IBM Field Engineering

Instruction-Maintenance

IBM 7040 to 7090/7094; 7044 to 7094/7094 II

Direct Coupled Multiprocessing

IBM®

Field Engineering

Instruction-Maintenance

IBM 7040 to 7090/7094; 7044 to 7094/7094 II

Direct Coupled Multiprocessing

Preface

This manual describes the IBM 7040/7044–7090/7094/7094 II Direct Coupled Multiprocessing Systems. The material is written assuming previous knowledge of both the 704X and 709X systems.

In the majority of cases the flow charts, timing charts. and text of this manual refer to a 7040-7094 multiprocessing operation.

The 7090 operations are, for the most part, identical to 7094 operations except for cycle timings (2.18 microseconds for the 7090 vs. 2.00 microseconds for the 7094) and no instruction overlap considerations. The 7044 operations are identical to 7040 operations except for cycle timings (2.5 microseconds for the 7040 vs 2.0 microseconds for the 7044) and no alpha or beta time restrictions connected with the 7107 core storage.

The last section of this manual (7094 II Direct Coupled Operations) discusses the differences from 7094 I operations. Therefore, 7094 I operations should be thoroughly understood before studying the 7094 II section.

Condensed logic diagrams used in this manual are as close to actual systems as possible. Most of these diagrams have been converted to positive logic by eliminating any references to + or − levels. In maintaining this positive logic, in-phase outputs are used to indicate an active (conditions met) state from the condensed logic block. Out-of-phase outputs are also used in some cases to simplify the diagrams by eliminating the cluttering effect of convert and invert blocks.

This manual has been written at engineering change level 407475 for the 7040/7044; at level 254039 for the 7094; and at level 254442 for the 7094 II. However, future engineering changes may change the logic and machine operations from the presentation in this manual.

A list of available manuals for reference use is given in Appendix C.

This manual obsoletes the following earlier publications:
IBM Field Engineering Instruction-Maintenance Manual Z22-2806-0;
IBM Field Engineering Manuals Supplement S23-4022-0;
IBM Field Engineering Manuals Supplement S23-4023-0

Copies of this and other IBM publications can be obtained through IBM Branch Offices.
Address comments concerning the contents of this publication to:
IBM Corporation, FE Manuals, Department B96, PO Box 390, Poughkeepsie, N.Y. 12602

Contents

Illustrations

The following safety practices should be observed:

1. At least two men should be within sight of each other when working on the machine with power on.

2. Safety glasses must be worn when soldering or performing other operations which may endanger the eyes.

3. Remove metal jewelry before servicing the computer.

4. Use caution when lowering a tailgate on the 709X systems. Keep fingers clear of gate slides when sliding a gate into a module. Avoid hitting laminar bus connections.

5. Discharge capacitors before working on DC supplies.

6. Always turn power off before replacing a fuse.

7. Replace safety covers after each operation, and before proceeding to another operation.

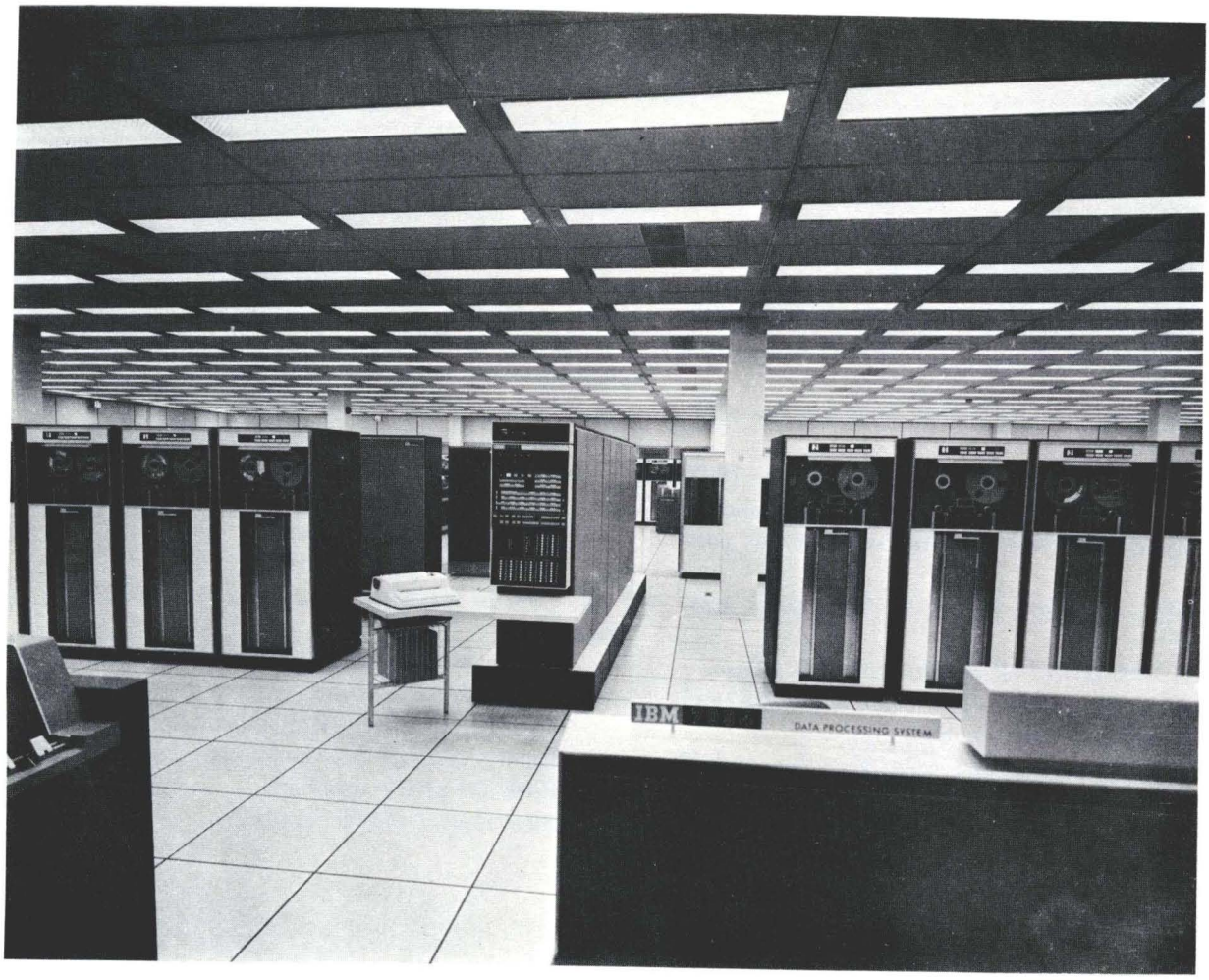
8. On the 709X systems, 120 volts, 60 cycles, and 48 VDC are still present inside SMS frames with frame power off and 7618 power on. If it is necessary to work near live power connections, convenience outlets, or inside the MG unit or core storage control, disconnect power cables, or turn power off at the wall circuit breakers.

9. On the 704X systems, line voltages of 208 VAC are always present at the following points:

Main power (K21) contacts.
F1 and F2 fuses for transformer T1.
R71, master power connect.
Master power connect switch.
Transformer T1 primary.
Console terminal board 3.

10. Prior to servicing, note and check the following items:

Master power switch locations_____
Air conditioning switch location_____
Fire extinguishers (CO_2 type)_____
Emergency exit door locations _____
First aid phone number _____
Fire control phone number_____



Direct Coupled Multiprocessing System

One of the main concerns with every computing organization is the job turn-around time: the time required for a computing center to perform a particular job and return the output results to the originator.

Each job submitted requires a certain amount of work such as peripheral card to tape, computer scheduling, actual running of the job on the computer, peripheral tape to printer and/or tape to card, bookkeeping, etc.

The multiprocessing approach to a better operating system includes:

Elimination of unnecessary work.
Performance of each step as fast as possible.
Performance of as many steps simultaneously as possible.

For our purposes, "multiprocessing" exists only when two or more processing units, each capable of interpreting and executing its own stored program, operate simultaneously on the same problem. During this processing there is a transfer of information between the processing units.

The multiprocessing concept has been performed for some time by using the 7090/7094 computer and associated IBM 7909 Data Channels. In the 7909, a stored program of instructions is actually interpreted and executed in parallel with interpretation and execution of the main program by the 7090/7094.

The multiprocessing concept has now been extended to include the 7040-7090 series of computers in the following configurations:

7040-7090
7040-7094
7040-7094 II
7044-7094
7044-7094 II

The systems of this directly coupled, multiprocessing complex (Figure 1) are coupled (processing unit to processing unit) by cables incorporating data lines, address lines, control lines, and timing lines. Direct coupling of this kind, when used as a "pure" multiprocessing configuration, allows the 7094 to be operated without data channels and input-output units. Input jobs are supplied by existing 7040 equipment such as card readers, magnetic tape units, disk or drum files, terminals, etc.

For an effective multiprocessing operation, each computer must have the ability to trap the other and cause it to start executing a predetermined program at a specific location. In this case, real-time traps provide the means of communication between the two systems. The 7094 locations 00003 and 00004, and the 7040 locations 00034_8 and 00035_8, provide the trapping data and instruction references.

The two computer systems are connected so that the 7040 appears to the 7094 as a data channel, and the 7094 storage appears as an extended storage facility to the 7040.

A modified 7040 transmit (TMT) instruction allows movement of blocks of data between the two core storages. The 7040 loads, starts, and monitors the processing function of the 7094. This increased ability of the 7040 likewise makes possible the performance of all input-output operations required by the 7094.

The 7040, by means of its own particular control program, can perform any number of functions such as:

Automatic input job stacking on the IBM 1301 Disk Storage Unit.
Priority sequencing of jobs.
Buffer administration for all I/O activities.
System program loading into the 7090/7094/7094 II.
Utility functions, including printing and punching.

While the 7040 is performing these functions, the 7094 system simultaneously performs all of the functions necessary to satisfy the 7040 input job requirements such as compiling, assembling, and executing.

The multiprocessing complex might be likened to a multiball pinball game. Many balls are in motion at the same time preprocessing an input, executing a main program, and postprocessing output printers and a punch. As soon as one ball (phase of a job) completes its operations, another shot (job phase) is initiated. In this manner a maximum number of jobs are always in operation.

Halt on I/O Operations (HIP) Mode

- **Allows existing programs to be run with little or no modification.**
- **7094 I/O instructions trap the 7040 rather than execute normally.**
- **HIP mode is under 7040 program control.**

In the pure multiprocessing system the 7094 informs the 7040 of all input-output requirements rather than address the I/O units directly. There are, however, many efficient 7094 stand-alone programs already written which require data channels. HIP mode allows these stand-alone programs to be executed without attached data channels.

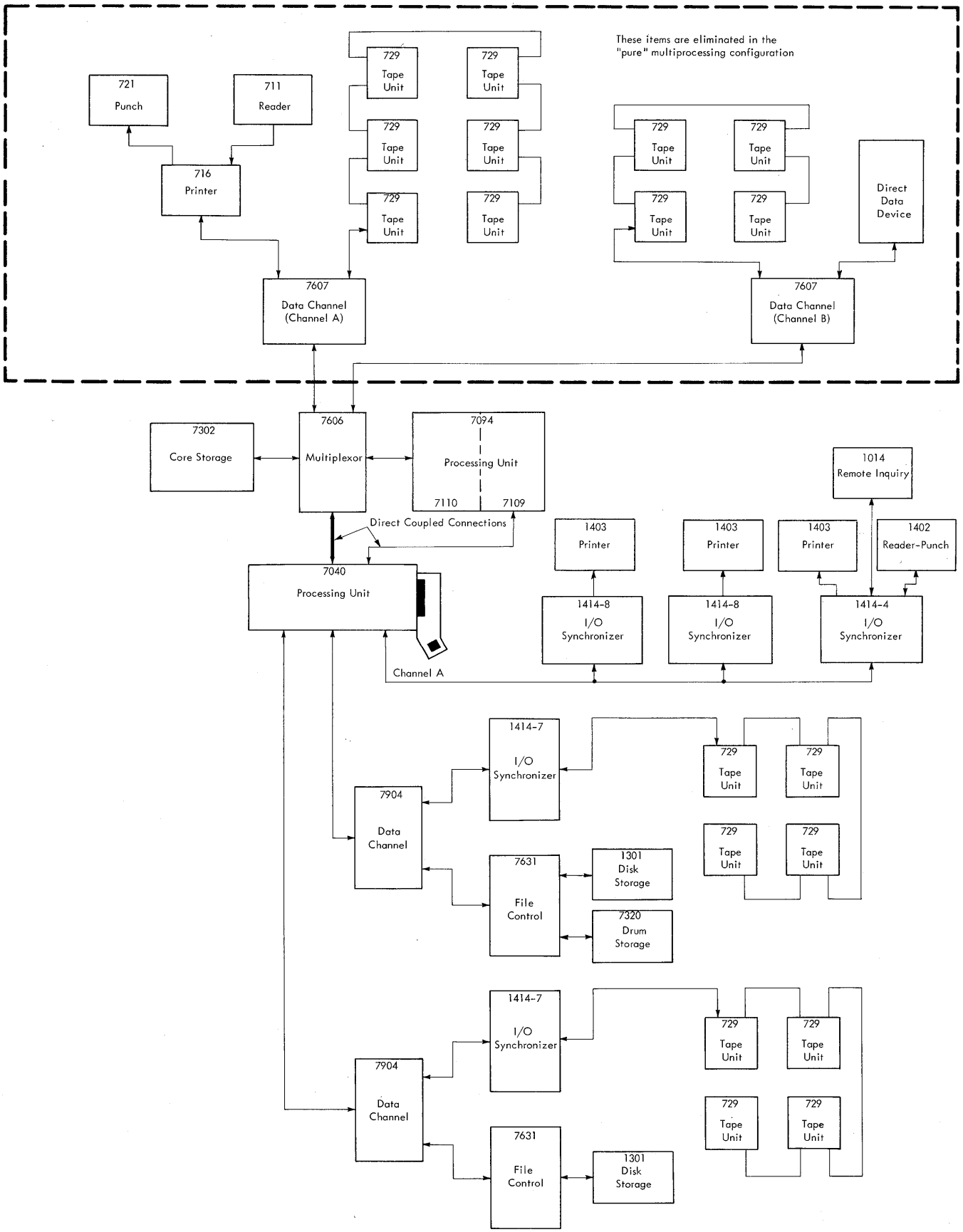


Figure 1. Typical Multiprocess System

When the system is in HIP mode, any I/O instructions except BTT(M), where M is the multiprocess channel, and IOT cause the 7094 to halt. The halt, in turn, causes a trap request to the 7040 which can then simulate the response that would have been received from the required channel.

With the aid of additional 7040 instructions and trap condition bits stored in location 00034_8 of the 7040, a system program can be used to simulate the 7094 I/O operations with little or no modifications.

Details of these new instructions and trap condition bits are covered in later sections of this manual.

Channel Exempt Mode

- **Allows specified channels to be maintained for 7094 use in HIP mode.**
- **Exempt mode is under 7040 program control.**

Cases exist where it is either desirable or necessary to physically attach I/O channels and devices to the 7094 (Figure 1).

Exempt mode with the new associated 7040 instructions allows the 7094 to operate selected channels without halting or causing the 7040 to trap while in HIP mode.

When in both HIP and exempt modes, the 7094 can be executing instructions and performing functions on the physically attached channels while, at the same time, the 7040 is simulating I/O operations associated with channels not physically attached to the 7094 system.

The selection of exempt channels is determined by the particular customer installation and is accomplished by physical wiring in the circuit logic. All of the 7094 exempt channels enter and leave exempt mode simultaneously under 7040 program control.

For example assume that a 7094 program is being executed which uses channels B and C (Figure 1). Channel B is an exempt channel physically attached for use of the direct data device. When in both exempt and HIP mode, instructions directed at Channel B are executed normally. Any instructions directed at Channel C, however, cause the 7094 to halt and the 7040 to trap.

Physical Connections and Operating Conditions

- **The 7040 is treated as a data channel.**
- **Both the 7040 and 7094 can have stand-alone capabilities.**
- **The 7094 card reader and printers must be retained for diagnostic maintenance requirements on a stand-alone system.**

The 7040 system is treated as a data channel by the 7094. Because of this, 20-position I/O input biscuit connectors have been added below the logic panels in 01C of the 7040. Corresponding terminator-jumper-connectors have also been added at 01D. These connectors accommodate the eleven signal, priority, and assignment cables which are normally routed from the 7606 multiplexor to the bank 1 and 2 data channels. In addition, a twelfth (multiprocess) cable is sent from CPU 2 of the 7094 to the 7040. This multiprocess cable (as true also of the assignment cable) comes directly from the 7094 and does not require rerouting to other channels on the bank.

