon as all-scans-complete becomes active.

The DELTA PROCESS latch is turned OFF if:

1. a process occurs during transfer and the process check stop switch is ON.
2. a print error occurs during transfer and the I-O check stop switch is ON.
3. a STAR error occurs.

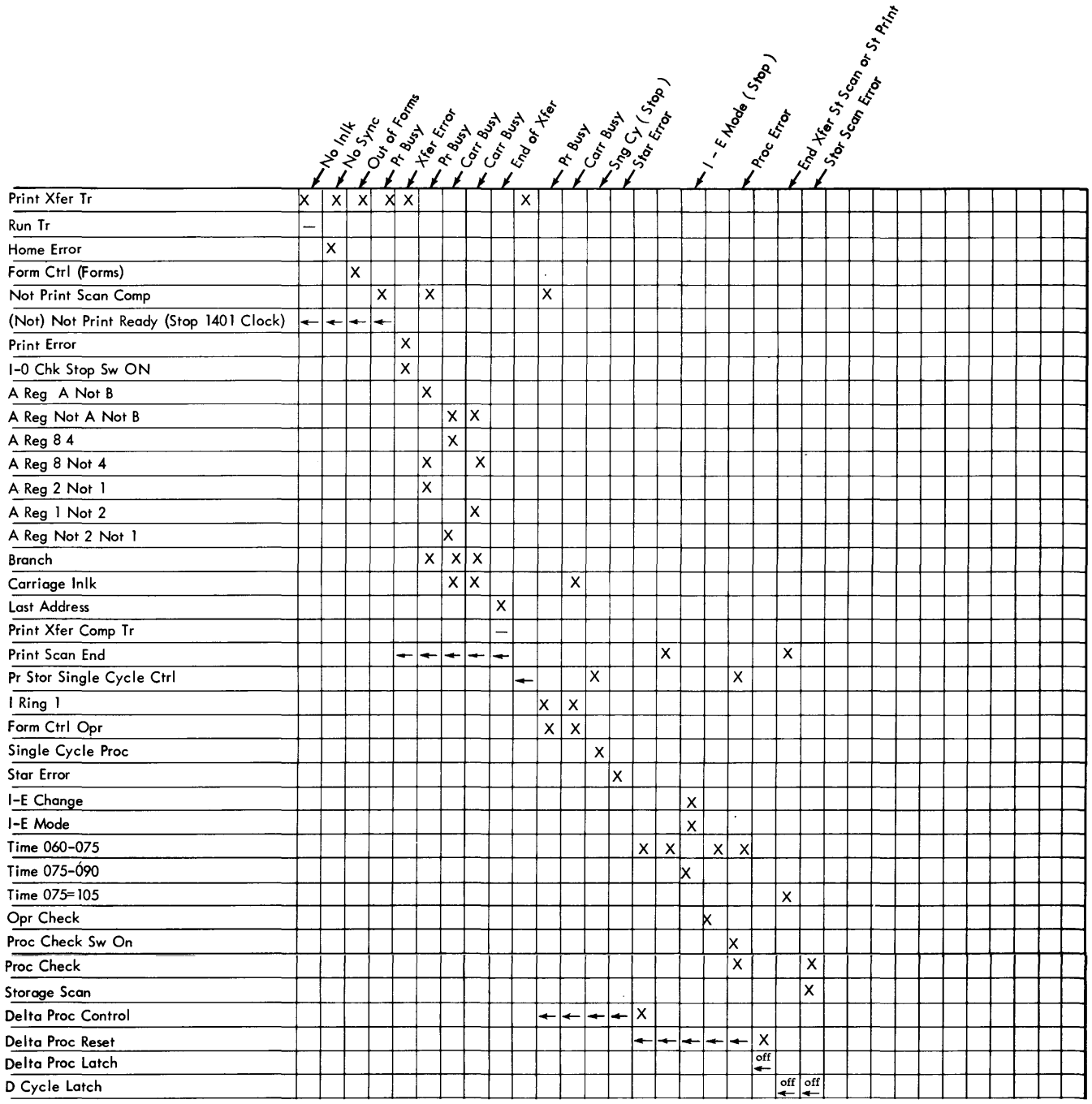


Figure 85. Deactivate 1401 Storage

4. a branch on printer error occurs while the printer is busy.
5. a branch on channel nine or twelve occurs while the carriage is busy.
6. a form control operation occurs while the printer or carriage is busy.

If a process check or print check occurs during transfer, the error condition can be reset. The 1401 can then be restarted by pressing the start key (if this action is desired). The 1401 will automatically restart when the interlocks listed above that involve a branch or form control operation have been deactivated. These interlocks will be deactivated as soon as the printer or carriage or both (for a form control operation) are no longer busy.

When the mode switch is set to STORAGE PRINT and the I-O check stop switch is ON, a print error occurring during transfer causes the D-CYCLE latch to turn OFF. The 1401 can be restarted by pressing the start key after the I-O check reset key has been pressed to reset the error, but the transfer will start over at location 201 (Figure 69). To keep the units and tens rings synchronized with 1401 addressing, the start reset key must be pressed before the start key. The best procedure would be to turn the I-O check stop switch OFF. Then print errors will not stop this operation,

When the mode switch is set to STORAGE SCAN, a print error causes a PROCESS CHECK. If a PROCESS CHECK occurs during transfer, the D-CYCLE latch is turned OFF. The 1401 can be restarted by pressing the start key after the I-O check reset key has been pressed to reset the error.