The 7040 can be operated as a stand-alone system, when power is off on the 7094. This could occur, for example, when the main processing has been completed by the 7094 and the 7040 is still needed for postprocessing and peripheral functions. The 7040 program should be stopped, however, while turning power on or off the 7094 to eliminate the possibility of 7040 errors due to electrical transients.

If the 7040 is the only connection on a channel bank of the 7094, power can be turned off of the 7040 without imparing stand-alone capabilities of the 7094.

If the 7040 is the terminating unit on a bank with other 7607 or 7909 Data Channels, power cannot be removed from the 7040 without affecting the stand-alone capabilities of the 7094. If the 7040 is not the terminating unit, however, power can be removed. Removing power from the 7040 system de-energizes a reed relay at card location 01C3A01 which allows the "remote required" signal from higher priority channels to pass through the 7040 and on to lower priority channels.

After installation of the direct couple feature, the 7094 channels may be removed. If the 7094 channels are removed to the extent that the IBM 711 Card Reader and IBM 716 Printer are removed, the 7040 must be available for required maintenance performed on the 7094.

The 7040 is the controlling computer in the multiprocessing complex. Therefore, it should not seem unusual to find the new instructions and operations associated with the 7040.

For ease in describing these new instructions, they have been grouped as follows:

7040 mode instructions
7040 start instructions
7040 functional instructions
7040 transmit instruction
7904 scatter read

7040 Mode Instructions

Enter Multiprocess Mode EMM −1774...40 (I)

- **Execution activates the multiprocess connection between the two systems.**
- **The EMM acts as a NOP if already in multiprocess mode.**
- **Any address other than 40 may cause the EMM instruction to operate differently.**
- **Any 7040 manual reset causes the system to leave the multiprocess mode.**

The purpose of the EMM instruction is to activate the multiprocess connection between the two systems.

During the 7040 I cycle of the enter multiprocess mode instruction, the storage bus is gated into the storage register and program register at I3 time as a normal I time function (Figures 2 and 3).

At I5(D1), the shift counter is set from AD(28-35). This setting occurs as a normal POD 7X function but performs no logic in the overall operation concerning any of the −1774...nn instructions.

Because the EMM instruction is also decoded in some respects as an AXT instruction (SOD 14), the address register is blocked from being set from the adders. This blocking also performs no actual logic with the EMM or any of the other −1774...nn instructions. Any accidental tagging associated with any of these mode instructions will not affect the specified index register.

End-op for the EMM instruction or any of the other −1774...nn instructions occurs at the end of I time through existing SOD 14(AXT) instruction circuitry.

After decoding 774 (Figures 2 and 3), the storage register is interrogated for a bit in SR(30) and, if a one, the "allow mulp mode" trigger is turned on. Outputs from the mulp mode trigger act to condition 7040 circuitry so that the multiprocess function can be carried out.

Note that there is no mulp mode signal gating in the 7094 to indicate a multiprocess condition. While in multiprocess mode only, the 7094 can function as either a stand-alone computer (with or without data channels) or as a part of the multiprocess system.

An indicator is located on the 7040 back panel at 01D3B18-9 to show the present status of the mulp mode trigger.

Enter HIP Mode EHM −1774...01 (I)

- **Execution places the system in HIP mode. HIP mode performs no logic in the 7040 system.**
- **Any address other than 01 may cause the EHM instruction to operate differently.**
- **The EHM instruction acts as a NOP if the system is not in multiprocess mode.**
- **The EHM instruction is not affected by the status of channel exempt mode.**
- **Any 7040 manual reset causes the system to leave HIP mode.**

The purpose of HIP mode is to allow existing 7094 channel programs and subroutines to be executed on a 7094 system which is devoid of any or all data channels.

While in HIP (compatability) mode, 7094 channel-type instructions are prevented from execution. Instead, the 7040 is forced into a trap condition to allow 7040 program simulation of the 7094 channel instruction.

The EHM instruction (Figures 2 and 3) performs all of the initial I time functions as explained previously for the EMM instruction.

After decoding 774, the storage register is interrogated for a bit in SR(35) and, if a one, the HIP mode trigger is turned on. The output of this trigger, HIP mode, is sent directly to the 7094 where circuitry is conditioned for detecting HIP halt and trap conditions.

Note that HIP mode performs no conditioning or logical functions in the 7040. An indicator is located on the 7040 back panel at 01D3B17-9 to show the present status of the trigger.

Enter Exempt Mode EEM —1774...10 (I)

- **Execution places the 7094 in exempt mode. Exempt mode performs no logic in the 7040 system.**
- **Any address other than 10 may cause the EEM instruction to operate differently.**
- **The EEM instruction acts as a NOP if the system is not in multiprocess mode.**
- **The EEM instruction is not affected by the status of HIP mode.**
- **Any 7040 manual reset causes the systems to leave exempt mode.**

Exempt mode allows certain selected data channels on the 7094 to be exempt from HIP traps. In this manner existing channels can be maintained and used on the 7094 system.

While in exempt mode, any channel instructions directed at an exempt channel cause the channel to operate normally as in a stand-alone situation. When in HIP motde, any channel-type instruction directed at a non-exempt channel causes a HIP halt and a 7040 trap.

The EEM instruction (Figures 2 and 3) performs all of the initial I time functions as explained previously for the EMM instruction.

After decoding 774, the storage register is interrogated for a bit in SR(32) and, if a one, the exempt mode trigger is turned on. The output of this trigger, exempt mode, is sent directly to the 7094 where circuitry is conditioned for detecting a channel instruction to an exempt channel.

Note that exempt mode performs no conditioning or logical functions in the 7040. An indicator is located on the 7040 back panel at 01D3B16-9 to show the present status of the trigger.

Leave Multiprocess Mode LMM —1774...04 (I)

- **Execution deactivates the multiprocess connection between the two systems.**
- **Leaving multiprocess mode also causes the system to leave HIP and exempt modes.**
- **Any 7040 manual reset causes the system to leave multiprocess, HIP, and exempt modes of operation.**

The purpose of the LMM instruction is to deactivate the multiprocess connection between the two systems. Each system then has stand-alone capabilities only.

The LMM instruction (Figures 2 and 3) performs all of the initial I time functions as explained previously for the EMM instruction.

After decoding 774, the storage register is interrogated for a bit in SR(33) and, if a one, the allow mulp mode trigger is turned off. Reset of this trigger deconditions all circuits in the 7040 concerned with multiprocess operations.

The off output condition of the mulp mode trigger acts as a continuous reset to both the HIP and exempt mode triggers.

Leave HIP Mode LHM —1774...02 (I)

- **Execution removes the system from HIP mode of operation.**
- **Any address other than 02 may cause the LHM instruction to operate differently.**
- **Execution of the LHM instruction has no effect on channel exempt mode status.**
- **Any 7040 manual reset also causes the system to leave HIP mode.**

Execution of the LHM instruction removes the 7094 from HIP mode operation and allows it to operate as either a pure multiprocess system or as a stand-alone computer.

The LHM instruction (Figures 2 and 3) performs all of the initial I time functions as explained previously for the EMM instruction.

After decoding 774, the storage register is interrogated for a bit in SR(34) and, if a one, the HIP mode trigger is turned off. Reset of this trigger deactivates the "HIP mode" signal to the 7094 and, therefore, prevents initiation of 7094 HIP traps due to channel-type instruction.

Removal of "HIP mode" gating in the 7094 also resets the block traps trigger so that subsequent channel traps can be honored. This block traps trigger is turned on at the time of any HIP trap to the 7040.

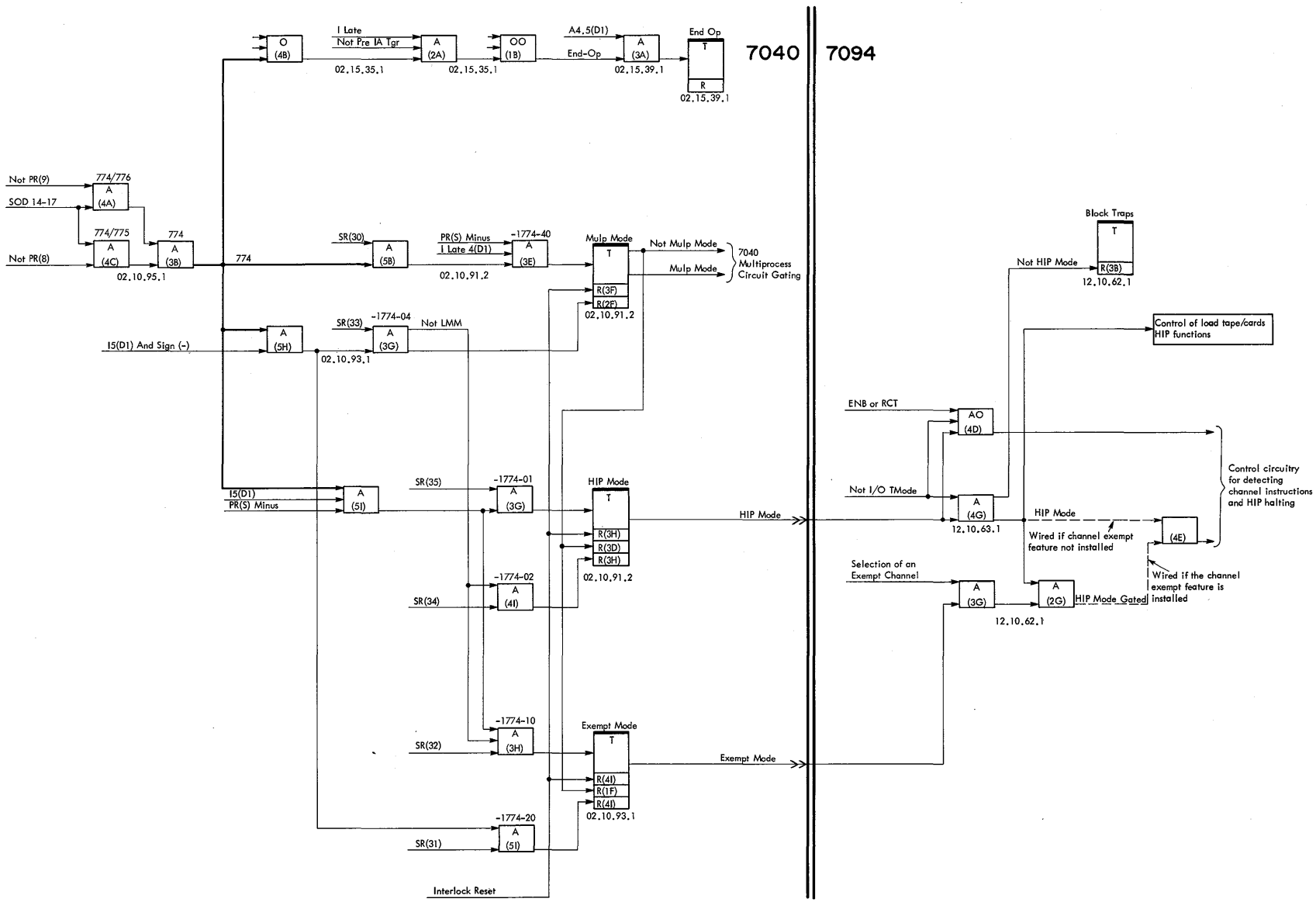


Figure 2. Enter and Leave Mode Condensed Logic

Valid -1774...nn Instruction Summary

Operation Code	SR (bit) and Operation						Final Mode Status		
	EMM (30)	LEM (31)	EEM (32)	LMM (33)	LHM (34)	EHM (35)	Mulp	HIP	Exempt
-1774...01*	0	0	0	0	0	1	Yes	Yes	--
-1774...02*	0	0	0	0	1	0	Yes	No	--
-1774...04	0	0	0	1	0	0	No	No	No
-1774...06	0	0	0	1	1	0	No	No	No
-1774...10*	0	0	1	0	0	0	Yes	--	Yes
-1774...11*	0	0	1	0	0	1	Yes	Yes	Yes
-1774...12*	0	0	1	0	1	0	Yes	No	Yes
-1774...20*	0	1	0	0	0	0	Yes	--	No
-1774...21*	0	1	0	0	0	1	Yes	Yes	No
-1774...22*	0	1	0	0	1	0	Yes	No	No
-1774...24	0	1	0	1	0	0	No	No	No
-1774...26	0	1	0	1	1	0	No	No	No
-1774...40	1	0	0	0	0	0	Yes	--	--
-1774...41	1	0	0	0	0	1	Yes	Yes	--
-1774...42	1	0	0	0	1	0	Yes	No	--
-1774...50	1	0	1	0	0	0	Yes	--	Yes
-1774...51	1	0	1	0	0	1	Yes	Yes	Yes
-1774...52	1	0	1	0	1	0	Yes	No	Yes
-1774...60	1	1	0	0	0	0	Yes	--	No
-1774...61	1	1	0	0	0	1	Yes	Yes	No
-1774...62	1	1	0	0	1	0	Yes	No	No

* This instruction is valid if the system is already in multiprocess mode
-- Final mode status is the same as the status prior to execution of the instruction provided the system is already in multiprocess mode

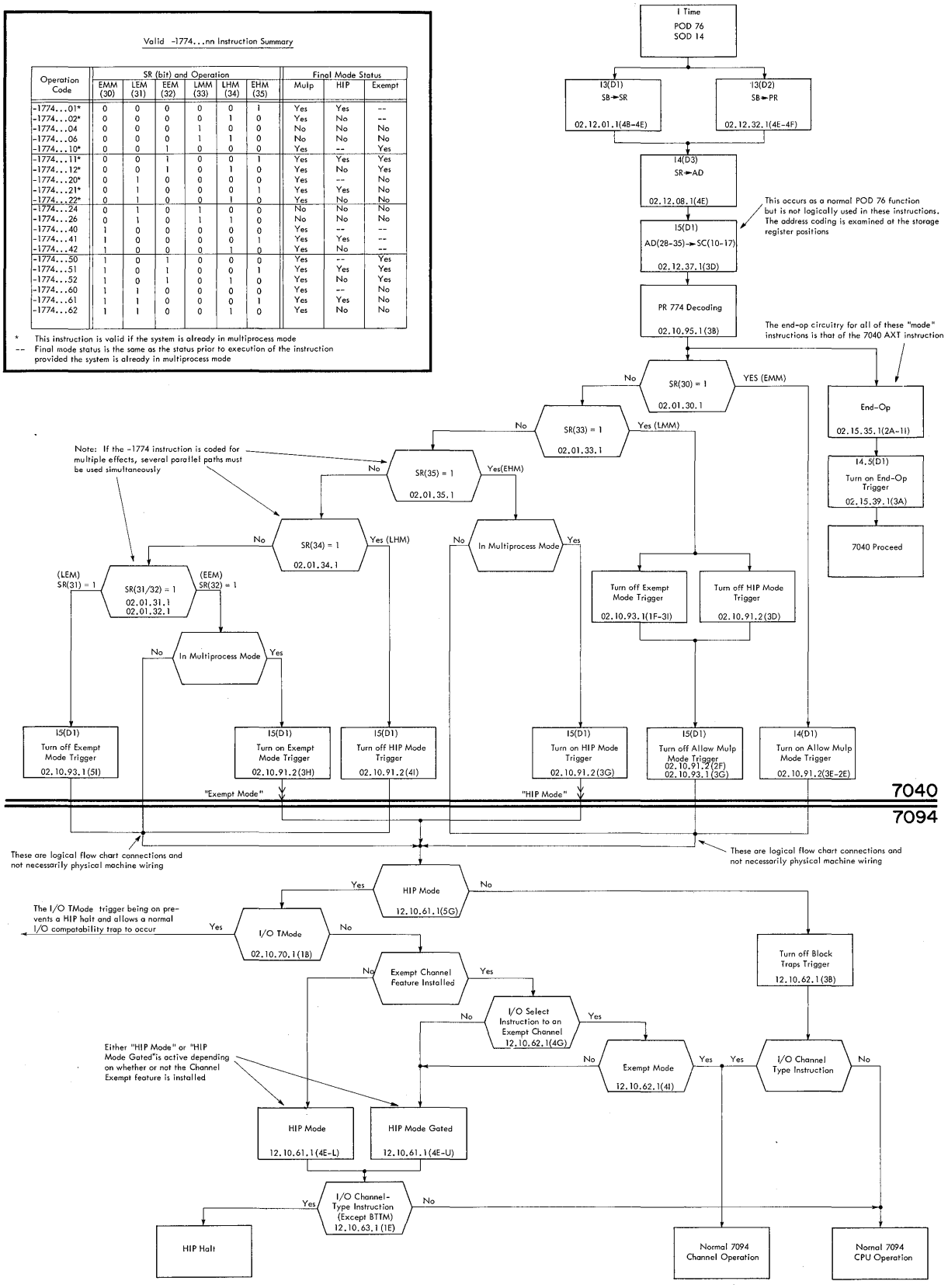


Figure 3. Six Mode Instructions:
Enter Multiprocess Mode (EMM)
Enter HIP Mode (EHM)
Enter Exempt Mode (EEM)
Leave Multiprocess Mode (LMM)
Leave HIP Mode (LHM)
Leave Exempt Mode (LEM)

Operation Code	SR (bit) and Operation						Final Mode Status		
	EMM (30)	LEM (31)	EEM (32)	LMM (33)	LHM (34)	EHM (35)	Mulp	HIP	Exempt
-1774...01 *	0	0	0	0	0	1	Yes	Yes	—
-1774...02 *	0	0	0	0	1	0	Yes	No	—
-1774...04	0	0	0	1	0	0	No	No	No
-1774...06	0	0	0	1	1	0	No	No	No
-1774...10 *	0	0	1	0	0	0	Yes	—	Yes
-1774...11 *	0	0	1	0	0	1	Yes	Yes	Yes
-1774...12 *	0	0	1	0	1	0	Yes	No	Yes
-1774...20 *	0	1	0	0	0	0	Yes	—	No
-1774...21 *	0	1	0	0	0	1	Yes	Yes	No
-1774...22 *	0	1	0	0	1	0	Yes	No	No
-1774...24	0	1	0	1	0	0	No	No	No
-1774...26	0	1	0	1	1	0	No	No	No
-1774...40	1	0	0	0	0	0	Yes	—	—
-1774...41	1	0	0	0	0	1	Yes	Yes	—
-1774...42	1	0	0	0	1	0	Yes	No	—
-1774...50	1	0	1	0	0	0	Yes	—	Yes
-1774...51	1	0	1	0	0	1	Yes	Yes	Yes
-1774...52	1	0	1	0	1	0	Yes	No	Yes
-1774...60	1	1	0	0	0	0	Yes	—	No
-1774...61	1	1	0	0	0	1	Yes	Yes	No
-1774...62	1	1	0	0	1	0	Yes	No	No

* This instruction is valid if the systems is already in multiprocess mode

\- Final mode status is the same as the status prior to execution of the instruction provided the system is already in multiprocess mode

Figure 4. −1774 . . . nn Instruction Summary

Leave Exempt Mode LEM −1774...20 (I)

- **Execution removes the system from exempt mode of operation.**
- **Any address other than 20 may cause the LEM instruction to operate differently.**
- **Execution of the LEM instruction has no effect on HIP mode.**
- **Being in exempt mode while not in HIP mode has no logical significance (same effect as pure multiprocess or stand-alone status).**
- **Any 7040 manual reset also causes the system to leave exempt mode.**

Execution of the LEM instruction removes the 7094 from exempt mode and when also in HIP mode prevents normal on-line operation of data channels.

The LEM instruction (Figures 2 and 3) performs all of the initial I time functions as explained previously for the EMM instruction.

After decoding 774, the storage register is interrogated for a bit in SR(31) and, if a one, the exempt mode trigger is turned off. Reset of this trigger deactivates the exempt mode signal to the 7094 and, therefore, prevents operating 7094 attached channels while operating in HIP mode.

−1774...nn Instruction Summary

Figure 4 is a list of valid −1774...nn instructions which can be executed for single or multiple effects.

7040 Start Instructions

Start Remote Computer
SRC (SRC) — 1777...00 (I, L)

- **Execution is only effective if in multiprocess mode. If not in multiprocess mode, the start remote computer instruction is effectively NOP'ed.**
- **Any address other than 00 may cause the start remote computer instruction to operate differently.**
- **Execution of the SRC instruction has no effect on a 7094 already in operation.**
- **If the 7094 had stopped because of an HTR(Y) instruction, the start remote computer instruction starts the 7094 and causes the next instruction to be executed from location Y.**
- **If the 7094 had stopped because of an HPR (or HIP halt), DVH, or DFDH instruction, the start remote computer instruction starts the 7094 and causes the next sequential instruction to be executed.**

The SRC instruction is designed to simulate depression of the start key on the 7094 operator's console.

The 7094 could have halted because of:

1. A normal HPR or HTR instruction.
2. A DVH or DFDH instruction.
3. An HPR resulting from a HIP operation.

During the 7040 I cycle of the start remote computer instruction, the storage bus is gated into the storage register and program register at I3 time as a normal I time function (Figures 5 and 6).

At I5(D1), the shift counter is set from AD(28-35), and the address register is set from AD(21-35). These settings occur as a normal POD 7X function but perform no logic in the overall operation.

Note that only the −1777 decoding is needed to turn on the start trigger in the 7040. The 00 address portion of the SRC instruction is not interrogated in setting the start trigger.

The output of this start trigger, "start 7090/94," is sent directly to the 7094 and initiates a start operation similar to that of a manual 7094 start operation. The first 7094 A0(D1) pulse turns on the 7094 start trigger. The 7094 master stop trigger is turned off at A6 time (provided the 7094 is in automatic); the B cycle interrupt trigger is turned off at A11 time, and the computer proceeds into an L cycle. The L cycle trigger had been turned on at the time of the 7094 halt but its output was degated by the B cycle interrupt trigger.

The L cycle serves as a means of ending-op on the halt instruction and addressing memory for the next instruction. If the 7094 has stopped because of an HTR instruction, the conditions met trigger would be on and the address register is gated to MAR at L10 time (Figure 6). If the 7094 had stopped because of an HPR, DVH, or DFDH instruction, the conditions met trigger would be off and the program counter is gated to MAR for the next sequential instruction.

During the next 7094 I cycle, the program counter is updated from the address register at I3 time for transfer conditions. If the conditions met trigger is not on, the program counter is not updated but remains at its present value. A 7094 I time signal is sent to the 7040 and is used as a reset to the start trigger.

The 7040 start remote computer instruction ends-op at the end of the 7040 L cycle and is not dependent on whether the instruction was effectively executed or NOP'ed.

Note that all of the −1777 instructions also appear to the 7040 as shift instructions (POD 76 and PR 9 at Systems 02.12.38.1, 5A). Because of this, the shift counter will step to zero before the instruction completely ends operation at the following I3 time. Existing shift end-op circuitry (Systems 02.15.35.1, 5G) is also active if the SC=0 before L4(D1) time. The functions just mentioned provide no logic for the start computer instructions but occur because of existing 7040 circuitry.

Start and Skip SRC (SSC) — 1777...01 (I, L)

- **Execution is effective only if in multiprocess mode. If not in multiprocess mode, the SRC is effectively NOP'ed.**
- **Any address other than 01 may cause the start and skip instruction to operate differently.**
- **The start and skip instruction must not be executed while the 7094 is in operation; doing so may cause incorrect program execution.**

The purpose of the start and skip instruction is to start the 7094 after a HIP-type halt (HPR) associated with 7094 ETT, BTT, and SPT instructions. After a HIP halt resulting from one of these instructions, the 7040 program can cause the 7094 either to execute the next sequential instruction, or to skip one instruction by executing either an SRC. . . 00 or SRC. . . 01 instruction respectively.

If a start and skip instruction is issued to a 7094 stopped because of an HTR instruction, the transfer function overrides the skip request and the 7094 performs a transfer.

The start and skip instruction (Figures 5 and 6) performs all of the functions explained previously for the start remote computer instruction (−1777...00). The storage register is interrogated for a bit in SR(35) and, if a one, a start skip trigger is turned on in addition to the start trigger.

The output of the start skip trigger, "start and skip," enters the 7094 to produce "I/O sense skip gated," which causes the program counter to be incremented through the normal skip circuitry at 9 time of the L cycle following reset of the master stop trigger.

Note that a start and skip instruction issued to the 7094 while not stopped could cause unwanted program counter incrementing which, in turn, would cause improper program execution.

Start and Transfer SRC (TSC) −1777...02 (I, L)

- **Execution is effective only if in multiprocess mode. If not in multiprocess mode, the start and transfer is effectively NOP'ed.**
- **Any address other than 02 may cause the start and transfer instruction to operate differently.**
- **The start and transfer instruction must not be executed while the 7094 is in operation; doing so may cause incorrect program execution.**
- **A start and transfer instruction issued to a 7094 stopped because of a normal (not a HIP-type) HPR or a non-transfer type instruction causes the address register contents to be gated to MAR. A transfer, therefore, is effected to an address indicated by positions 21-35 (including address modification) of the instruction causing the halt.**

The purpose of the start and transfer instruction is to start the 7094 after a HIP-type halt (HPR) associated with 7094 TCO, TCN, TEF, and TRC instructions. After a HIP halt resulting from one of those instructions, the 7040 program can cause the 7094 either to execute the next sequential instruction or to transfer by executing either an SRC. . . 00 or SRC. . . 02 instruction, respectively.

The start and transfer instruction (Figures 5 and 6) performs all of the functions explained previously for the start remote computer instruction (−1777...00). In addition, the storage register is interrogated for a bit in SR(34) and, if a one, a start transfer trigger is turned on in addition to the start trigger.

The output of the start transfer trigger, "start 9X and Xfer" enters the 7094 to turn on the conditions met trigger at 10 time of the L cycle following reset of the master stop trigger.

Note that a start and transfer instruction issued to the 7094 while not stopped could cause an unwanted program transfer at the end of the current 7094 instruction being executed.

An SRC...03 (−1777...03) issued either on purpose or by mistake will attempt a transfer and skip. The transfer function, however, overrides the skip function.

Start and Enable SRC (ENS) −1777...04 (I, L)

- **Execution is effective only if in multiprocess mode. If not in multiprocess mode, the start and enable instruction is effectively NOP'ed.**
- **Any address other than 04 may cause the start and enable instruction to operate differently.**
- **The start and enable instruction will not affect 7094 operation if issued while the 7094 is in operation (other than enabling for MPT traps).**
- **Other than enabling for MPT, the start and enable instruction has all of the characteristics of a start instruction.**

The purpose of the start and enable instruction is to start the 7094 and, at the same time, enable the 7040 for MPT traps to the 7094. MPT enabling is also accomplished by a CW bit (position 31 for channel E) associated with the 7094 enable (ENB) instruction. This latter method, however, requires modification of existing 7094 programs and is therefore undesirable in some instances.

The start and enable instruction (Figures 5 and 6) performs all of the functions explained previously for the start remote computer instruction (−1777...00). In addition, the storage register is interrogated for a bit in SR(33) and, if a one, an enable 7090 trap trigger is turned on in addition to the start trigger.

The start and enable instruction may be combined with the start and skip, or start and transfer instructions as follows:

−1777 . . . 05 Start, enable and skip
−1777 . . . 00 Start, enable and transfer

When the combinations are used, the appropriate additional restrictions apply for the skip and transfer conditions.

A summary of the valid SRC instructions follows:

SRC (SRC)	−1777 . . . 00	Start remote computer
SRC (SSC)	−1777 . . . 01	Start and skip
SRC (TSC)	−1777 . . . 02	Start and transfer
SRC (ENS)	−1777 . . . 04	Start and enable
SRC (SES)	−1777 . . . 05	Start, enable, and skip
SRC (SET)	−1777 . . . 06	Start, enable, and transfer

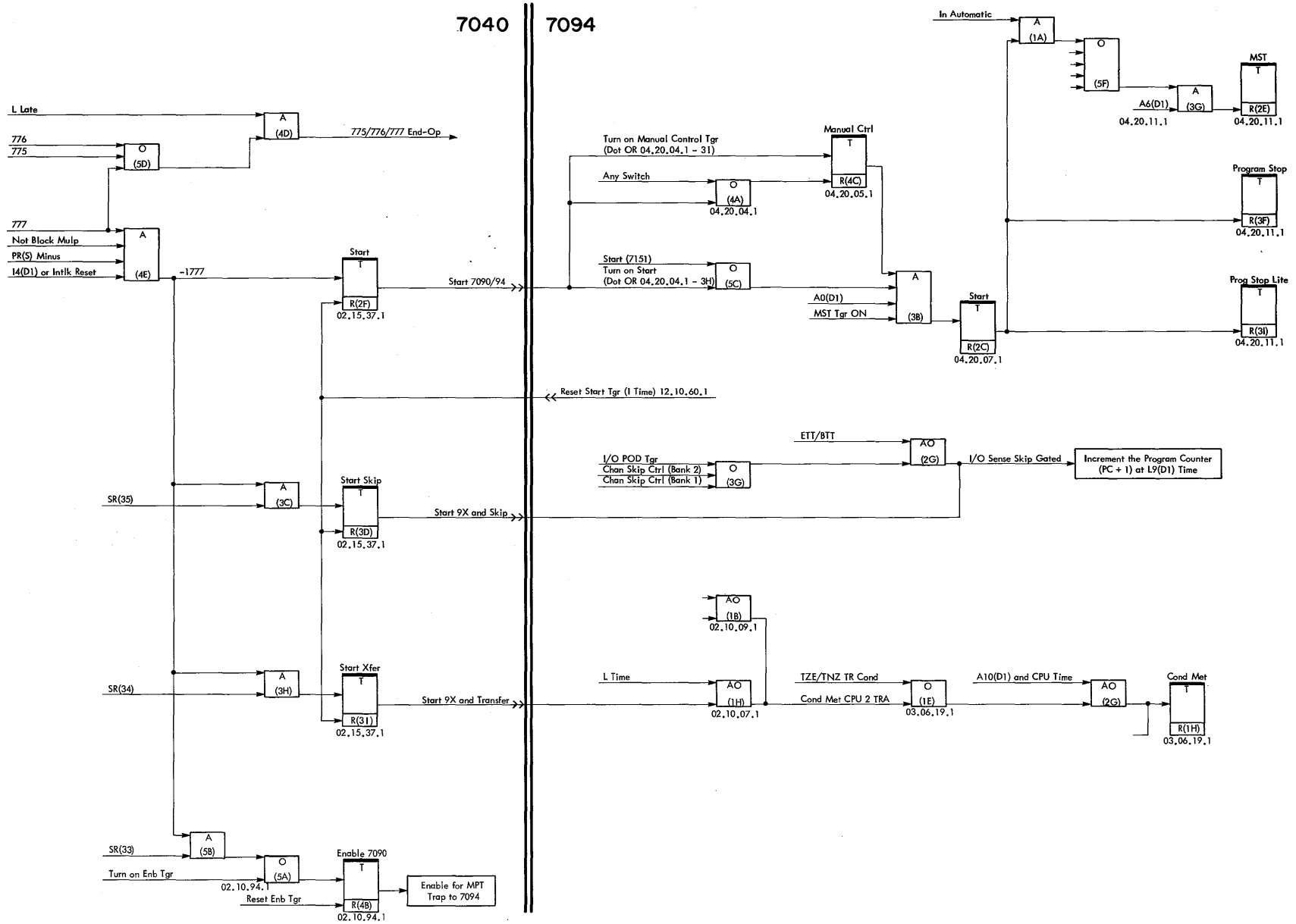


Figure 5. Start Computer Condensed Logic

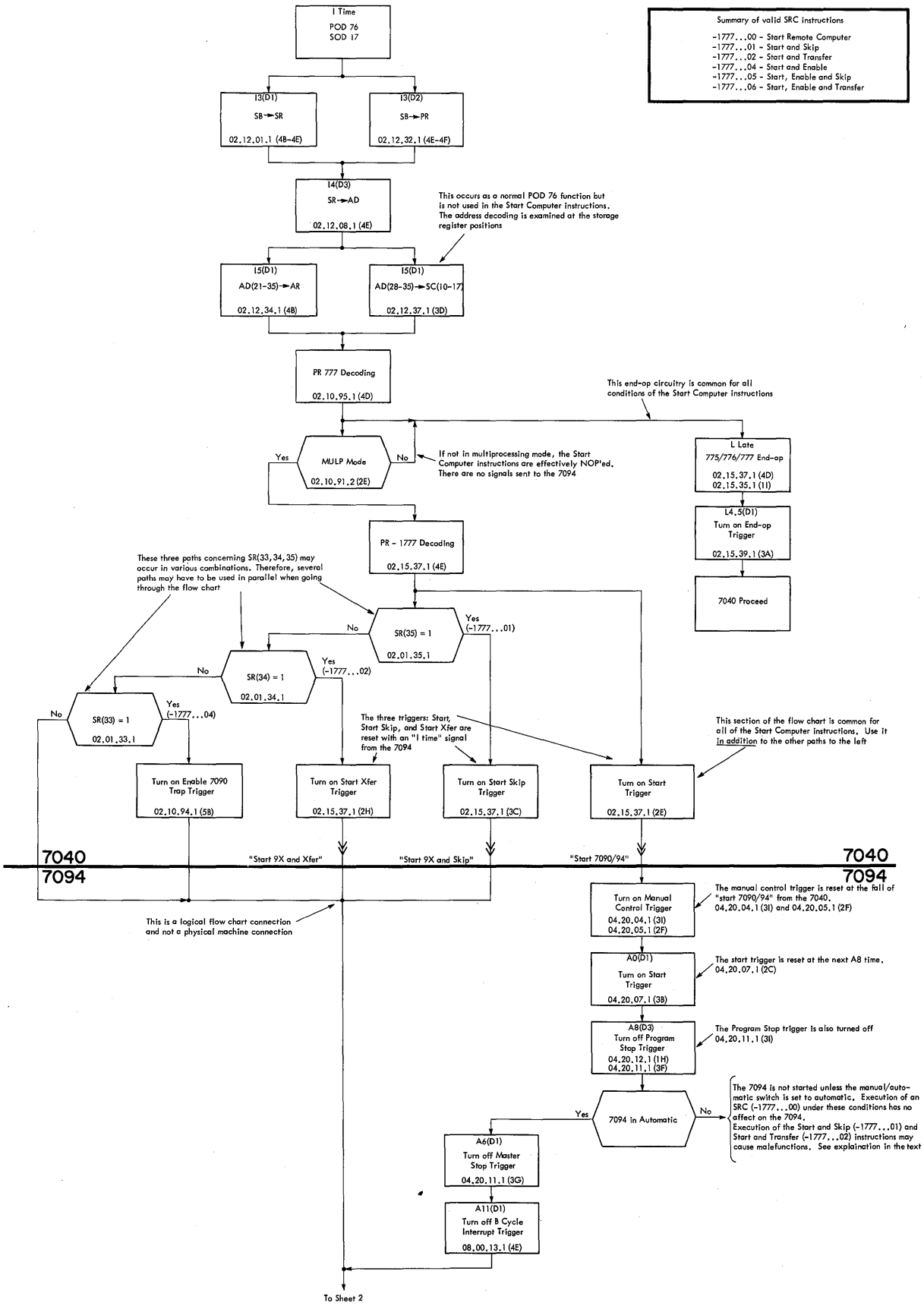


Figure 6. Four Start Instructions: (Sheet 1 of 2)
Start Remote Computer (SRC)
Start and Skip (SSC)
Start and Transfer (TSC)
Start and Enable (ENS)