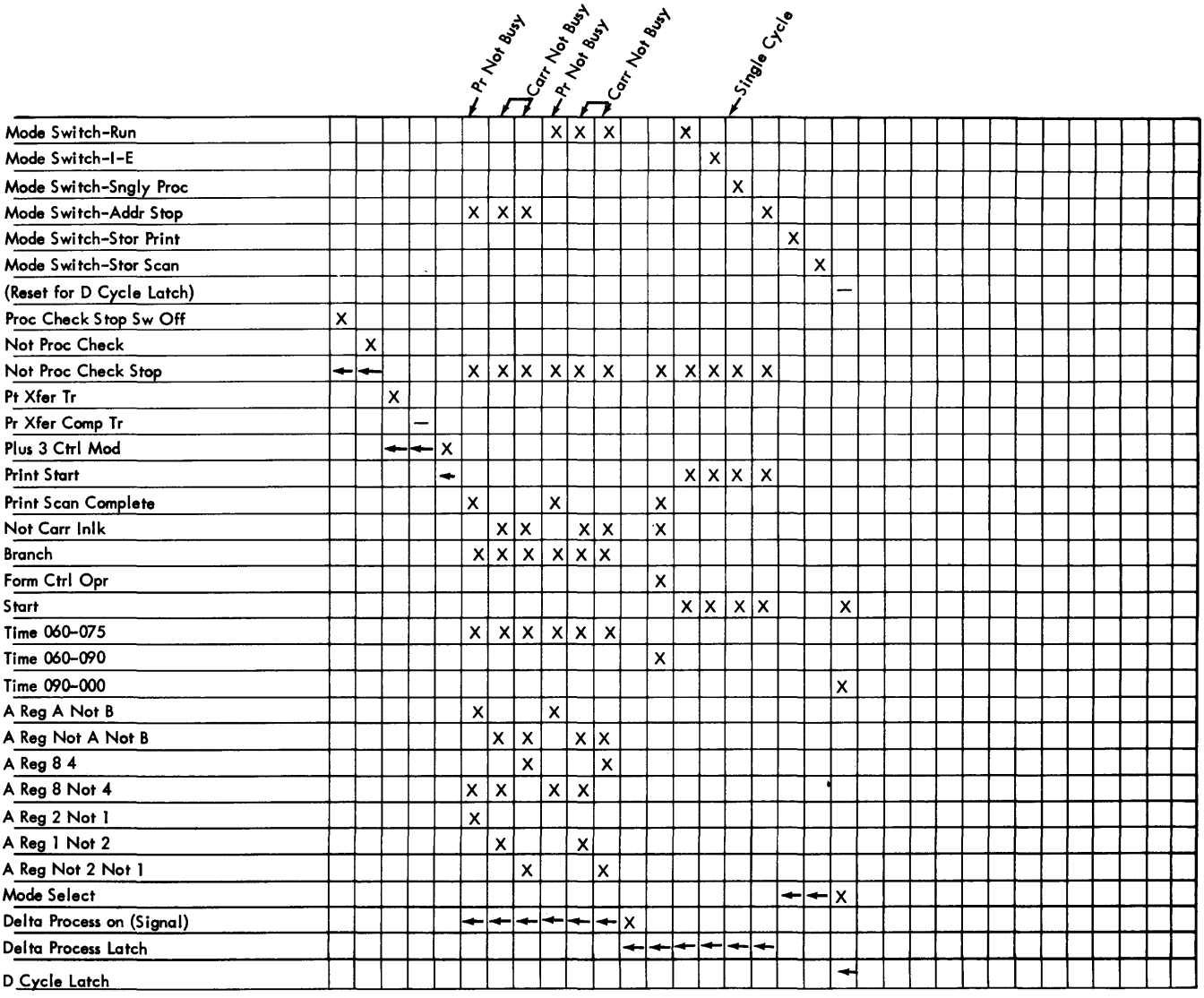


Figure 86. Activate 1401 Storage

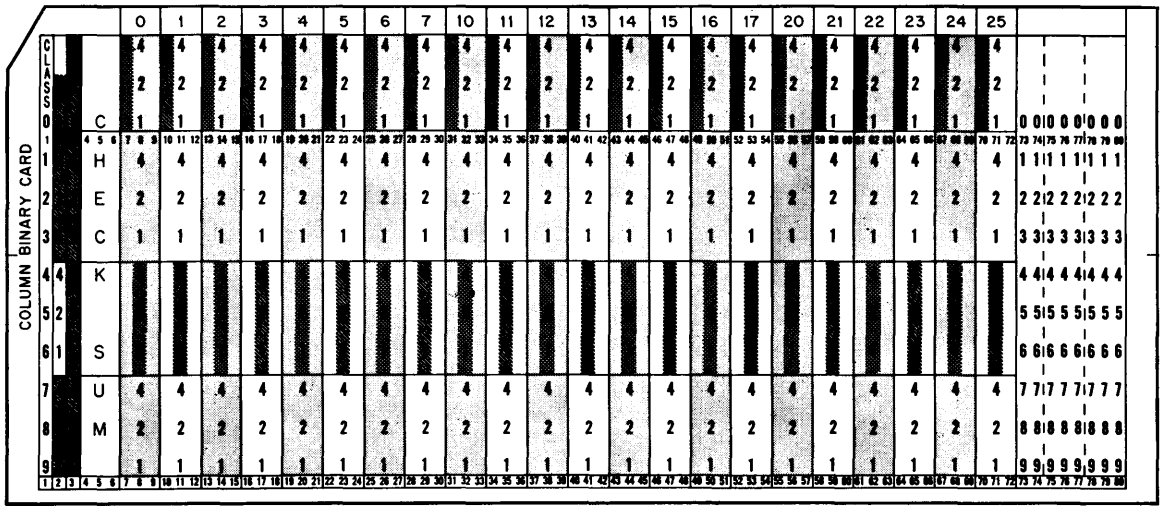


Figure 87. IBM Card — Column Binary

Column Binary

This optional feature provides for the conversion of IBM cards punched in a column binary form, Figure 87, to a binary tape form, Figure 88. The reverse process may also be accomplished. The feature may also be used in the 1401 to handle multiple significant digit coding from a single card column. The following operation codes are used in the conversion of column binary cards to binary tape, and in the conversion of binary tape to column binary cards.

Read Column Binary—1C

Operation Read Column Binary: The first objective to be accomplished in the conversion of binary cards to tape is to read an image of the binary card into the 1401 core storage. A feed cyle is taken which reads the information into the row bit cores as on a normal feed cycle. The scan of the row bit cores results in cores being set in storage as indicated by Figure 89. In addition to the storage locations shown in Figure 89, a BCD coded image of the card is read into the normal storage locations 001 to 080. If the punched column contains column binary data, locations 001 to 080 will contain *hash*. If the punched column contains BCD coded information, the contents of the 001-080 area may be used as input to the 1401. Conversely, any alphamerical character read into locations 401-480 or 501-580 will appear as *hash*.

Validity-check is suspended for the read column binary operation since all characters read are considered valid. The normal overlapping of the operation of the reader and punch is not possible when column binary information is being read because additional scan operations are necessary to read the data

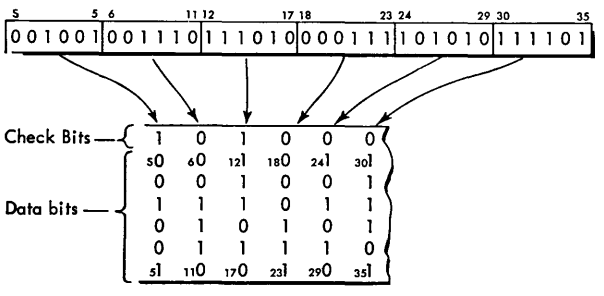


Figure 88. Magnetic Tape — Binary System

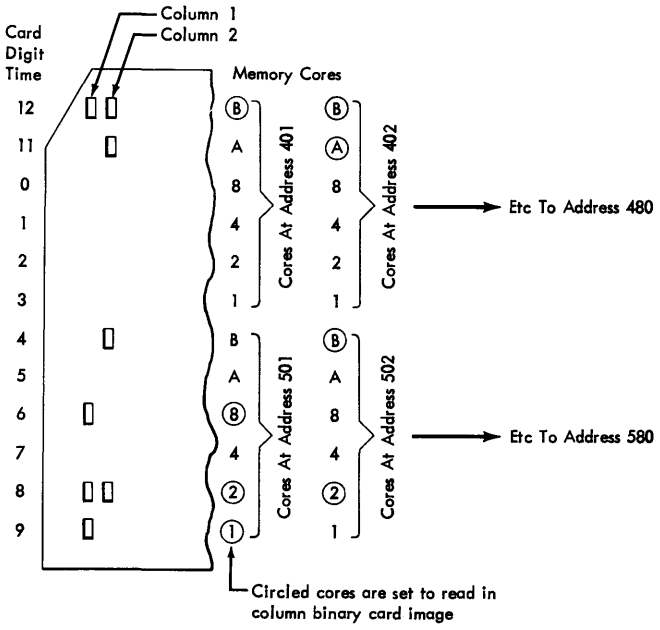


Figure 89. Column Binary Card to Core Format

from each card column into two different storage locations. The time left is not sufficient to allow for scan of both read and punch data.

The read operation is started in the normal manner. During 9 through 4 card reader time the 1401 scans the row bit cores 1 to 80 once for each card digit time. For example, if row bit core #1 is set at 9 time, the 1401 wil scan a BCD coded 9 into location 001. During the next following B-cycle, the 1-bit core in location 501 will be set. This process proceeds with the storage address register contents advancing as follows 001-501-002-502-003-503-004-504 . . . . . 579-080-580. Thus, 160 scans are needed for each card digit, with the same row bit core being scanned into two locations, one for the BCD code (if required) and one for the column binary image.

At 3-12 card reader time the process described above is continued except that the storage address register advances 001-401-002-402 . . . . . 479-080-480.

Data Flow for Read Column Binary: Each row bit core is sampled twice; first, in normal mode (see the following section, paragraph 2, for an explanation of normal and col bin mode) to enter the BCD code in location 0XX, and second, in column binary mode to enter the card image in location 5XX (or 4XX after