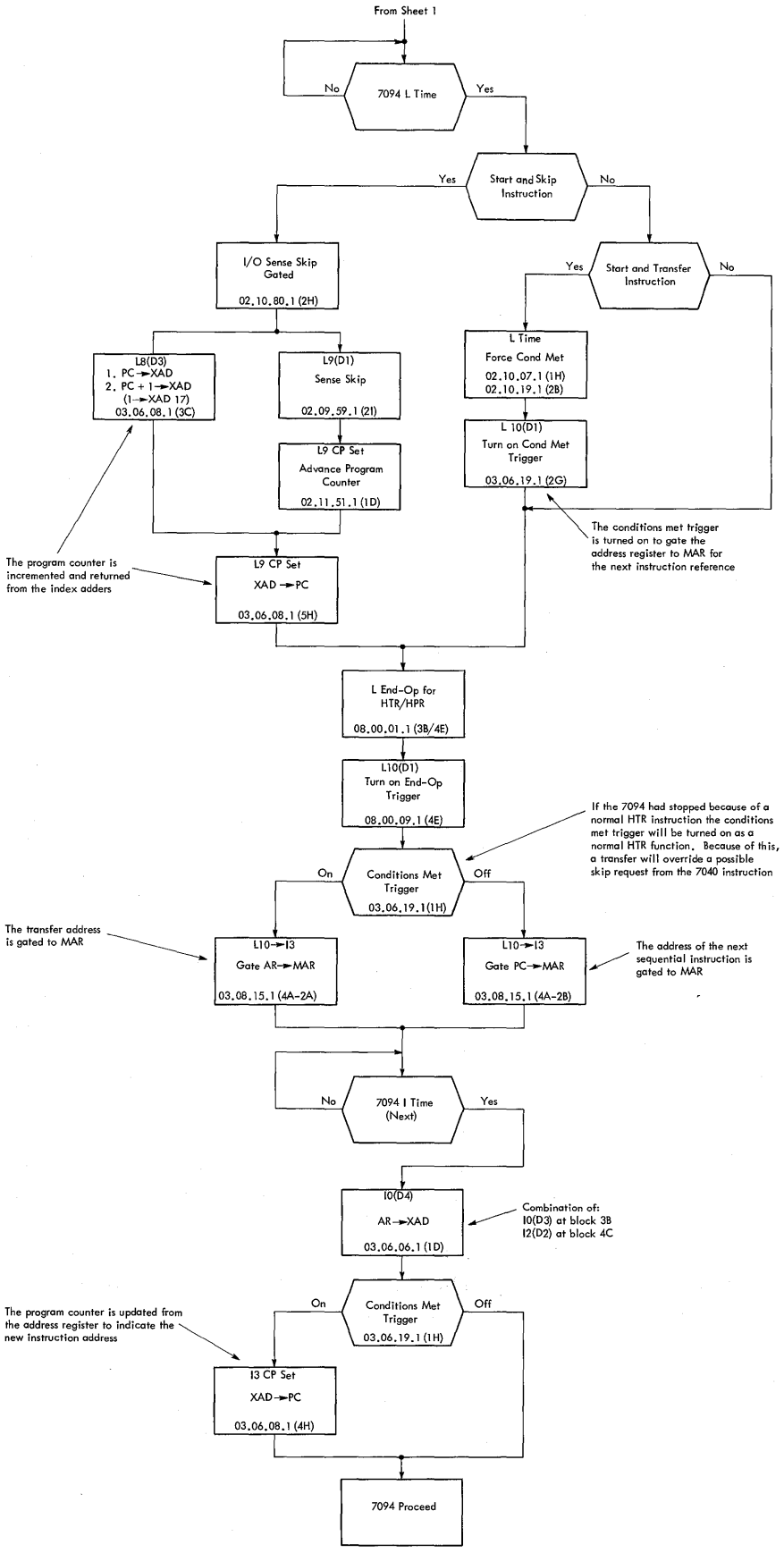


Figure 6. Four Start Instructions (Sheet of 2 of 2)

7040 Functional Instructions

Set I/O Check SIC +0775 (I, L)

- **Execution turns on the 7094 I/O check trigger only if the system is in multiprocess mode.**
- **The SIC instruction acts as a NOP if the system is not in multiprocess mode.**
- **The address portion of the SIC instruction has no effect on the overall operation.**

The SIC instruction allows the 7040 control program to turn on the I/O check trigger in the 7094. This setting duplicates the normal function of the 7607 data channels when detecting invalid instruction sequences or timing violations. The 7040 program, when simulating 7094 channel operations in HIP mode, checks for these invalid instruction sequences and if detected, turns on the I/O check trigger in the 7094.

During the 7040 I cycle of the SIC instruction, the storage bus is gated into the storage register and program register at I3 time as a normal I time function (Figures 7 and 8).

At I5(D1), the shift counter is set from AD(28-35), and the address register is set from AD(21-35). These settings occur as a normal POD 7X function but perform no logic in the overall operation.

There is no 7040 trigger associated with the SIC operation; only a conditioned I5(D1) pulse is sent to the 7094 as a set pulse to the I/O check trigger.

The L cycle of the SIC instruction performs no logic in the operation, and serves only as a means of ending-op (Figures 7 and 8).

Operation decoding (+0775) of the SIC instruction could also appear as a shift instruction (Systems 02.12.38.1, 5A). However "775" acts as a DOT-AND function to prevent stepping the shift counter and ending operation through other circuitry.

Reset Trap Inhibit RTI +0777 (I, L)

- **Execution of the RTI instruction resets the "block traps" trigger in the 7094.**
- **The RTI performs no logic (effectively NOP'ed) if the system is not in multiprocess and HIP modes.**
- **The address portion of the RTI instruction has no effect on the RTI function.**
- **The RTI instruction is not interlocked at the 7094 and can be effectively executed by the 7040 at any time. (only logically executed if in HIP mode with the channel exempt feature installed)**
- **The RTI also appears in some respects as a 7040 shift instruction causing the shift counter to step during L and I (next) times.**

All data channel traps to the 7094 are immediately suspended by a HIP halt. This suspension occurs only if the channel exempt feature is installed and is necessary to preserve the program counter value for 7040 use.

The RTI instruction allows the 7040 control program to recondition the 7094 for accepting data channel traps.

During the 7040 I cycle of the RIT instruction, the storage bus is gated into the storage register and program register at I3 time as a normal I time function (Figures 7 and 8).

At I5(D1) the shift counter is set from AD(28-35), and the address register is set from AD(21-35). These settings occur as a normal POD 7X function but perform no logic in the overall operation.

There is no 7040 trigger associated with the RTI instruction; only a conditioned I5(D1) pulse is sent to the 7094 as a reset pulse to the block traps trigger. The block traps trigger is held reset when not in HIP mode (Figure 7); therefore, the RTI instruction performs no active logic when the system is not already in HIP mode of operation.

There is no multiprocess mode interlocking needed with the RTI circuitry. Execution while not in multiprocess mode performs no logic and, therefore, has no effect on either system.

Note that the +0777 (RTI) instruction also appears to the 7040 as a shift instruction (POD 76 and PR 9 at Systems 02.12.38.1, 5A). Because of this, end-op circuitry is also active (Systems 02.15.35.1, 5G) at L4(D1) time. These functions perform no logic for the RTI instruction but occur because of existing 7040 circuitry.

The L cycle of the RTI instruction performs no logic in the operation, and serves only as a means of ending-op (Figures 7 and 8).

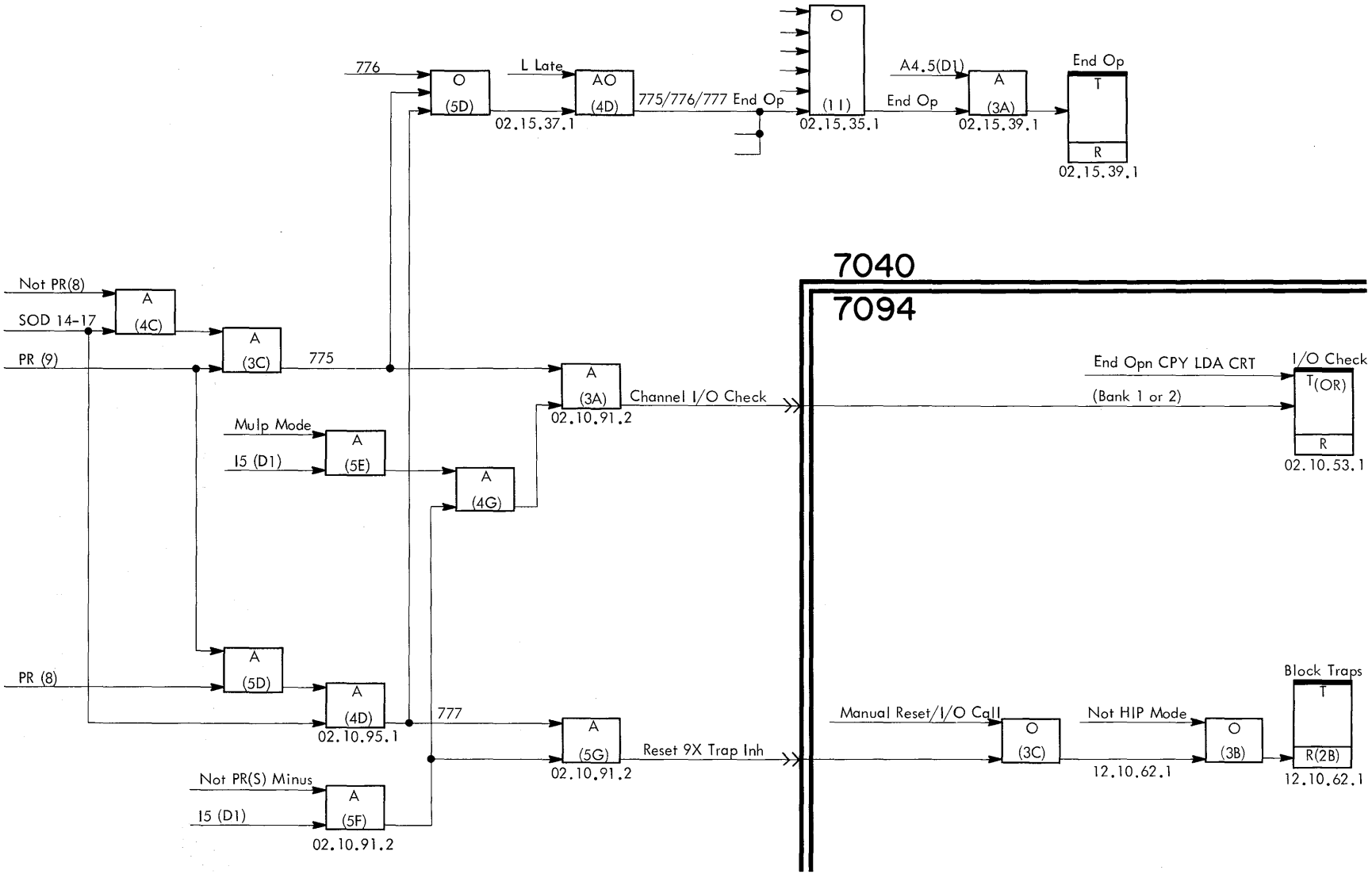


Figure 7. SIC and RTI Condensed Logic

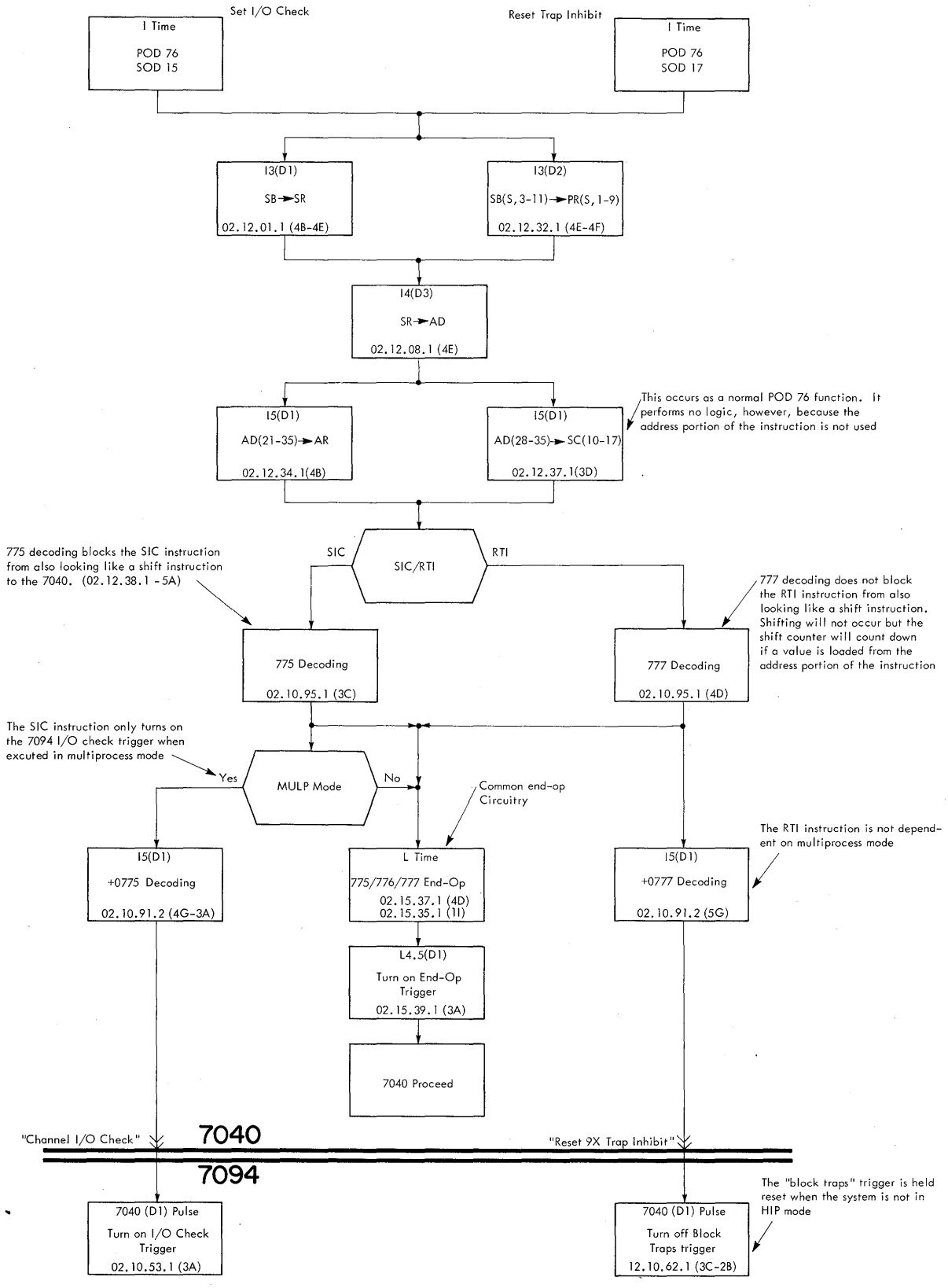


Figure 8. Flow Chart: Set I/O Check (SIC); Reset Trap Inhibit (RTI)

Set Column Binary
SCB −1440/−1441/−1442 (I)

- **Execution of the SCB instruction is *not* dependent on multiprocess, HIP, or channel exempt modes of operation.**
- **A separate SCB instruction is required for each row binary card read.**
- **A row binary card must not be the first card in the deck when initially readying the card reader.**
- **The SCB instruction must be executed not later than 7 milliseconds following the RCH instruction to the card reader.**
- **Only a 1402 card reader on the 1414-4 I/O synchronizer can be selected for row binary reading.**
- **−1440 Selects Interface 3**
  −1441 Selects Interface 1
  −1442 Selects Interface 2

The SCB instruction allows the 7040 system to read row binary cards from a 1402 card reader attached to the 1414-4 I/O synchronizer.

During the 7040 I cycle of the SCB instruction, the storage bus is gated into the storage register and program register at I3 time as a normal I time function (Figures 9 and 10).

POD 44 is decoded from the program register, and a I4(D1) pulse is sent to the 1414-4 I/O synchronizer to turn on the read column binary trigger.

The end-op trigger is turned on at I4.5 time and the SCB instruction completes its operation in one cycle.

There is no circuit interlocking for the SCB instruction. The read column binary trigger is reset each CB9 time. Therefore any SCB instruction issued prior to this time for the next card cannot be stacked, but, instead, will be reset and lost. To be effective for a particular card, the SCB instruction must also be issued not later than 7 milliseconds after the reset and load channel (RCH) instruction to the card reader.

The first card in the card reader can not be a row binary card because the contents are read and stored into the read buffers before the 7040 can exercise program control of the SCB instruction.

Note that the SCB instruction does not provide a unit address in positions 21-35. The SCB operation is, therefore, confined to the 1402 card reader attached to the 1414-4 synchronizer.

1402 Read Column Binary Operation

- **Both BCD and binary cards can be intermixed within the card deck.**
- **Each card hole represents a binary bit of information.**
- **Two 80-position buffers are required for the 160 6-bit characters read from the 960 holes in the card.**
- **Read-encoding circuitry is bypassed.**
- **Reader validity check circuitry is disabled when reading binary cards.**
- **A column binary identification signal is returned to the CPU for a sense (SEN) instruction data bit.**

Column binary cards are recognized by a 7-9 punch combination in column 1 of each column binary card as it passes the first reading station. This combination of punches turns on a read column binary trigger (latch) to condition card reading at the second reading station.

Row binary cards must also have all holes read as binary information. However, there are no specific 7-9 punches in column 1 to indicate binary information. This 7-9 punch condition is "simulated" by means of the SCB instruction which turns on the same read column binary trigger mentioned in the preceding paragraph.

Note that turning on the read binary trigger does not affect the data presently in the read buffers.

When the card is read at the second reading station, each column is treated as two 6-bit characters. Card rows 9 through 4 are treated as one character while rows 3 through 12 are treated as the other character in each card column.

Because the column binary (or row binary) card contains 160 characters, two 80-position read buffers are required. At the end of the card feed cycle rows 9 through 4 of the binary information are stored in the optional read buffer, and rows 3 through 12 are in the normal read buffer.

The column binary circuits remain active until after the transfer scan to the CPU. A normal read instruction starts the transfer scan, but a double-length transfer scan occurs because the column binary control circuits are not reset until the next reader 9 time.

Because the card holes are to be accepted as they are, normal BCD read encoding circuitry is blocked. In addition, the read validity checking circuitry is also blocked.

The normal signal from the read column binary trigger is also available to the 7040 sense (SEN) instruction and will place a bit in the low-order position of the status character (i.e., 000001).

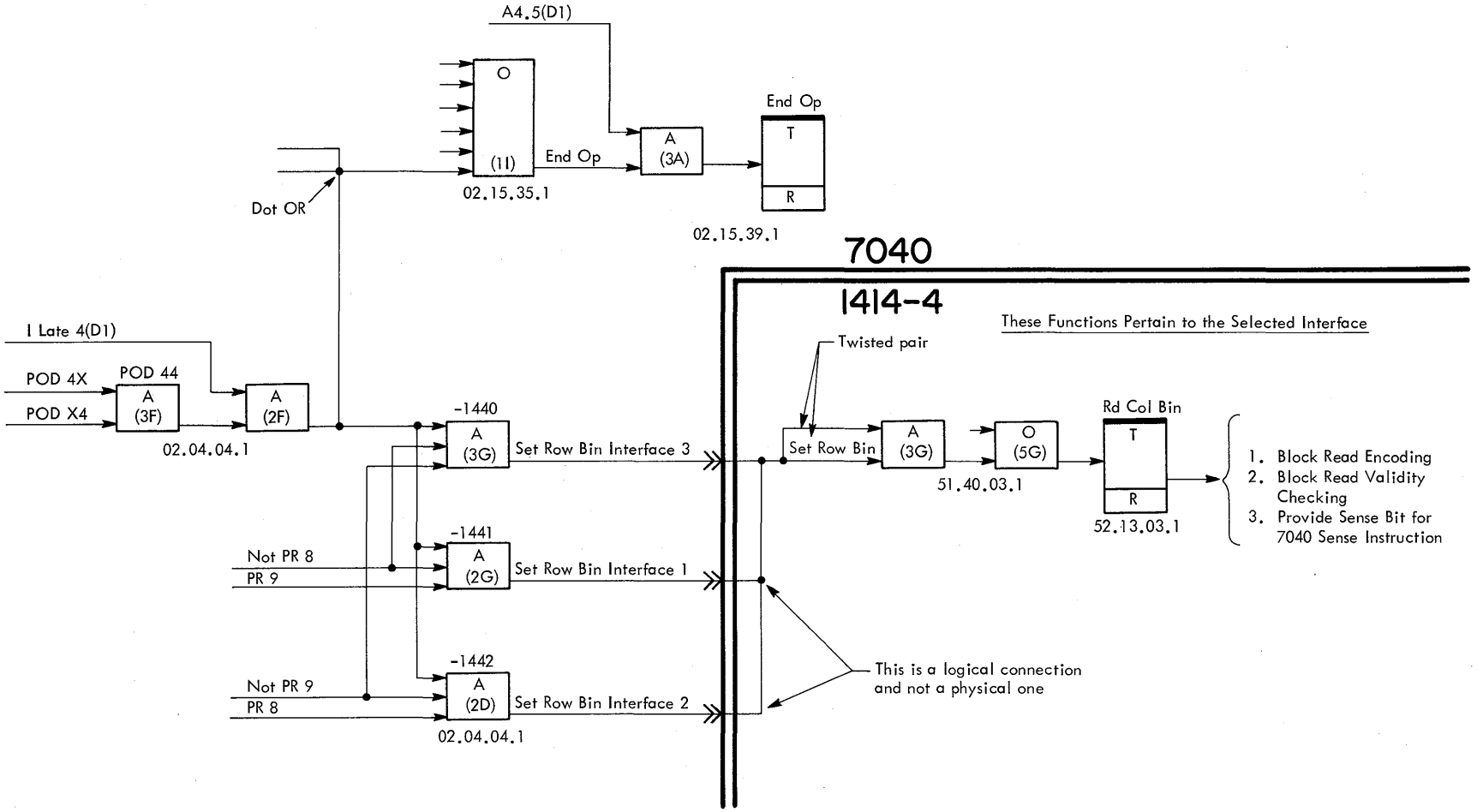


Figure 9. SCB Condensed Logic

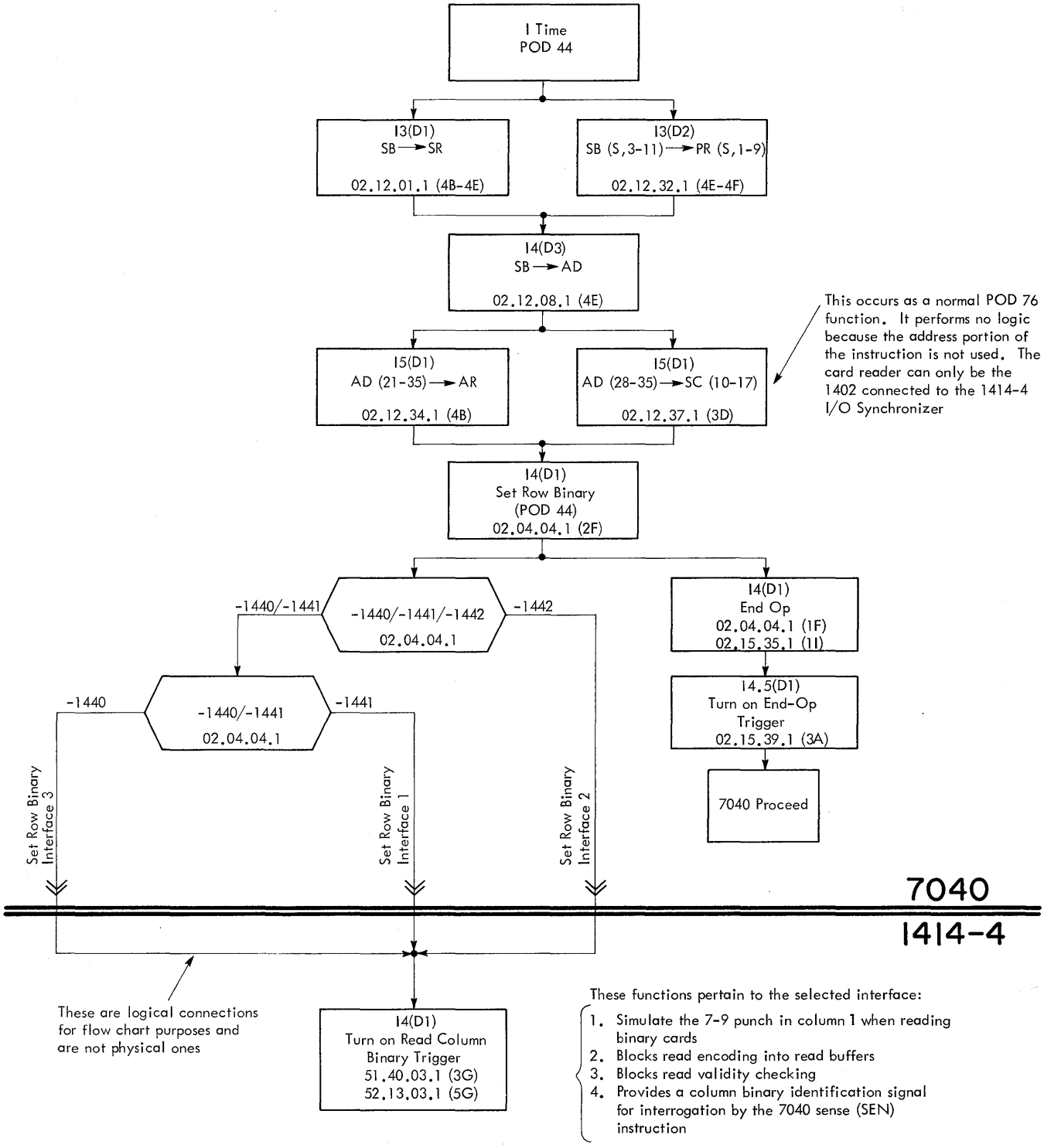


Figure 10. SCB Flow Chart

7040 Transmit Instruction TMT — 1704

- **Modified 7040 transmit (TMT) instruction.**
- **Number of data words transmitted depends on value set into shift counter by TMT instruction.**
- **"From" and "to" data addresses are specified by AC(3-17) and AC(21-35), respectively.**
- **Modifications:**
  AC(2) indicates "from" 7094;
  AC(20) indicates "to" 7094;
  AC(19) indicates compatibility transmit.
  Summary:
  7040 to 7040 — AC(2) = 0; AC(20) = 0
  7040 to 7094 — AC(2) = 0; AC(20) = 1
  7094 to 7040 — AC(2) = 1; AC(20) = 0
  7094 to 7094 — AC(2) = 1; AC(20) = 1.
- **A normal 7040 to 7040 transmit is performed if not in multiprocess mode.**

The object of the transmit instruction is to move blocks of data between core storage locations. The existing 7040 transmit instruction moves data between locations within the 7040 memory. For the multiprocess system, this instruction has been modified to allow the movement of data in the four logical combinations of:

1. 7040 to 7040
2. 7040 to 7094
3. 7094 to 7040
4. 7094 to 7094

The last three combinations allow the movement of data to or from the 7094. In this manner the 7040 can retrieve or send data to the 7094 in the course of program execution.

The 7094 core storage appears as an extension of the 7040 core storage while the 7040 appears as a data channel to the 7094. The transmission of data to and from the 7094 is, therefore, accomplished on a B cycle basis similar to 7607 or 7909 data channel operations.

When transmitting data, the 7040 storage register acts as the buffering register between the two systems. New storage bus gating and control circuitry have been added to the 7040 system to take care of this data buffering (Figure 11).

The 7094 memory selection is accomplished by additional gating and control circuitry at the 7040 address register. The content of the address register is gated to the 7094 channel address switch (CAS) and controls the operation similar to a 7607 or 7909 data channel (Figure 12).

The 7040, acting as a data channel, is restricted to the same priorities as any 7607 or 7909 data channel. One change has been made, however, to allow power-off conditions on the 7040 system without impairing stand-alone capabilities of the 7094. This change is the the addition of a reed relay (Figure 13). In the 7040 when power is on, R01 is held picked and the set of normally open points gate the "remote required" through normal circuitry. When power is removed from the 7040, R01 drops and the normally closed points allow the "remote required" signal to pass through unimpaired.

The compatibility transmit makes use of a new register added to the 7040. This PEAR (Program Effective Address Register) constantly monitors the progress of the 7094 program (Figure 14). At the time of a HIP halt, this register reflects the location of the last instruction executed. Knowledge of this address greatly increases the effectiveness and efficiency of the 7040 control program. Operation of this register will be discussed later in connection with the compatibility transmit instruction.

Note in reading the following sections that the various phases of the operations are explained in detail only at the first operation in which they occur. For example, the "to 7094" phase is explained in detail under 7040 to 7094 transmit. Only general references are made to this same phase under 7094 to 7094 transmit. For this reason the sections should be read in the order presented.

Two indicators have been added for transmit operations to and from the 7094:

INDICATION	LOCATION	SYSTEMS
Block 2	01D3B05-9	02.10.91.1
Block 20	01D3B03-9	02.10.91.1

7040 to 7040 Transmit

- **Normal existing transmit instruction.**
- **Number of words transmitted is dependent on value placed in shift counter by TMT instruction.**
- **Independent of multiprocess mode.**
- **"From" address AC(3-17) — "to" address AC(21-35); "From" 7040 AC(2) = 0 — "to" 7040 AC(20) = 0.**
- **If executed in multiprocess mode, AC(19) must equal 0.**

Objectives of the 7040 to 7040 transmit operation:

1. Decode transmit and set initial word count into shift counter.
2. Route "from" address from AC(3-17) to address register.
3. Select memory and route data word into 7040 storage register.
4. Route "to" address from AC(21-35) to address register.
5. Increment both "from" and "to" addresses (+1) in the adders.

6. Decrement shift counter (−1).

7. Select memory and initiate store cycle to send data word from storage register to 7040 MDR.

8. If SC ≠ 0 return to objective 2.

   If SC = 0, end operation.

Decoding of the transmit instruction (POD 70) takes place as soon as the program register is set through normal circuitry at I3 time (Figure 15, sheet 1, and Figure 16).

The storage register contents are routed to the main adders for address modification (if an index register is specified by the tag field), and AD(28-35) are then set into the shift counter. This shift counter value specifies the number of data words to be transmitted. Note that this count is treated modulo 377_8 (255_{10}). Transmission of larger blocks of data is accomplished by successive transmit instructions.