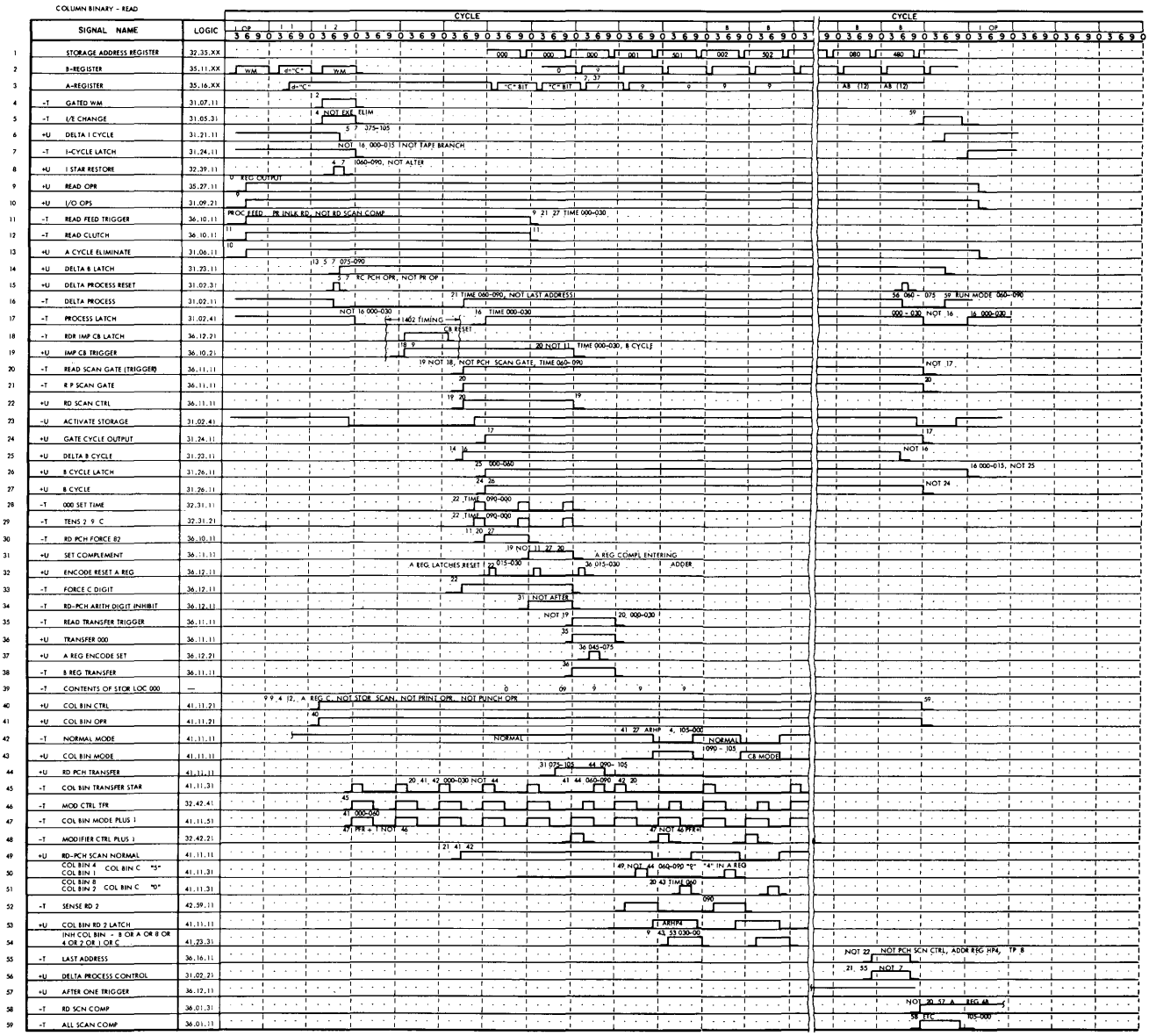


Figure 90. Column Binary Read Sequence

4 card time). The COL BIN RD 2 latch accomplishes the delay of the SENSE RD 2 signal to the end of the column binary mode B-cycle, see ILD 67.

During the normal mode B-cycle, the A-reg digit and zone inhibit lines are brought up in the usual manner if a SENSE RD 2 signal is present. This causes the BCD code in the A-register to be stored in the location 0XX.

In the next following Col Bin mode B-cycle, the ADDRESS REGISTER HUNDREDS POSITION 4 latch (ARHP 4) is forced ON so that either location 4XX or 5XX may be addressed depending upon what cycle point digit time the card reader is at. The STAR hundreds position 4 signal switches with SENSE RD 2 to turn ON the COL BIN RD 2 latch. This remains ON during the Col Bin mode B-cycle until the ARHP 4 line drops. Thus, the 4XX or 5XX location is addressed while the COL BIN RD 2 latch is ON. The COL BIN RD 2 signal is switched with COL BIN MODE and with the output of the A-register to provide inhibit storage impulses.

During the following example, refer to Figure 89 and ILD 67. Assume that the card reader is at 4 digit time and that the address register contains 502. If punching exists as shown in Figure 89, the B-core must be set in location 502. At 4 card time, the A-register contains a decimal 4 and the COL BIN RD 2 latch is on. A-Reg-12 or -4 switches with a gated RD 2 signal to provide an INHIBIT COL BIN-B signal. Similarly, at 12 card digit time, the COL BIN RD 2 signal is up, the A-register is set to decimal 4 and the INHIBIT COL BIN-B line comes up to enter a B-bit in location 402.

Circuits for Read Column Binary, Figure 90 and ILD 67: The following objectives are accomplished in addition to those normally required on a read operation:

1. Identify the Column Binary operation at 12 time. The C-character in the d-position of the instruction causes the turn ON of the COL BIN OPR latch. READ OPR comes ON in the usual manner.

2. Signal Normal Mode and Column Binary Mode. On alternate B-cycles the STAR must change from 0XX to a 4XX or 5XX address depending upon whether the card is at 9-4 time or 3-12 time. This is accomplished by means of a column binary mode trigger which is flipped on alternate B-cycles. ILD 67 shows how the address register hundreds position 4 (ARHP 4) impulse flips the trigger to the COL BIN MODE position.

3. Generate Hundreds Position 5 or 4 to Address Register. When the column binary mode trigger is in the normal position and the card is at 9-4 time, 060-090 impulses are OR'd to the modifier output to cause the turn ON of the 4, 1 and C B STAR latches and subsequently the STAR latches. The COL BIN 4, 1 and C-gates shown on ILD 67 accomplish this on each normal cycle when the card is at 9-4 time. COL BIN 4 is generated similarly from 3-12 card time.

4. Generate Hundreds Position 0 to Address Register. When the column binary mode trigger is in the column binary position, 060-090 impulses are OR'd to the modifier output to turn on the 8, 2 and C B STAR latches, and subsequently the STAR latches. The COL BIN 8, 2 and C gates, ILD 67, accomplish this on each column binary mode cycle.

5. Modify Units Position of STAR. During COL BIN mode the COL BIN MODE PLUS-1 signal occurs on each B-cycle as long as the COL BIN OPR latch is ON. This conditions one input to the MOD CTRL PLUS-1 switching. Recall that the units position of the storage address register is modified at 000-030 time. Note from ILD 67 that the COL BIN TRANSFER STAR signal is OR'd to become MOD CTRL TRANSFER, and then inverted to hold MOD CTRL PLUS-1 OFF. COL BIN TRANSFER STAR is controlled during normal mode B-cycles only, as shown in ILD 67. Thus, during column binary mode cycles MOD CTRL TRANSFER is down and a MOD CTRL PLUS-1 is up. This results in the units position of the address register contents being increased by one in preparation for the next following 0XX address.

6. Control Generation of STAR Hundreds Position 4 or 5. A study of the BCD code shows that digits 9 through 4 are coded in the A-register using either a 4- or an 8-bit. Digits 3 through 12 use neither 4- or 8-bits. ILD 67 shows how the A-REG NOT 84 signal is switched with a time 060-090 gate to cause a COL BIN 4 to be OR'd to the modifier output. This causes the 4XX addresses to be generated during 3-12 card time. Similarly, the (NOT) A-REG NOT 84 signal is inverted at 9-4 time to cause a 5 to be OR'd to the modifier output. Note that both COL BIN 4 and COL BIN 5 signals occur during normal mode time in preparation for the next following COL BIN MODE B-CYCLE.

Punch Column Binary—4C

Operation Punch Column Binary: The image of the column binary card placed in 1401 storage from either a tape or a card read operation may be punched out in a column binary format. Figure 89 indicates the objective which must be accomplished. For example, the A-core set in address 402 must be punched out as an *11* in column 2 of the card. To accomplish this, 160 scans are made at each punch digit time. The storage address register contents must be altered so that the 1401 storage is addressed in the following order during each card digit time.

At 12-3 card time:
401-101, 402-102, 403-103 . . . 479-179, 480-180
At 4-9 card time:
501-101, 502-102, 503-103 . . . 579-179, 580-180

Addresses 101 to 180 are developed so that the check planes may be addressed for the hole bit check count. The normal comparison of the card digit time in the A-register with the BCD code from storage is suspended for the punch column binary operation. When the addresses 401-480 and 501-580 are being scanned, the punch magnets are set up comparing the BCD representation of the card digit time with the storage output (B-register) using the circuit illustrated in ILD 67.

All validity checking is suspended during the punch column binary operation since all characters are considered valid. The overlapping of reader and punch operations is not possible when column binary information is being punched because 160 scan operations are needed to address and set the check plane cores, and to address the binary data and read it out into the punch magnets. Sufficient time does not remain for the scan of both read and punch data.

Data Flow for Punch Column Binary: The scan of core storage during each card digit time takes place in a manner similar to the read operation. The CB mode trigger alternates between column binary and normal mode on every other B-cycle. During normal mode the check planes are addressed (101 to 180), and during column binary mode the locations 401-480 and 501-580 are addressed. In normal mode the hole count check planes are operated in the usual manner. In column binary mode, the value of the A-register is changed with each change in punch cycle point timing in the order 12, 11, 0, 1 . . . 8, 9. This occurs in the same manner as on a punch operation. An examination of Figure 89 shows that a B-core set in any of the addresses 401-480 must result in the punching of a 12-hole. Also, any B-core set in any of the addresses 501 to 580 must result in the punching of a 4-hole. The circuit shown in ILD 67 accomplishes this punching. At either 12 (A-Reg AB) or 4 (A-Reg 8421) card time, a B-bit from storage (B-Reg-B) results in a COL BIN PCH DECODE signal. This switches with the decoded units and tens position of the address register to select a punch magnet driver in the usual manner.