An L cycle follows the I cycle to provide an initial test of the shift counter (Figure 15, sheet 1). If the initial count equals zero, the transmit ends operation and proceeds to the next instruction in sequence. If the word count is not zero, the initial "from" address is routed from AC(3-17) to the storage register and, from there, left to AD(21-35) where a routing path exists to the address register. This "from" address is placed in the address register at L5 time which is sufficient to control memory selection for the following E cycle.

Two triggers in the 7040 control the fetching and storing of each data word. These triggers are, respectively, TMT read early and TMT read late. It is during the initial L cycle that the TMT read early trigger is turned on to send the 7040 into the first fetch cycle. Data will be transmitted during subsequent E cycles with two E cycles required for each data word.

Memory is selected at the next alpha early time and the data word is set into the storage register (See Figure 15 sheet 2, and Figure 16). With the data word in the storage register, the TMT read early phase is logically completed and the trigger turned off.

The TMT read late phase begins immediately by routing the "to" address to the address register for the store cycle memory selection. AC(1-35) is gated to the adders all during TMT read late time. AD(21-35), the "to" address is then routed and set into the address register for the next memory reference.

With the "to" address in the address register at E5 time and the accumulator still being gated into the adders, "hot 1's" are forced into AD(35) and both inputs of AD(18) during the next A1(D2) time. These 1's cause incrementing of both the "from" and "to" addresses, and the new addresses are returned to the accumulator until needed with the next data word.

Memory is selected at the next alpha early time and a store cycle is initiated to gate the data word from the storage register onto the storage bus and into the MDR at E2 time. (Figure 15, sheet 2; and Figure 16.)

The shift counter is decremented (−1) during the second E cycle at a time early enough to detect a zero condition and stop the operation.

If the word count is not zero, the new "from" address must be gated to the address register as was explained previously during the L cycle. AC(3-17) is gated to the storage register all during E early time. Note, however, that the "from" address is not set into the storage register until E3 time which is after the data word has been transferred to the MDR. AC(3-17) is then gated left to AD(21-35) and from there into the address register.

If the shift counter is not equal to zero, the TMT read early phase begins again with the next data word.

Note that when the transmit operation is completed, both addresses in the accumulator indicate the next memory location to be affected. This location could represent the starting location of a second transmit instruction. For example, to transmit a continuous block of 400 words requires two transmit instructions: TMT,255 and TMT,145. The accumulator needs to be loaded only once prior to the initial TMT,255 instruction.

7040 to 7094 Transmit

- **Effective only in multiprocess mode. (Interpreted as a 7040 to 7040 transmit when not in multiprocess mode.)**
- **Number of words transmitted is dependent on value placed in shift counter by TMT instruction.**
- **"From" address AC(3-17) — "to" address AC(21-35); "From" 7040 AC(2) = 0 — "to" 7094 AC(20) = 1.**
- **AC(19) must equal 0.**
- **7094 channel B times are used to transmit data to the 7094 memory.**

Objectives of the 7040 to 7094 transmit operation:

1. Decode transmit and set initial word count into shift counter.

2. Route "from" address from AC(3-17) to address register.

3. Select memory and route data word into 7040 storage register.

4. Route "to" address from AC(21-35) to address register.

5. Request B cycle from 7094.

6. Block 7040 memory selection until data word is stored in 7302 core storage.

7. Gate 7040 address register ("to" address) to channel address switches for 7094 memory selection.

8. Gate 7040 storage register onto 7094 channel storage bus for routing data word into core storage.

9. Increment both "from" and "to" address (+1) in the 7040 adders.

10. Decrement shift counter (−1).

11. If SC ≠ 0, return to objective 2.
    If SC = 0, end operation.

The initial I and L time functions are similar to the 7040 to 7040 transmit explained earlier (Figure 15, sheet 1). In addition, an AC(20) trigger is turned on from detecting a bit in AC(20). This trigger remains on throughout the operation as an indication to the system that data words are to be stored into the 7094 memory.

Note that the system must be in multiprocess mode to turn on the AC(20) trigger. If not in multiprocess mode, the AC(20) trigger remains off and the transmit instruction reverts to the normal 7040 to 7040 transmit operation. A timing chart of the operation is shown in Figure 17.

The 7040 data word is obtained from storage and is placed into the storage register. The "read early" phase is now completed; the read early trigger is turned off and the read late trigger is turned on (Figure 15, sheet 3).

At this point new circuitry comes into use. A data loaded latch is turned on as a logical signal, indicating to the operation that the 7040 storage register is loaded with a data word. This latch stays on until after the data word has been stored in the 7094.

The "to" address is routed from the accumulator to the address register so that it can be gated to the channel address switch to control 7094 memory selection (Figure 12).

A "block 20" condition is generated to supplement the "read late" phase of the operation and to define the operation: sending data to the 7094. The "block 20" implies operation with a block of data concerned with AC(20), and is not meant to imply a blocking (inhibiting) function.

This phase of the operation requires synchronizing with the 7094 and possible delays while other channels are being serviced on either or both systems. Because of these delays, the "read late" condition can not be conveniently reset as with the 7040 to 7040 transmit. It is the "block 20" condition which holds the "read late" phase until the data word is successfully stored in the 7094.

The 7040 memory select is blocked all during "block 20" (Figure 15, sheet 3). This blocking is necessary so that the data word can be properly stored in the 7094. If precautions were not taken, unwanted data could produce an OR'ing condition at the 7040 storage bus as the data word is being gated to the 7094.

Storing of the data word occurs during a conventional 7094 B cycle. The 7040 must wait until the B cycle can be granted by the 7094. Waiting may also be necessary because of priority conditions. Having obtained both B time and priority, the "to" address in the 7040 address register is gated to the 7094 channel address switches and from there to MAR in the 7302 core storage for address selection.

During the B cycle (Figure 15, sheet 7), the data word is gated out of the 7040 storage register and onto the 7094 channel bus. From here it is sampled into the MDR of the 7302 and into the selected core location.

With the data word now stored, the transmit operation must be synchronized again with 7040 circuitry to allow counter updating, and to obtain the next data word for transmission. Synchronization is initiated by turning on a data loaded sync trigger during the previous B cycle.

This, in turn, causes the data loaded latch to be reset at E4 time. The data load latch going off is an indication to the system that the storage register is no longer loaded (logically at least) with data. "Block 20" falls and allows the "TMT read late" trigger to be turned off at the next A3 time.

Before ending the "read late" phase, both the "from" and "to" counters are incremented through the 7040 adders with the new values being returned to the accumulator. Because of the longer time that TMT read late can be active, precautions must be taken to prevent extra stepping of these counters. The "allow increment" allows incrementing only once during the last portion of TMT read late.

At this point (Figure 17) memory cannot be selected because the new "from" address has not been placed in the address register. The next 7040 E cycle is, in effect, used as an updating cycle. During this time the shift counter is stepped (−1) to indicate the number of data words remaining for transmission. In addition, AC(3-17) is set into SR(3-17) and is then gated left to AD(21-35) as a routing path for the new "from" address into the address register.

With all counters properly updated and set, the "read early" phase is initiated by turning on the TMT read early trigger. Memory selection is allowed, and the next 7040 E cycle will be a fetch type cycle with a new data word being set into the storage register (Figure 15, sheet 3).

The operation continues in this manner until the shift counter is reduced to zero. At this time, the transmit operation ends and the computer proceeds with the next instruction in sequence.

7094 to 7040 Transmit

- **Effective only in multiprocess mode. (Interpreted as a 7040 to 7040 transmit when not in multiprocess mode.)**
- **Number of words transmitted is dependent on value placed in shift counter by TMT instruction.**
- **"From" address AC(3-17) — "to" address AC(21-35); "From" 7094 AC(2) = 1 — "to" 7040 AC(20) = 0.**
- **AC(19) must be equal to 0.**
- **7094 channel B times are used to transmit data to the 7040 storage register.**

Objectives of the 7094 to 7040 transmit operation:

1. Decode transmit and set initial word count into shift counter.
2. Route "from" address from AC(3-17) to address register.
3. Request B cycle from 7094.
4. Block 7040 memory selection and parity checking circuitry until after the data word has arrived from the 7094.
5. Gate 7040 address register ("from" address) to channel address switches for 7094 memory selection.
6. Gate 7094 channel storage bus (data word) into the 7040 storage register.
7. Route "to" address from AC(21-35) to address register.
8. Increment both "from" and "to" address (+1) in the 7040 adders.
9. Decrement shift counter (−1).
10. Select memory and initiate store cycle to send data word from storage register to 7040 MDR.
11. If SC ≠ 0, return to objective 2.
    If SC = 0, end operation.

The initial I and L time functions are similar to the 7040 to 7040 transmit explained earlier (Figure 15, sheet 1). In addition, an AC(2) trigger is turned on from detecting a bit in AC(2). This trigger remains on throughout the operation as an indication to the system that data words are to be obtained from the 7094.

Note that the system must be in multiprocess mode to turn on the AC(2) trigger. If not in multiprocess mode, the AC(2) trigger remains off and the transmit instruction reverts to the normal 7040 to 7040 transmit operation. A timing chart of the 7094 to 7040 operation is shown in Figure 18.

A "block 2" condition (Figure 15, sheet 4) is generated to supplement the read early phase of the operation and to define the operation: obtaining data from the 7094. The "block 2" implies operation with a block of data concerned with AC(2) and is not meant to imply a blocking (inhibiting) function.

This phase of the operation requires synchronization with the 7094 and possile delay while the other data channels are being serviced on either or both systems. It is the "block 2" condition which holds the read early phase until the data word is successfully set into the storage register of the 7040.

7040 memory select is blocked all during "block 2." If selection were not stopped, unwanted data could produce an OR'ed condition at the 7040 storage bus as the 7094 data word is being gated into the storage register.

Obtaining the data word requires a conventional 7094 B cycle. After receiving both B time and priority, the "from" address in the 7040 address register is gated to the 7094 channel address switches for address selection. See Figure 15, sheet 4, and Figure 18.

During the B cycle, the data word is read from the 7302 core storage and is placed on the 7094 channel storage bus. The 7094 channel storage bus is then gated onto the 7040 storage bus in preparation for setting into the 7040 storage register (Figure 12). Note that this storage register gating occurs from B1-B10 time. During this period of time, several cell drive pulses will set the storage register (System 02.15.20.1). Even though some of the early pulses will sample invalid data, a valid 7094 data word setting is always insured by at least the last cell drive pulse under the gate.

The data word arriving from the 7094 is capable of having either even or odd parity because a parity bit is not included. A 7040 store cycle condition is forced under these circumstances to suppress the parity checking circuits and thereby prevent false error indications.

With the data word now in the storage register, the read early phase is logically completed. However, the operation must be synchronized with the 7040 circuitry to allow proper storing of the data.

A data loaded sync trigger was turned on during the later portion of the 7094 B cycle. This, in turn, caused the data loaded latch to be turned on at the next 7040 E2 time (Figure 15, sheet 4). The data loaded latch coming on is a logical indication to the system that the read early phase is completed. As a result, "block 2" falls, the TMT read late trigger is turned on, and the TMT read early trigger is turned off.

At this point (Figure 18) a slippage of one 7040 E cycle occurs, allowing the "to" address to be moved from AC(21-35) to the address register (Figure 15, sheet 5). Following the setting of the address register, both the "from" and "to" counters are incremented in the adders and are returned to the accumulator for future use.

The next E cycle is provided with a memory selection, and the data word is stored into the "to" address by gating the contents of the storage register into the MDR.

The shift counter is also stepped (−1) during this store E cycle at a time early enough to detect a zero condition and to stop the transmit operation from continuing. If the word count is not zero, the new "from" address is routed from AC(3-17) to the address register via the storage register and adders (Figure 15, sheet 5).

With a word count not equal to zero, the TMT read early trigger is turned on and a read early phase is initiated by requesting another data word from the 7094 (Figure 15, sheet 4).

7094 to 7094 Transmit

- **Effective only in multiprocess mode. (Interpreted as a 7040 to 7040 transmit when not in multiprocess mode.)**
- **Number of words transmitted is dependent on value placed in shift counter by TMT instruction.**
- **"From" address AC(3-17) — "to" address AC(21-35); "From" 7094 AC(2) = 1 — "to" 7094 AC(20) = 1.**
- **AC(19) must equal 0.**
- **7094 channel B times are used to transmit data from and to the 7094.**

Objectives of the 7094 to 7094 transmit operation:

1. Decode transmit and set initial word count into shift counter.
2. Route "from" address from AC(3-17) to address register.
3. Request B cycle from 7094.
4. Block 7040 memory selection and parity checking circuitry until after the data word has arrived from the 7094.
5. Gate 7040 address register ("from" address) to channel address switches for 7094 memory selection.
6. Gate 7094 channel storage bus (data word) into the 7040 storage register.
7. Route "to" address from AC(21-35) to address register.
8. Request B (store) cycle from 7094.
9. Continue to block 7040 memory selection until after the data word has been sent to the 7302 core storage.
10. Gate 7040 address register ("to" address) to the channel address switches for 7094 memory selection.
11. Gate 7040 storage register onto 7094 channel storage bus for routing data word into core storage.
12. Increment both "from" and "to" addresses (+1) in the 7040 adders.
13. Decrement shift counter (−1).
14. If SC ≠ 0, return to objective 2.
    If SC = 0, end operation.

The initial I and L time functions are similar to the 7040 to 7040 transmit explained earlier (Figure 15, sheet 1). In addition, AC(2) and AC(20) triggers are turned on from detecting bits in AC(2) and AC(20). These triggers remain on throughout the operation as an indication to the system that data words are to be both obtained and stored in the 7302 core storage on the 7094 system.

Note here that the system must be in multiprocess mode to turn these triggers on. If not in multiprocess mode, the triggers remain off and the transmit instruction is interpreted as a normal 7040 to 7040 transmit operation. A timing chart of the 7094 to 7094 operation is shown in Figure 19.

The read early phase is similar to that explained previously for the 7094 to 7040 transmit operation (Figure 15, sheet 4). "Block 2" defines the operation as one of obtaining data from the 7094. A 7094 B cycle is requested; during this time the 7040 address register ("from" address) selects memory and the data word is set into the 7040 storage register. The 7040 memory select circuitry is blocked and parity circuits are suppressed to prevent false parity error indications.

After the data word has been received, a 7040 E cycle is required for routing the "to" address to the address register. Synchronization with the 7040 is necessary to obtain these E time pulses.

At this point the read late phase begins (Figure 15, sheet 6). "Block 20" now defines the operation as one of sending data to the 7094. During the same time that a B cycle is requested, the 7040 address register ("to" address) selects memory and the data word is placed on the 7094 channel storage bus and is set into the MDR of the 7302 core storage (Figure 15, sheet 7). A "channel store control" generated by the 7040 causes the data to be stored. The 7040 memory select circuitry is also suppressed during the read late phase to prevent unwanted data from appearing on the 7040 storage bus.

With the transmission of this data word completed, synchronization must be obtained again with the 7040 so that E time pulses can be used for incrementing the "from" and "to" counters (+1) and decrementing the shift counter (−1). Because of the longer time that TMT read late can be active, precautions must be taken to prevent extra stepping of these counters. "Allow increment" allows incrementing only during the last portion of TMT read late. The counters are therefore incremented the one time as required (Figure 15, sheet 7).

With all counters properly updated and the new "from" address in the address register, a new read early phase is initiated to obtain a new data word from the 7094 (Figure 15, sheet 4).

The operation continues until the shift counter is reduced to zero (Figure 15, sheet 7). At this time the transmit operation ends and the computer proceeds with the next instruction in sequence.

Compatibility Transmit

- **Used with HIP mode of operation.**
- **Must be executed only when the 7094 is halted.**
- **Effective only in multiprocess mode. (Interpreted as a normal 7040 to 7040 transmit when not in multiprocess mode.)**
- **Word count of 2 must be specified by TMT instruction.**
- **"From" address AC(3-17) not used — "to" address AC(21-35); "From" 7094 AC(2) = 1 — "to" 7040 AC(20) = 0.**
- **AC(19) = 1.**
- **PEAR register in 7040 used as both 7094 data "from" address, and source of data.**
- **The two data words stored into the 7040 are:**
  1. **The 7094 instruction causing the trap (S, 1-35).**
  2. **The modified address portion (21-35) of the instruction causing the trap.**

When the 7094 is operating in HIP mode, the program is almost constantly detecting I/O instruction and halting with a trap signal to the 7040. It is then the responsibility of the 7040 to interrogate the 7094 to determine the instruction causing the halt.