Circuits for Punch Column Binary, Figure 91 and ILD 67: The following objectives are accomplished in addition to those normally required on a punch operation:

1. Identify the column binary operation at 12 time. The C-character in the d-position of the instruction causes the COL BIN OPR latch to turn ON. The PUNCH OPR signal is developed in the usual manner.
2. Signal Normal Mode and Column Binary Mode. The location addressed in storage changes from 4XX to 1XX or from 5XX to 1XX on alternate B-cycles as previously discussed. The column binary mode trigger controls the generation of the hundreds position 4, 5, or 1. The column binary mode trigger is flipped to the column binary mode on alternate B-cycles by hundreds position 4-bit address register output. The hundreds position 4-bit is turned on when the column binary mode trigger is in the normal position.
3. Generate Hundreds Position 4 or 5 to Address Register. When the column binary mode trigger is in the normal position and the card at 12-3 time, 060-090 impulses are OR'd to the modifier output to cause the turn ON of the 4-bit B-STAR latches and subsequently the STAR latches. The COL BIN-4 gate is shown in ILD 67. COL BIN-4 and -1 are generated similarly during 4-9 time.
4. Generate Hundreds Position-1 to Address Register. When the column binary mode trigger is in the column binary mode position, 060-090 impulses are OR'd to the modifier output to turn ON the 1-bit B-STAR latch and subsequently the 1-STAR latch. The COL BIN-1 gate does this on each column binary mode cycle, ILD 67.
5. Modify the Units Position of STAR During Normal Mode. The COL BIN MODE PLUS-1 signal occurs on each B-cycle as long as the COL BIN OPR latch is ON. This conditions one input to the MOD CTRL PLUS-1 switching as shown on ILD 67. Recall that the units position of the storage address register is modified at 000-030 time. ILD 67 shows that the COL BIN TRANSFER STAR signal is OR'd to become MOD CTRL TRANSFER and then inverted to hold MOD CTRL PLUS-1 OFF. COL BIN TRANS STAR is controlled during column binary mode only on a punch operation. This holds MOD CTRL PLUS-1 OFF during column binary mode B-cycles, and allows the modification of the units position of the STAR during normal mode B-cycles. Thus on each normal mode cycle, the STAR units digit is increased by one in preparation for the next following column binary mode cycle.
6. Control Generation of STAR Hundreds Position 4 or 5. A study of the BCD code shows that digits 9 through 4 are coded in the A-register using either a 4- or an 8-bit. Digits 3 through 12 use neither 4- or 8-bits. ILD 67 shows how the A-REG NOT 84 signal is switched with a time 060-090 gate to cause a COL BIN 4 to be OR'd to the modifier output. This causes the 4XX addresses to be generated during 3-12 card time. Similarly, the (NOT) A-REG NOT 84 signal is inverted at 9-4 time to cause the 5 to be OR'd to the modifier output. Note that both COL BIN 4 and COL BIN 5 signals occur during normal mode time in preparation for the next following column binary mode B-cycle.

Read Column Binary and Transfer Program—1 (AAA) C
Punch Column Binary and Transfer Program—4 (AAA) C

These operation codes function in the same manner as punch and read column binary except that at the end of the operation, the next instruction is taken from the location (AAA) instead of from the next instruction address in sequence.

A COLUMN BINARY PROGRAM SKIP signal is developed (ILD 67) by switching COL BIN OPR with gated WM and I-RING-5. This turns ON PROGRAM SKIP which turns ON I-A STAR GATE AND SET I-OP (ILD 14). At the end of the read or punch operation, when DELTA I-CYCLE comes back ON, the I-RING OP latch is turned ON and the I-A STAR GATE causes the address (AAA) to be transferred into the STAR for the I-Op cycle.

Write Tape Binary—M (% B 1) (AAA) W
Read Tape Binary—M (% B 1) (AAA) R

The column binary information must be written on tape or read from tape with an odd-redundancy parity bit. This is accomplished by using a modified tape control instruction with a B-character in the third position of the instruction. This causes the 1401 to read or write tape in an odd-redundancy mode even though the manual switch setting is in the normal even-redundancy mode position.

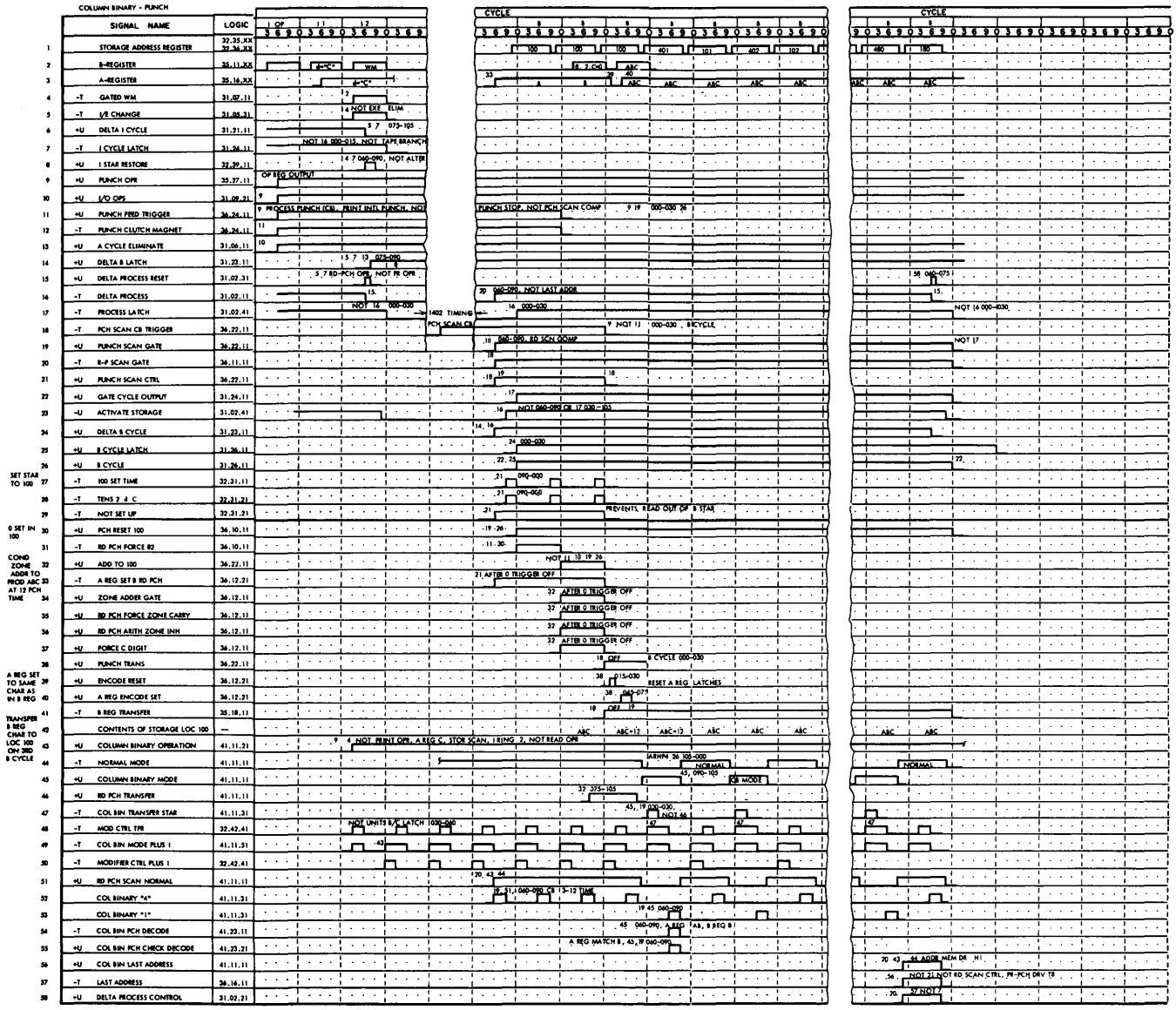


Figure 91. Column Binary Punch Sequence

The forcing of odd-redundancy takes place on ILD 80 where A-Reg-B is switched with I-Ring-2, Unit Select and time 075-105 to turn the REDUNDANCY latch to the odd position.

Move and Unscramble Column Binary—M(AAA)(BBB)A

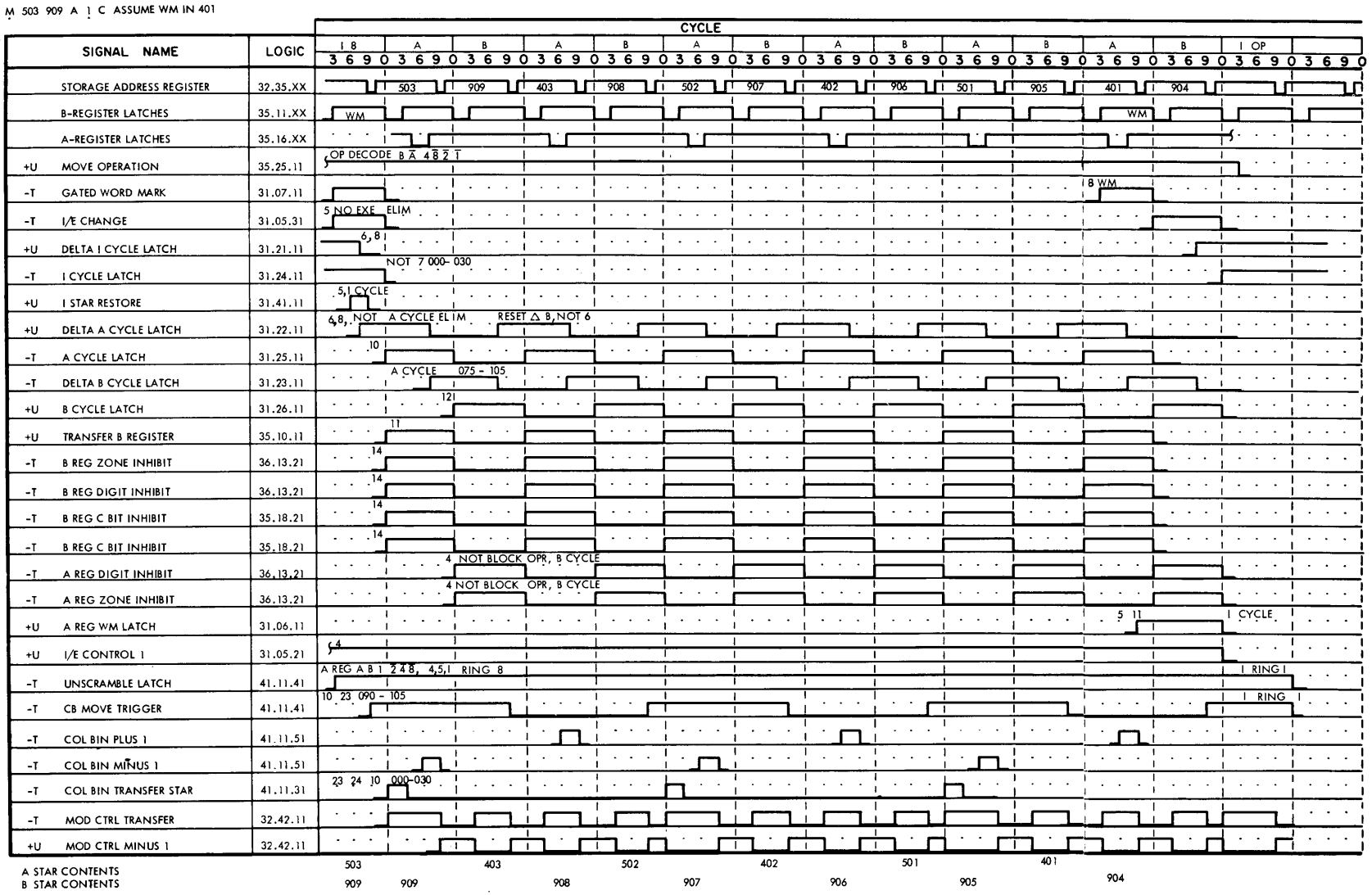
This instruction causes the data in the A-field to be stored in the B-field as with a normal move operation. Address (AAA) must be between 501 and 580, however (AAA) will normally contain either 572 or 580 depending upon whether the card held 72 or 80 columns of column binary information. Address (BBB) can be any storage address.

The unscramble column binary operation code causes the column binary information arranged in card image format, Figure 89, to be rearranged in storage in binary tape format, Figure 88. To accomplish this, a series of alternate A- and B-cycles are performed. The A-field characters are brought to the A-register on A-cycles and stored in the B-field on B-cycles. If the (AAA) field of the instruction is (503), the following addresses are generated for use in the STAR during A-cycles: 503-403-502-402-501-401. If the (BBB) field of the instruction is 909, the following addresses are generated for use in the STAR during B-cycles: 909-908-907-906-905-904. See Figure 92.

In addition to the objectives accomplished on a normal move operation, the following modifications must be made to the storage address register (Figure 92):

1st, 3rd, 5th . . . A-Cycles

Figure 92. Move and Unscramble Column Binary Sequence



1. The hundreds position 5 of the STAR is reduced by one by the modifier circuits and transferred to the A-STAR.
2. The units and tens positions of the address register are transferred to the A-STAR without modification.

1st and Succeeding B-Cycles

1. No modification of the hundreds position takes place (except by the borrow/carry circuits).
2. The units position of the STAR is reduced by one in the usual manner.

2nd, 4th, 6th . . . A-Cycles

1. The hundreds position 4 of the STAR is increased by one by the modifier circuits and transferred to the A-STAR in preparation for the next A-cycle.
2. The units position of the STAR is decreased by one by the modifier in preparation for the next A-cycle.

The pattern outlined above is repeated until a word-mark (usually in 401) is sensed. The operation ends and a new instruction is read into the Op register.

Circuits for Move and Unscramble Column Binary, Figure 92: The COL BIN MOVE trigger (ILD 68) is turned ON just before the 1st, 3rd, 5th . . . A-cycles and turned OFF just before the 2nd, 4th, 6th . . . A-cycles. During the A-cycle in which the COL BIN MOVE trigger is ON, the hundreds position of the STAR is reduced by one, and the tens and units positions are transferred to the A-STAR without modification. The following gates are developed under control of the Move trigger to accomplish this:

1. Col Bin Minus-1 at 060-090 during A-cycles 1, 3, 5, 7, etc. causes the reduction of the hundreds position of STAR by one.
2. Col Bin Transfer STAR at 000-030 during A-cycles 1, 3, 5, 7, etc. causes the transfer of the units position of the STAR without modification.

During the A-cycle in which the COL BIN MOVE trigger is OFF, the hundreds position of the STAR is increased by one, and the units position is decreased by one. The following gates are developed during the 2nd, 4th, 6th, etc. A-cycles:

1. Col Bin Plus-1 at 060-090 during A-cycles 2, 4, 6, 8 etc. causes the increase of the hundreds position of STAR by one.
2. Mod Ctrl Minus-1 at 000-030 causes the reduction of the units position of the STAR by one.

During each B-cycle, the units position of the STAR is reduced by one at 000-030 by the Mod Ctrl Minus-1 signal. The tens and hundreds position may also be modified in the usual manner under control of the borrow/carry circuits.

Move and Scramble Column Binary — M (AAA) (BBB) B

The move and scramble instruction causes the data in the (AAA) field to be moved to the B-field as with a normal move operation. Address (AAA) may be any storage address. Address (BBB) must be between 501 and 580. However, (BBB) will normally contain either 572 or 580 depending upon whether the card held 72 or 80 columns of column binary information.

The scramble column operation causes the binary information arranged in tape image format, Figure 88, to be arranged in storage in column binary card format, Figure 89. A series of alternate A- and B-cycles are performed as in a normal move operation. The A-field characters are brought to the A-register on A-cycles and stored in the B-field on B-cycles. If the (BBB) field of the instruction is 503, the following addresses are generated for use in the STAR during B-cycles: 503-403-502-402-501-401, assuming a word-mark in 401. If the (AAA) field of the instruction is (909), the addresses 909, 908, 907, 906, 905, 904 are generated for use in the STAR during A-cycles (see Figure 93).

The usual move operation objectives are accomplished with a scramble column binary instruction. In addition, the following modifications are made to the storage address register (see Figure 93):

1st and Succeeding A-cycles

1. No modification of the hundreds position takes place except by the borrow/carry circuits.
2. The units position of the STAR is reduced by one in the usual manner.

1st, 3rd, 5th . . . B-cycles

1. The hundreds position 5 of the STAR is reduced by one by the modifier circuits and transferred to the B-STAR.
2. The units and tens positions of the address register are transferred to the B-STAR without modification.