Determining all of the facts about the instruction by conventional trap procedures can be unwieldy and time consuming. For example, take the following 7094 program:

```
00477
00500    WBTA 1
00501    RCHA, 600, 2 (assume XR2 = 5)
00502
  {
00573    IOCD, Data,  , 25
```

After the first HIP halt (due to the WTBA 1) the 7094 program counter contains 00501. By conventional trapping methods, this value can be obtained by trapping the 7094 and then transmitting location 00003 into the 7040. Another transmit instruction then transmits location 00500 into the 7040 where a table-lookup is performed to determine the instruction and I/O unit specified. Having performed this testing, 7094 locations 00003 and 00004 must be modified to cause the 7094 to transfer to location 00501 when started.

The RCHA instruction in location 00501 causes a second HIP halt and trap to the 7040. Again the 7094 must be trapped so that location 00501 can be transmitted into the 7040 for table-lookup. An additional problem now exists in obtaining the channel IOCD command because the address is modified by an index register. Not only must the contents of the index register be obtained from the 7094, but accounts must be made for either 3XR or 7XR mode. All of these require "save" and modification routines in both the 7040 and 7094.

The new PEAR register in the 7040 in conjunction with the compatibility transmit instruction allows many of these previous routines to be eliminated with a significant increase in program speed.

For each I/O HIP trap, the 7040 program requires two basic items: (1) the instruction causing the trap; and (2) the modified address portion of the instruction, i.e., positions (21-35) as modified by the index registers. The PEAR (Program Effective Address Register) register is constantly being set to the location of the instruction being executed in the 7094. Therefore, at a HIP halt, the PEAR register can act as a transmit "from" address to the 7094 for obtaining the desired instruction. At the time of the HIP halt, all address modification has also been performed on the instruction with the modified address in the 7094 address register. This modified address may be stored in the 7040 by first gating the 7094 address register into the PEAR register, and then gating the PEAR register (as data) on the storage bus to the 7040 memory. The two 7040 storage locations are specified by AC(21-35). The final setting of the two 7040 locations for each of the two compatibility transmits in the previous program example are:

(a)	TMT	+ 0760	0	01221
	TMT + 1			01221
(b)	TMT	+ 0540	2	00600
	TMT + 1			00573

Setting the PEAR Register

- **PEAR register continually reflects location of the 7094 instruction being executed.**
- **During compatibility transmit the PEAR register contains:**
  1. **Address of 7094 instruction causing trap.**
  2. **Modified address portion (21-35) of instruction causing the trap.**

The PEAR register in the 7040 is constantly reflecting the progress of the 7094 (or 7090) program. For every I cycle, either the program counter or address register is gated to MAR for memory selection of the next instruction to be executed (See Figure 14).

During this period of time (A10 dlyd to I3) when the MAR gate is active, the address is also available on the channel MAR bus. By generating a 7094 I2(D1) as a set

PEAR register pulse, the PEAR register will always indicate the location of the current 7094 instruction.

Store overlap (SLA) on the 7094 requires a special "set PEAR register" pulse at E3(D3) so that the PEAR register can be set to the overlapping instruction. An example of a program sequence is given:

```
00200    STO     Data
00201    WTBA    1
```

In this case the WTBA is an overlapping instruction which occurs under SLA conditions and will cause a HIP halt and trap. The PEAR register was set to 00200 at I2 time of the STO instruction. At I8 time of the STO instruction the program counter is stepped (+1) to indicate location 00201: the location of the WTBA instruction. During an SLA operation, cyclic makeup for the above sequence is I, E/I where a cyclic transition is made from E6 time of the STO instruction to I6 time of the WTBA. Note that there exists no normal I2(D1) pulse to set the PEAR register to 00201. This problem is overcome (Figure 14) by making use of E3(D3) time to force the program counter to MAR and at the same time generate the necessary set PEAR register pulse to the 7040.

Prior to any HIP halt, address modification is performed on the I/O instruction causing the halt and the new modified address is retained in the address register. When the 7094 halts, this value is gated to the MAR bus every A10 dlyd to A3 time by the MAR gate trigger and circuitry. During the latter phase of the compatibility transmit operation, this gating is used to set the PEAR register.

Compatibility Transmit Operation

- **Basically a 7094 to 7040 transmit operation.**
- **"From" address not used in AC(3-17).**
- **PEAR register supplies "from" address for first phase of operation.**
- **A word count greater than 2 causes additional unwanted data to be stored into the 7040.**

Objectives of the compatibility transmit operation are:

1. Decode transmit and set initial word count of 2 into shift counter.
2. Request B cycle from 7094.
3. Block 7040 memory selection and parity checking circuitry until after the data word has arrived from the 7094.
4. Gate 7040 PEAR register ("from" address) to channel address switches for 7094 memory selection.
5. Gate 7094 channel storage bus (7094 instruction which caused trap) into the 7040 storage register.
6. Route "to" address from AC(21-35) to address register.
7. Increment both "from" and "to" address (+1) in the 7040 adders. Incrementing AC(3-17) performs no logic.
8. Decrement the shift counter (−1) to 1.
9. Select memory and initiate store cycle to send data word from storage register to 7040 MDR.
10. Signal the 7094 to gate the 7094 address register (modified address portion of instruction causing trap) into the 7040 PEAR register.
11. Request B cycle from 7094. This is really a dummy cycle mainly for housekeeping purposes.
12. Gate the 7040 PEAR register into the 7040 storage register.
13. Route "to" address from AC(21-35) to the address register.
14. Increment both "from" and "to" address (+1) in the 7040 adders. Incrementing AC(3-17) performs no logic.
15. Decrement the shift counter (−1) to 0.
16. Select memory and initiate store cycle to send data word from storage register to 7040 MDR.
17. End operation and proceed to next instruction in sequence.

The compatibility transmit instruction is basically the same as a 7094 to 7040 transmit. Because of this, AC(2) must equal 1; AC(20) must equal 0; and AC(19) must equal 1. Settings other than these will cause the operation to function differently.

During the first read early phase, the PEAR register, instead of the address register, is gated to the 7094 for memory selection (Figure 15, sheet 8). The PEAR register contains the address of the 7094 instruction which caused the trap. The 7094 B cycle places the instruction on the channel storage bus where it is sampled into the 7040 storage register.

Following this B cycle, the "to" address is routed to the address register and the 7040 goes into a store cycle (first "read late" phase) to store the instruction from the 7094 (Figure 15, sheet 9). Both the "from" and "to" counters are incremented (+1); incrementing the "from" address, AC(3-17), provides no logic but occurs as a matter of course.

The operation now starts the second and final read early phase. The shift counter is decremented from 2 to 1, causing SC(17) to be turned on. This condition sends a control line, "4X BIT 19," to the 7094 so that the 7094 address register can be gated into the 7040 PEAR register. This value represents the modified address portion of the 7094 instruction causing the trap and will be treated as data by the 7040. Note that more than one "set PEAR register" pulse can occur prior to the time that the PEAR register is actually used. (Figure 20).

Even though not logically necessary, the second 7094 B cycle is requested and executed to take advantage of the normal 7094 to 7040 housekeeping and gating functions. No address is sent to the 7094 for memory selection (Figure 15, sheet 8), and no data word is gated into the 7040 storage register from the 7094. Instead, the "90 to 40" gating line is used to set the PEAR register into the 7040 storage register. Remember that the contents of the PEAR register are to be treated as data this time and stored into the 7040 memory. Note also that the PEAR register can not be modified during the 7094 B cycle because of degating in the 7094 (Figure 20).

Following the B cycle, the second and final "read late" phase starts (Figure 15, sheet 9). The "to" address is routed to the address register and the 7040 goes into a store cycle to store the modified address from the 7094. Both "from" and "to" counters are incremented (+1) and the shift counter decrements (−1) from 1 to 0. This zero condition indicates the end of the operation, and the 7040 proceeds to the next instruction in sequence.

Note that circuit control of the compatibility transmit is based on SC(17) as an indication of the word count. Therefore, an initial word count greater than two will cause more than two words of data to be stored.