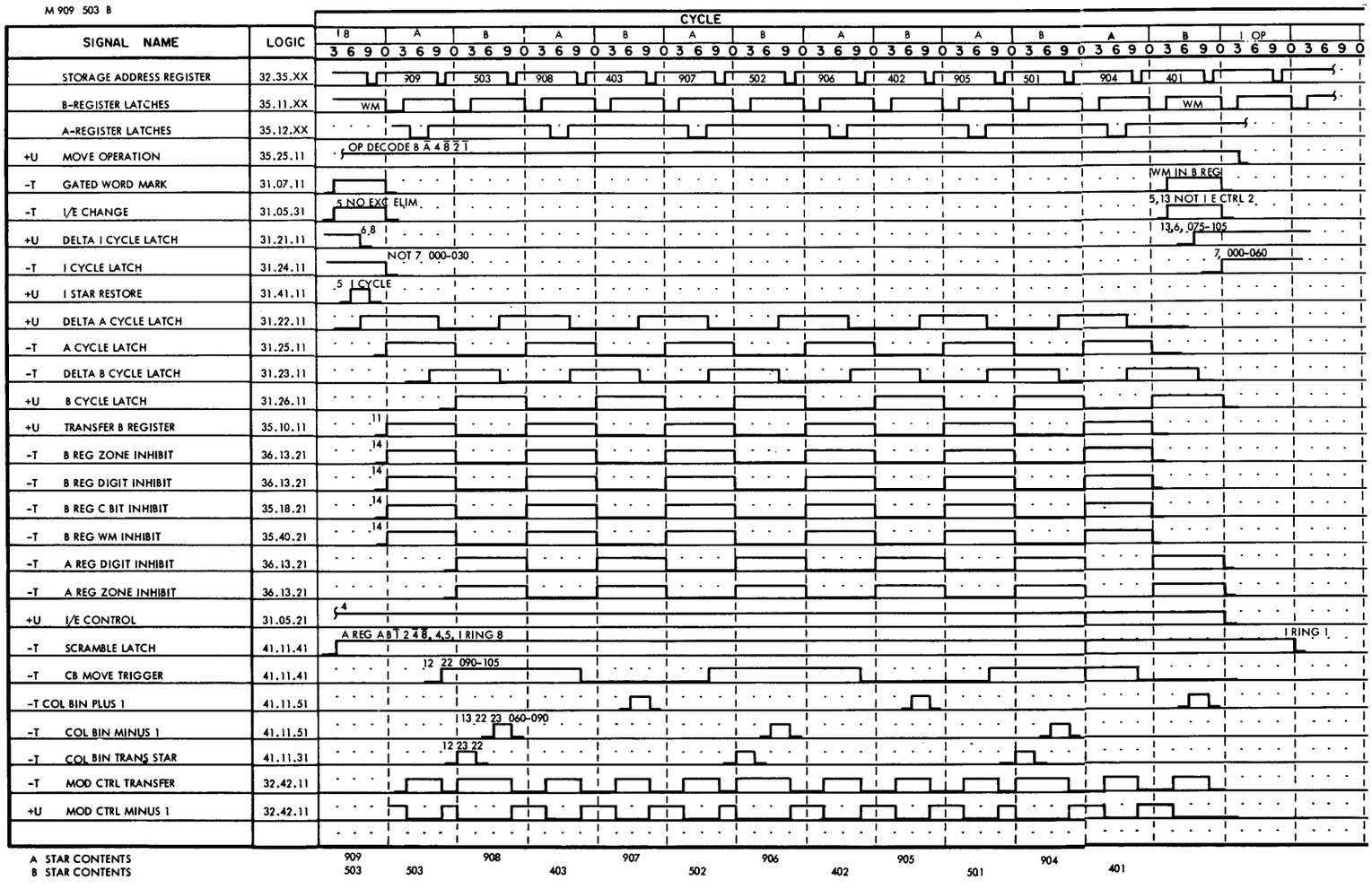
2nd, 4th, 6th . . . B-cycles

1. The hundreds position 4 of the STAR is increased by one by the modifier circuits, and transferred to the B-STAR in preparation for the next B-cycle.
2. The units position of the STAR is decreased by one by the modifier in preparation for the next B-cycle.

The pattern outlined above is repeated until a word-mark (usually in 401) is sensed.

Circuits for Move and Scramble Column Binary: The COL BIN MOVE trigger is turned ON at 090 time before the 1st, 3rd, 5th . . . B-cycles (ILD 68). While it is ON during these B-cycles, it controls the objectives listed

Figure 93. Move and Scramble Column Binary Sequence



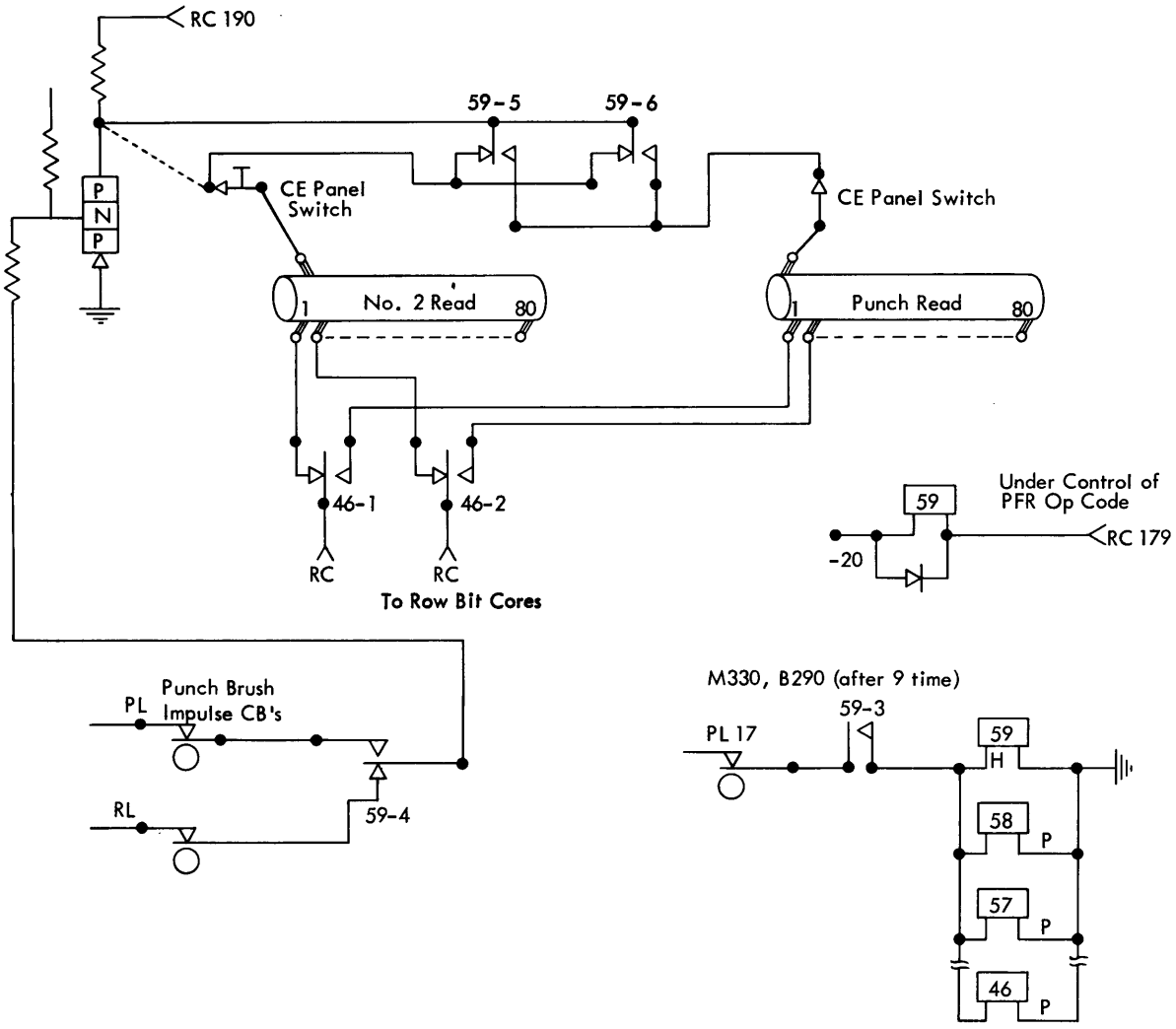


Figure 94. Relay Switching for Punch Feed Read

above for the odd-numbered B-cycles. The following gates are developed under the control of the move trigger to accomplish this:

1. Col Bin Minus-1 at 060-090 during B-cycles 1, 3, 5, 7, etc. causes the reduction of the hundreds position of the STAR by one.
2. Col Bin Transfer STAR at 000-030 during B-cycles 1, 3, 5, 7 causes the transfer of the units position of the STAR without modification.

During the B-cycle in which the COL BIN MOVE trigger is OFF, the hundreds position of the STAR is increased by one and the units position is decreased by one. The following gates are developed during the 2nd, 4th, 6th . . . B-cycles.

1. Col Bin Plus-1 at 060-099 during B-cycles 2, 4, 6, 8, etc. causes the increase of the hundreds position of the STAR by one.
2. Mod Ctrl Minus-1 at 000-030 causes the reduction of the units position by one.

Note that the tens and hundreds positions may also be modified in the usual manner under control of the borrow/carry circuits.

Punch Feed Read (PFR)

The punch feed read optional feature adds an additional set of 80 read brushes to the blank station ahead of the punches in the punch feed. With this feature, data may read from a card, and the results of a calculation may be punched into the same card on the following 1402 punch cycle. Normal operation of the 1401 machine is retained with PFR operation. The only restriction is that no combination operation code involving the use of the read feed may be programmed in conjunction with PFR Op code, 4R. For example, the instruction 5R will result in a read and punch (5) operation. The R, identifying a PFR operation, will be ignored.

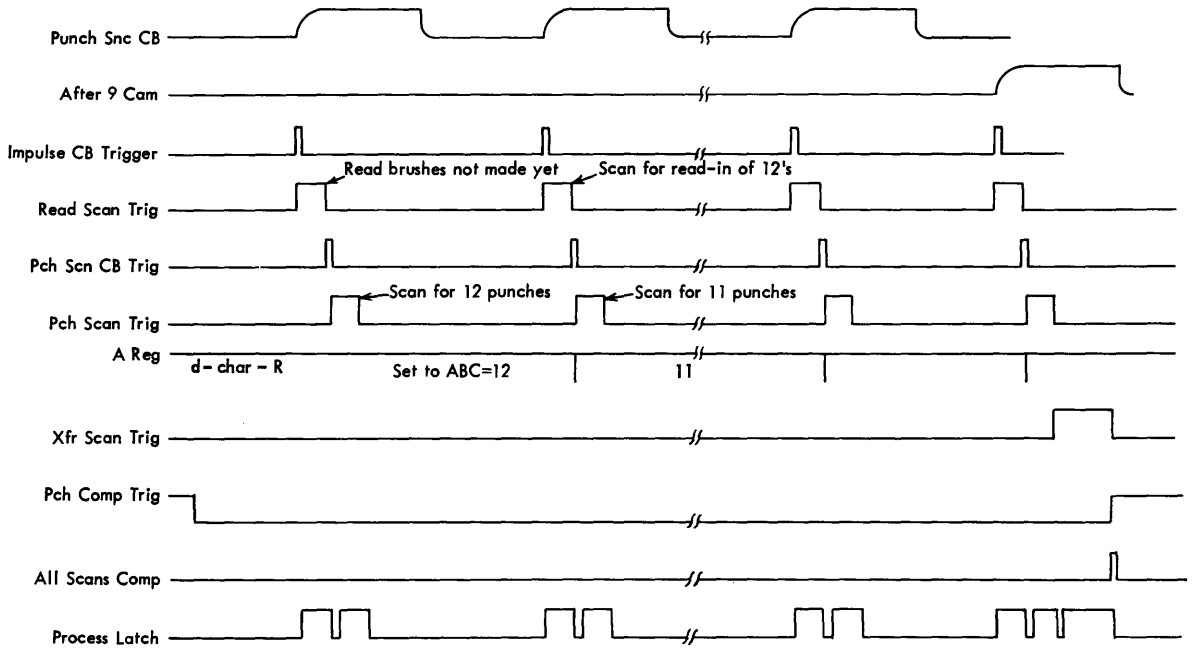


Figure 95. Punch Feed Read Timing Scan Relationship

Punch Feed Read Operation—4R

The PFR Op trigger sets with gated WM of I-cycles as shown on ILD 64. It remains ON to control the entire operation, and is switched OFF at the next I-Op time.

PFR DATA FLOW

The 80 PFR brushes are switched through relay points, as shown in Figure 94, so that data read at the punch read brushes enters the read area of main storage in the same manner as data from the 2nd read brushes on a read operation. The punch clutch is energized in the normal manner as on an ordinary punch operation. When the cam impulse picking the punch clutch breaks, NOT PROCESS PUNCH comes ON and causes a transfer read brush signal. This picks relay 59 in the 1402. Relay 59 holds along with 13 other 6-position relays. The points of these relays transfer common brushes and the 80 punch read brushes to the 2nd read row bit cores. Also, the punch brush impulse CB's are transferred to feed the punch read brush common.

PFR SCAN SEQUENCE

The circuitry required to control the scans on a PFR operation is the same as that used on a normal read and punch operation, with some changes to force the correct sequence of read and punch scans. The first make of the punch scan CB is during the cycle point prior to 12-punch and read time. During the following cycle point the 12-holes are read at the PFR brushes, and the 12-holes are punched in the preceding card. Just after the row bit cores, set from 12 holes, are scanned during 12-punch time, a punch scan must take place to set the punch magnets for 11-punch time. Figure 95 shows this sequence of operation.

The punch scan CB starts the sequence by timing on the impulse CB trigger instead of the usual punch scan CB trigger. This causes a read scan even though the PFR brushes have not made for 12-time. When the read scan gate goes off, the punch scan CB trigger is turned ON. This results in a punch scan which operates in a normal manner, scanning the punch area of main storage out to the punch magnet drivers. The sequence of read scan followed by punch scan continues at each make of the punch scan CB or the after 9-cam.

Recall that the A-register is used to indicate cycle point timing, and that the scan for specific punch read row bits takes place one cycle point apart from the scan for the same characters in the punch area of storage. Thus, the A-register must be set to 12 (AB) for the punch scan just before 12 cycle point time and must also be set to 12 for the read scan during 12 cycle point time. Figure 95 shows how the A-register is set at the beginning of each punch scan, remains the same for the next read scan, and is changed to the next cycle point indication for the following punch scan. The A-register is controlled with the usual A-register setup circuitry by blocking the A-register setup by read scan, setting it only with punch scan and retaining the character until the next punch scan.

PFR READ ENCODER

Since the card is being fed 12-edge first instead of 9-edge first, the read encoder normally used is bypassed by a special PFR encoder shown on ILD 64.

When an output signal arrives from the 2nd read core row buffer (RD 2), the A-register and B-register digit and zone inhibit lines are brought up under the restrictions listed below:

1. At Punch-Scan-Before-11 time the A-register AB line prevents the B-register digit and zone inhibit gates so that old data is not regenerated. The FORCE C-DIGIT C-ZONE line is brought up to substitute C for the old data.
2. With card code 12-0 and card code 11-0, the RD PCH FORCE 82 line is brought up at 0 time to complete the BCD coding. Because the A-register contains A not B at this time, the 82 must be forced. In addition, the A-bit in the A-register is prevented from entering read storage by holding down the A-register zone inhibit line.
3. When a 0-only is read at 9 cycle point time, assuming no 9 read, the encode 82 line is brought up to force in a BCD ZERO RD-PCH FORCE 82. At the same time, the B-register zone and digit inhibit lines are held down to prevent the A in the B-register from entering read storage.

PFR Checking

The information read at the punch read brushes is entered in the read area of storage. The hole count for each column read is entered in a set of two check planes above the read area (for example: XU and XL). When the same card is punched, the punch check decode signal causes setting of the hole count for the holes punched in the corresponding check planes above the punch area of general storage (XU and XL). If the card is to be checked after punching, the hole count made at the punch read brushes and entered over the read area must be transferred to the corresponding set of check planes (XU and XL) above the punch area. This is accomplished just after the last scan as the card passes the read brushes.

A transfer scan operation controls the movement of check data from the set of check planes above the read area to the corresponding set of planes above the punch area. This special scan is initiated after the end of the last punch scan and during after-9-cam time. The operation is controlled by the transfer scan trigger shown on Figure 96. The B-STAR is controlled during the transfer scan to address first the read area of storage and then the punch area in the sequence 001, 101, 002, 102 . . . 080, 180. During each storage cycle in which the read area is addressed, all four check planes and the read area are read out.

The check planes to be transferred are also read into the PFR check plane register one position at a time. One pair of the four check plane registers may be set during a cycle in which the read location is addressed. The check plane register retains the bit until the end of the next storage cycle in which the punch area is addressed. During the punch area addressed storage cycle, the presence of a bit in one of the check plane registers gates with the plane select line to transfer the core setting to the corresponding plane over the punch area. The punch cycle trigger determines the check planes selected over the punch area.

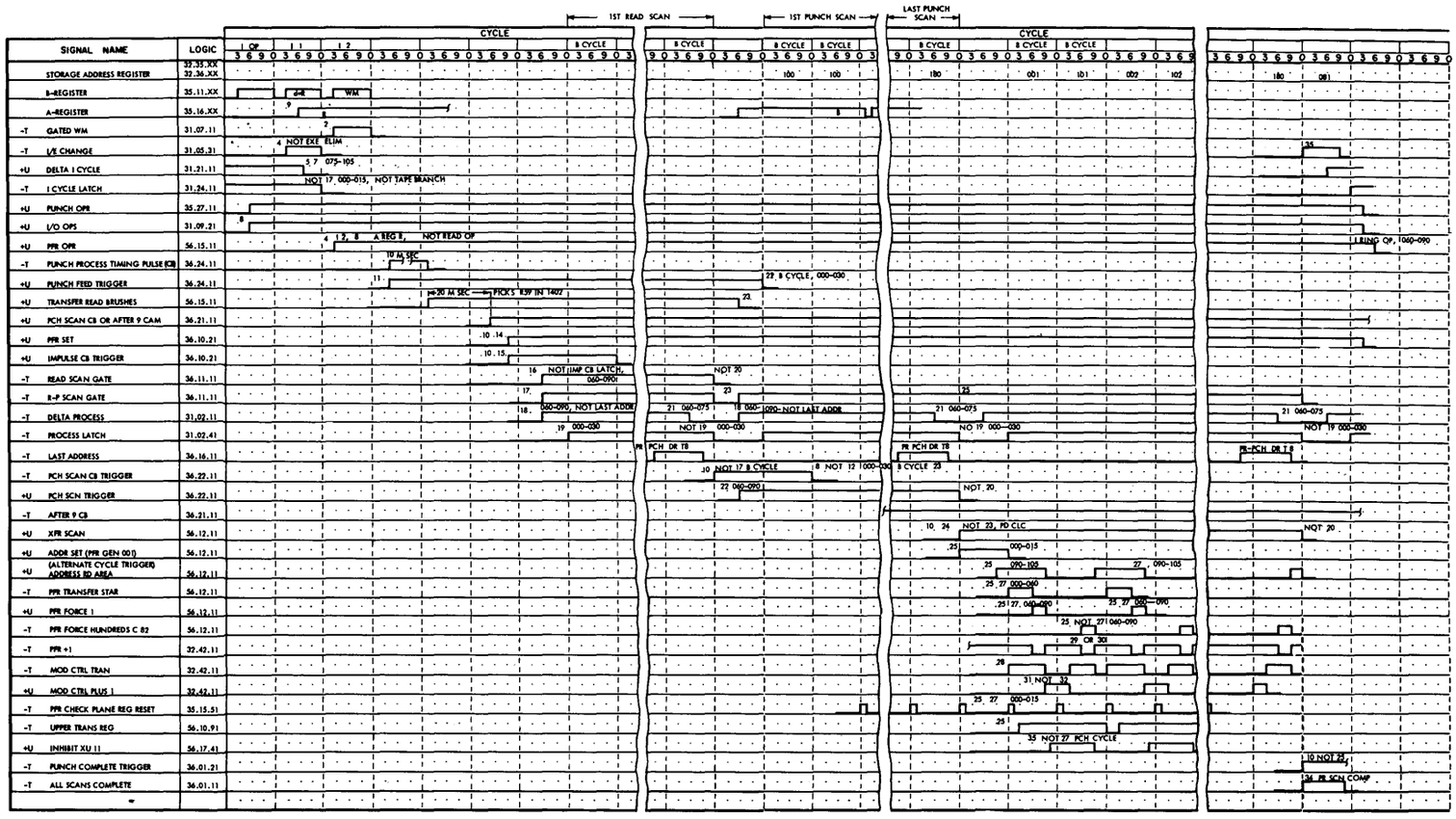
Modifier Controls: As noted above, the STAR must be controlled so that the addresses 001, 101, 002, 102 . . . 080, 180 are generated. This is accomplished as follows (see Figure 96):

1. The address set trigger is turned ON by transfer scan, remains ON for one cycle, and is turned OFF. This signal is sent to the auto scan setup circuits as PFR Gen 001 to set the STAR to 001.
2. The alternate cycle trigger is flipped to its opposite state on every storage cycle. The trigger outputs (address read area and address punch area on alternate cycles) control the modification of the STAR.
3. The PFR FORCE-1 signal, during each read addressed cycle, sets the hundreds position of the B-STAR to 1 in preparation for the next following punch addressed cycle.
4. The PFR FORCE HUNDREDS 2, 8, C signals, during each punch addressed cycle, sets the hundreds position of the B-STAR to 0 in preparation for the next following read addressed cycle.
5. The PFR TRANS STAR signal, during each read addressed cycle, causes the units and tens position of the STAR to be transferred to the B-STAR without modification.
6. MOD CTRL PLUS-1 is on at 000-030 time of each punch area addressed cycle to modify the units position of the STAR by 1 in preparation for the next following read addressed B-cycle.

The normal regeneration path is blocked during the storage cycle in which the punch area is addressed to prevent the regeneration of bits from previous data. When the punch cycle trigger is OFF, Figure 97 indicates that the Y planes are used. *Not punch cycle, transfer scan* and *address punch area* are switched together (ILD 64), and then inverted to hold down the normal regeneration path for the Y planes. A similar blocking of regeneration is accomplished at 12-time when reading into the read area.

Plane Selection: The plane entered from the PFR brushes is always the one which was last entered from the 2nd read brushes. This is accomplished since the

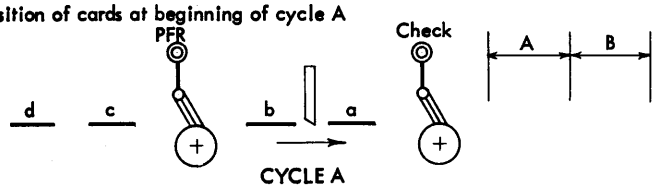
Figure 96. Punch Feed Read Sequence



		PFR READ BRUSHES	PUNCH CHECK DECODE	PUNCH CHECK BRUSHES	1ST READ BRUSHES	2ND READ BRUSHES
Read Cycle Trigger On (Y Gate)		X Planes (Over Read Area)			Y Planes	X Planes
Read Cycle Trigger Off (X Gate)		Y Planes			X Planes	Y Planes
Punch Cycle Trigger On (Y Gate)			Y Planes	X Planes		
Punch Cycle Trigger Off (X Gate)			X Planes	Y Planes		
				Check Cycle		Check Cycle
One Punch	Upper Single Bit	0-1	0-1	1-0	0-1	1-0
	Lower Binary Bit	0-1	0-1	1-0	0-1	1-0
Two Punches	Upper Single Bit	0-1	0-1	1-0	0-1	1-0
	Lower Binary Bit	0-1-0	0-1-0	0-1-0	0-1-0	0-1-0
Three Punches	Upper Single Bit	0-1	0-1	1-0	0-1	1-0
	Lower Binary Bit	0-1-0-1	0-1-0-1	1-0-1-0	0-1-0-1	1-0-1-0

Figure 97. Reader Punch Brush-Check Plane Selection

Position of cards at beginning of cycle A



1 Read card c at PFR, enter check planes X over read area.

2 Punch card b, enter check planes Y over punch area.

3 Check read card a into check planes X above punch area.

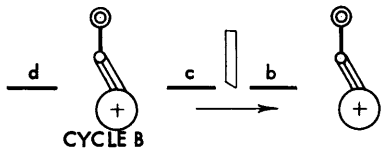
4 After (3) is complete, transfer scan moves X over read (card c) to X over punch.

Circuit Outline for Data Flow. See Figure 7.7-2

Assume: Read Cycle Trigger On
Punch Cycle Trigger On

1	2	3	4
Y Gate	Y Gate	Y Gate	B Reg Ch Pl XU 11
RD 2 Signal	Pch Check Decode	Pch Signal	Tran Scan On
Row bit X	Row bit Y	Row bit X	PFR Ch Pl Reg
Rd Scan Gate	Regen YU	Regen XU	Upper Tran Reg
Regen XU	Regen XU	Regen XL	Pch Cycle Trig On
Regen XL			Address Pch Area
			Inhibit XU 11

Position of cards at beginning of cycle B



1 Punch card c, enter check planes X over punch area. All check cover now set from both PFR read brushes and punch check decode.

2 Check read card b into check planes Y over punch area.

3 Read card d at PFR, enter check planes X over read area.

4 After 2 is complete, transfer scan moves X over read (card d) to Y over punch.

Circuit Outline for Data Flow. See Figure 7.7-2

Assume: Read Cycle Trigger On
Punch Cycle Trigger Off

1	2	3	4
X Gate	X Gate	Y Gate	B Reg Ch Pl XU 11
Pch Check Decode	Punch	RD 2 Signal	Tran Scan On
Row bit X	Row bit Y	Row bit X	PFR Ch Pl Reg
Regen XU	Regen YU	Rd Scan Gt	Upper Tran Reg
Regen XL	Regen YL	Regen XU	Pch Cy Trig Off
			Addr Pch Area
		Regen XL	Inhibit YU 12

Figure 98. Example of PFR Plane Selection

read trigger remains in the position in which it was last used and is not changed until the next read operation.

The plane over the punch storage to which data is transferred is always the plane which was last used by the punch check brushes. Recall that checking is completed for the previous card before the transfer scan takes place. The condition of the punch cycle trigger, ON or OFF, causes the selection of the plane to be used by the punch check brushes. On the same card feed cycle, the punch cycle trigger also selects the plane to which data is transferred. The punch cycle trigger is flipped to its opposite condition at the beginning of each punch cycle. Figure 97 shows the planes selected for entry from each set of brushes in the 1402 for various positions of the read and punch cycle. For example, if the read cycle trigger is ON from the last read cycle, the RD-2 plane will be set from the PFR read brushes; the Y gate will be up and the signals coming from the second row bit plane will be gated to the XU and XL check planes over the read area, as indicated by Figure 97. If the punch cycle trigger is also ON, the punch check decode output is entered in the YU and YL planes, and the punch check brush row bit output is entered in the XU and XL planes. Recall that the XU and XL check planes over the read area are transferred to the corresponding planes over the punch area after the check is completed.

A specific example of the selection of planes, using two 1402 machine cycles, is shown in Figure 98. The data flow to cause the inhibiting of the cores in the XU, XL or YU, YL is also shown.

Error Operation

If the I/O check stop switch is ON, any error sensed during after-9-punch scan time will cause the TRANSFER SCAN latch to be held OFF, the PFR Op trigger will be turned OFF, and the machine will come to a stop. Circuitry to accomplish this is shown on ILD 64. The operator must rerun the card because the sense error detection is too late to prevent processing. Also, the original data is not available, and the next read card hole count cannot be transferred.

If the I/O check stop switch is OFF the machine continues to run, and normal error interrogation and stacker select procedure may be followed.

A validity error stops the machine in the same manner except that the validity condition is checked simultaneously with reading. Therefore, the invalid bits will still be present in the read area of storage when the machine stops. The operator may either correct the error with the bit switches or rerun the card.

End-of-File Operation

There are no provisions for end-of-file run-out on the punch feed when the customer has the punch feed read option. If an end-of-file procedure is required, the customer may program a branch instruction on an X-punch in a digit-only column. The card containing the last read information must be followed by at least three trailer cards to complete the checking of the data.

Form 225-6541-0

IBM 1401
Printed in U.S.A.
Form 225-6541-0

IBM
International Business Machines Corporation
Data Processing Division
112 East Post Road, White Plains, New York