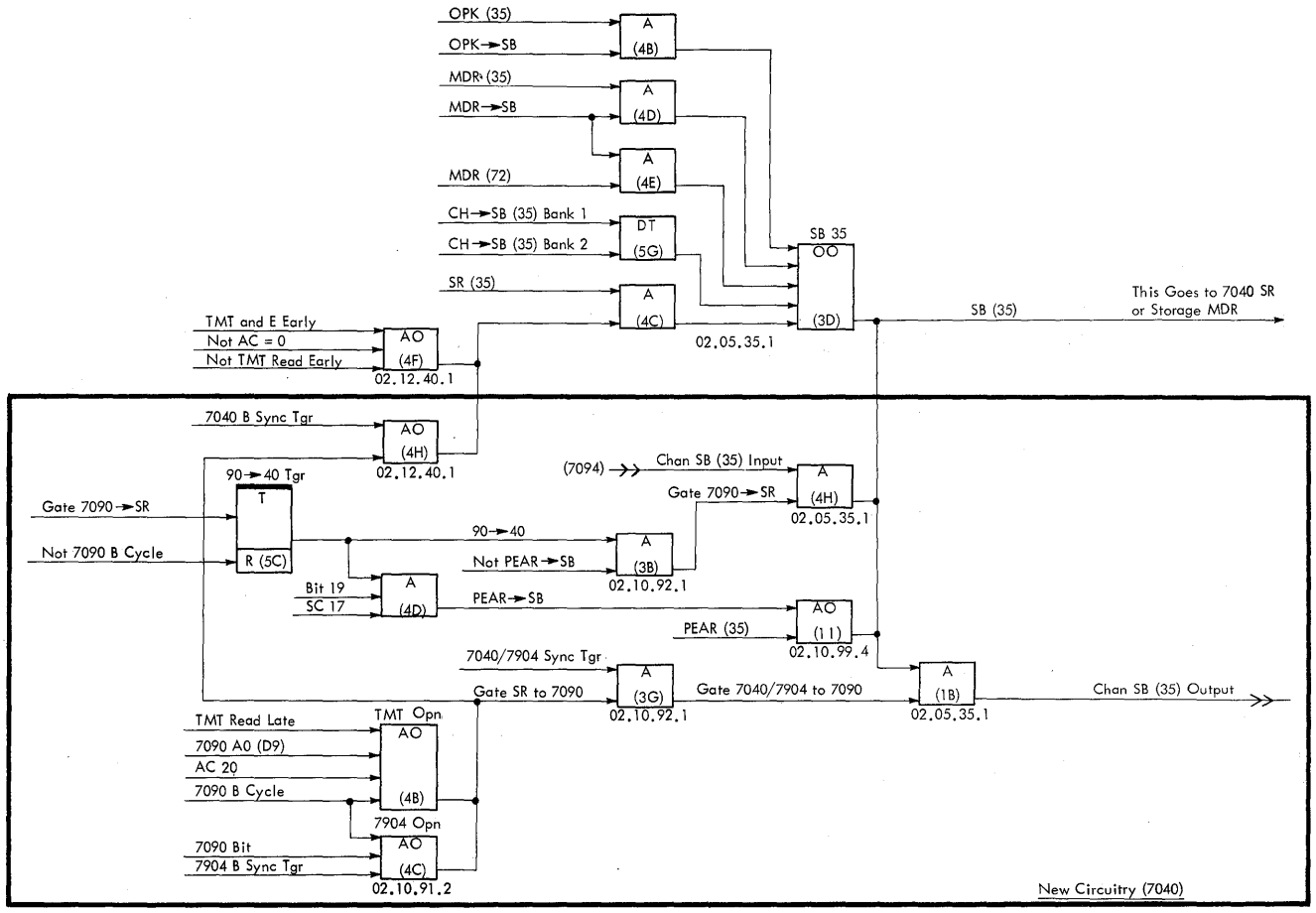


Figure 11. 7040 Storage Bus Gating

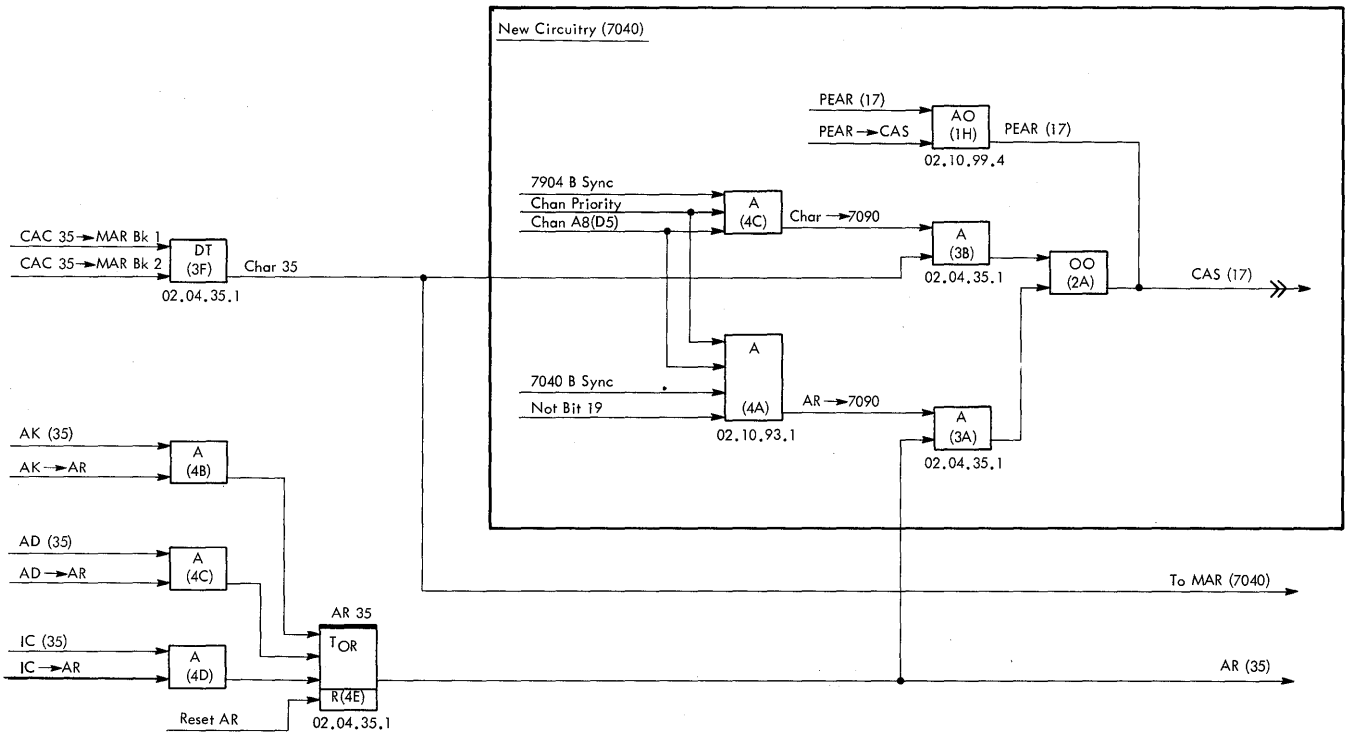


Figure 12. 7040 to 7094 Memory Select Circuitry

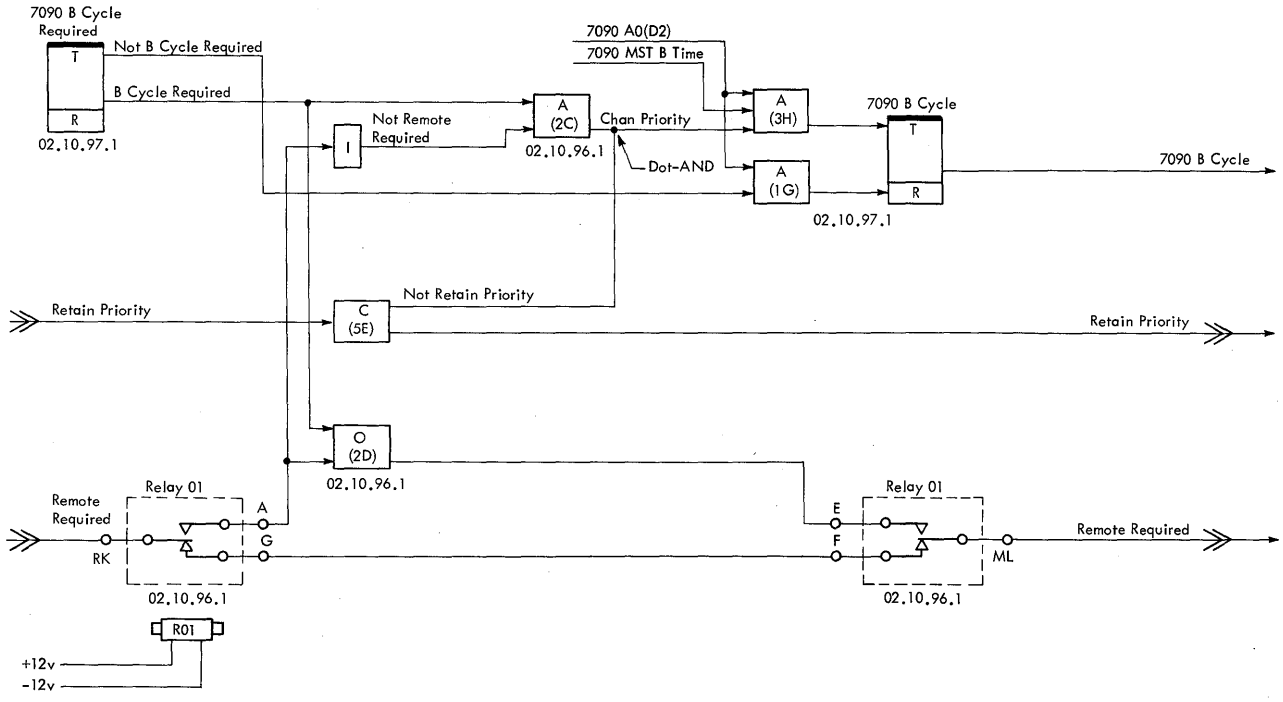


Figure 13. 7040 Multiprocess Channel Priority

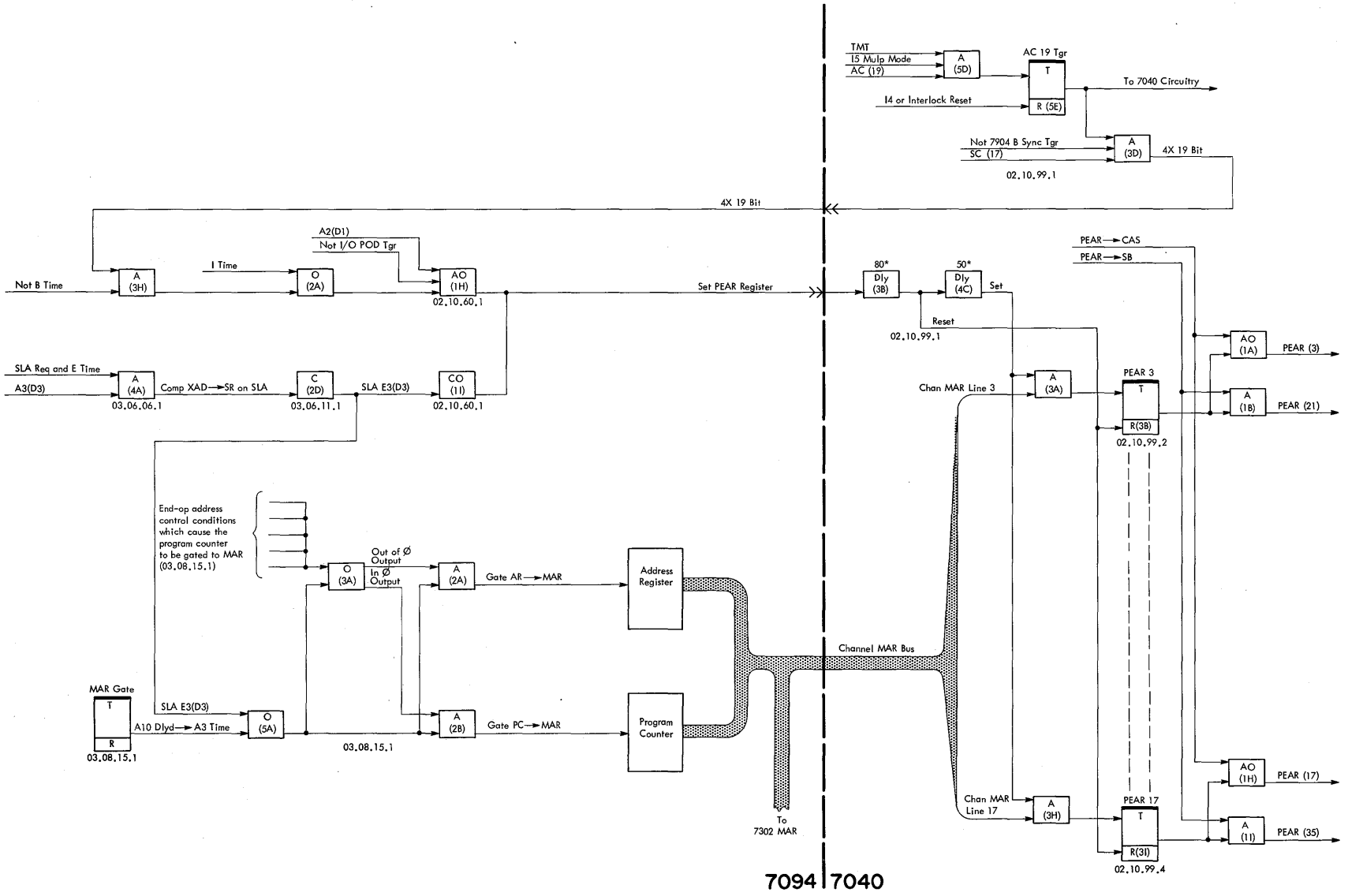


Figure 14. PEAR Register Controls

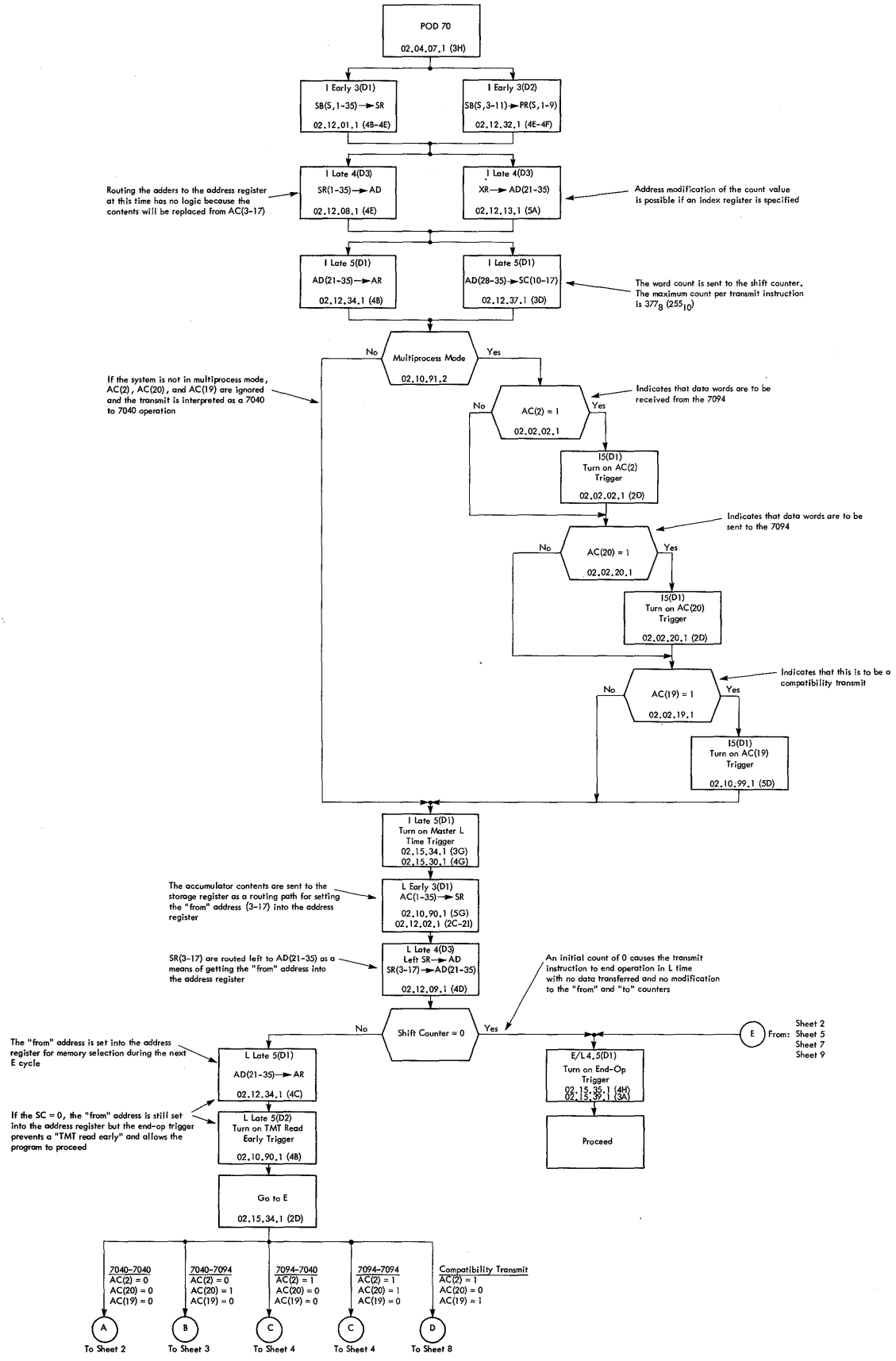


Figure 15. Transmit Flow Chart (Sheet 1 of 9)

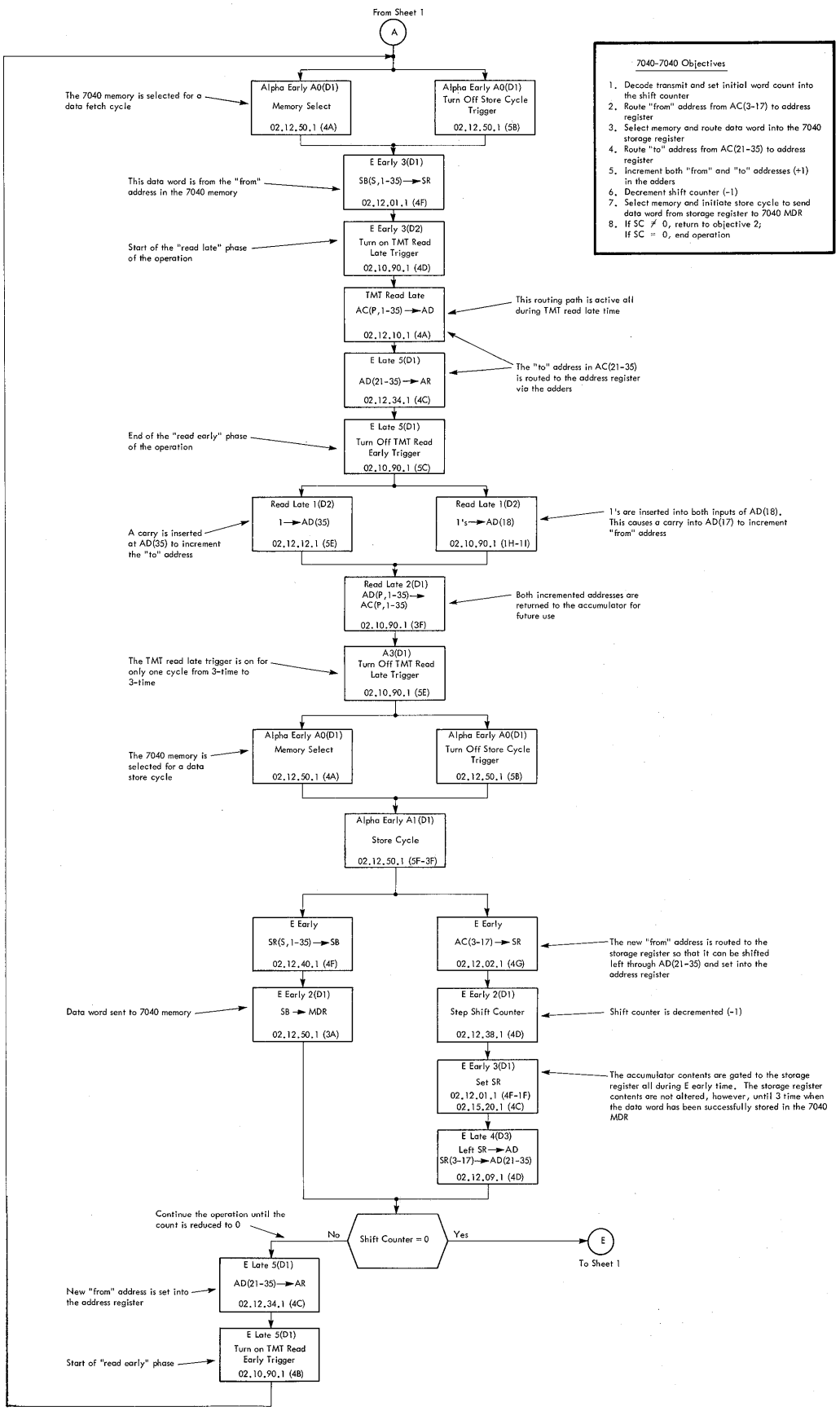


Figure 15. Transmit Flow Chart (Sheet 2 of 9)

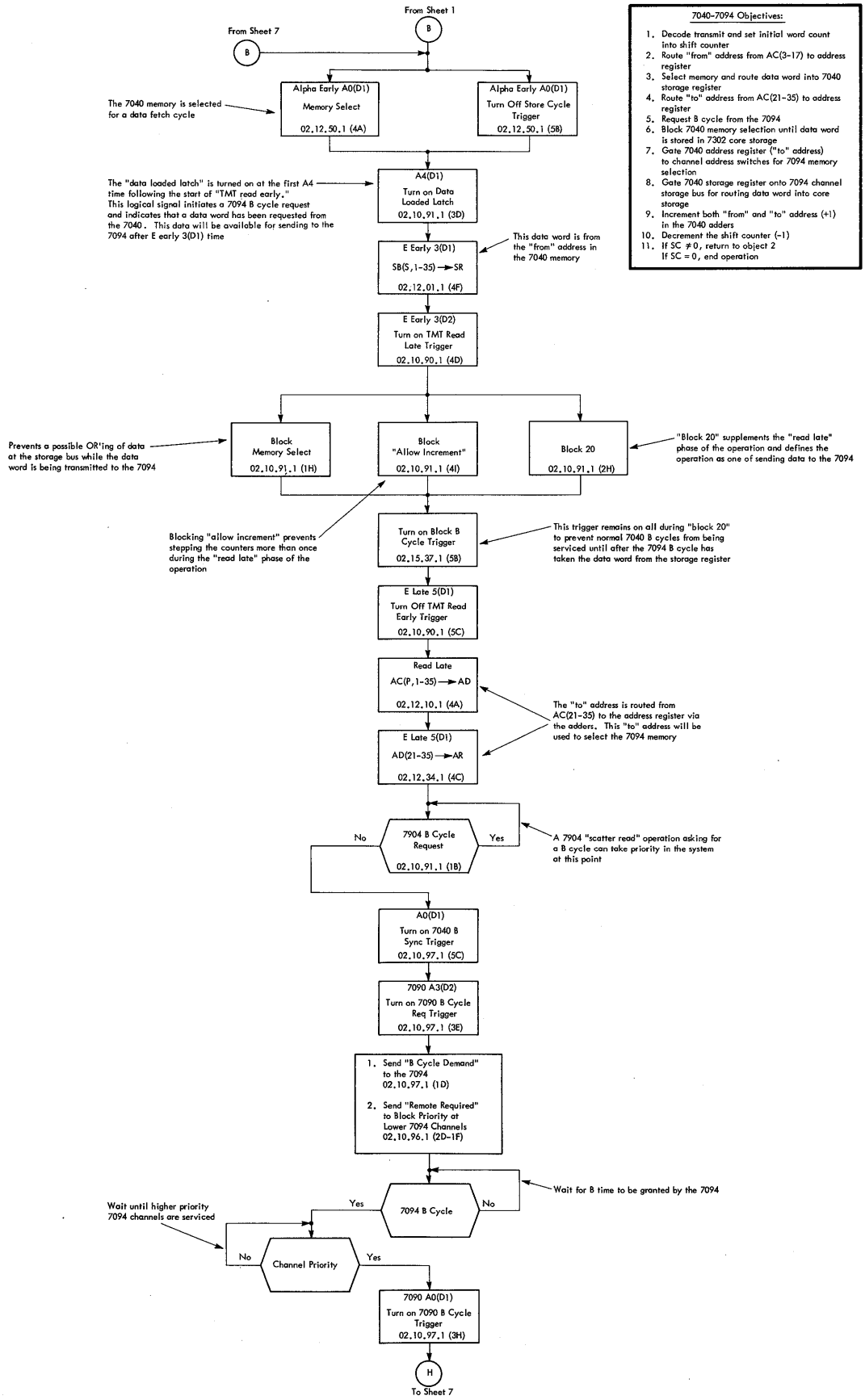


Figure 15. Transmit Flow Chart (Sheet 3 of 9)

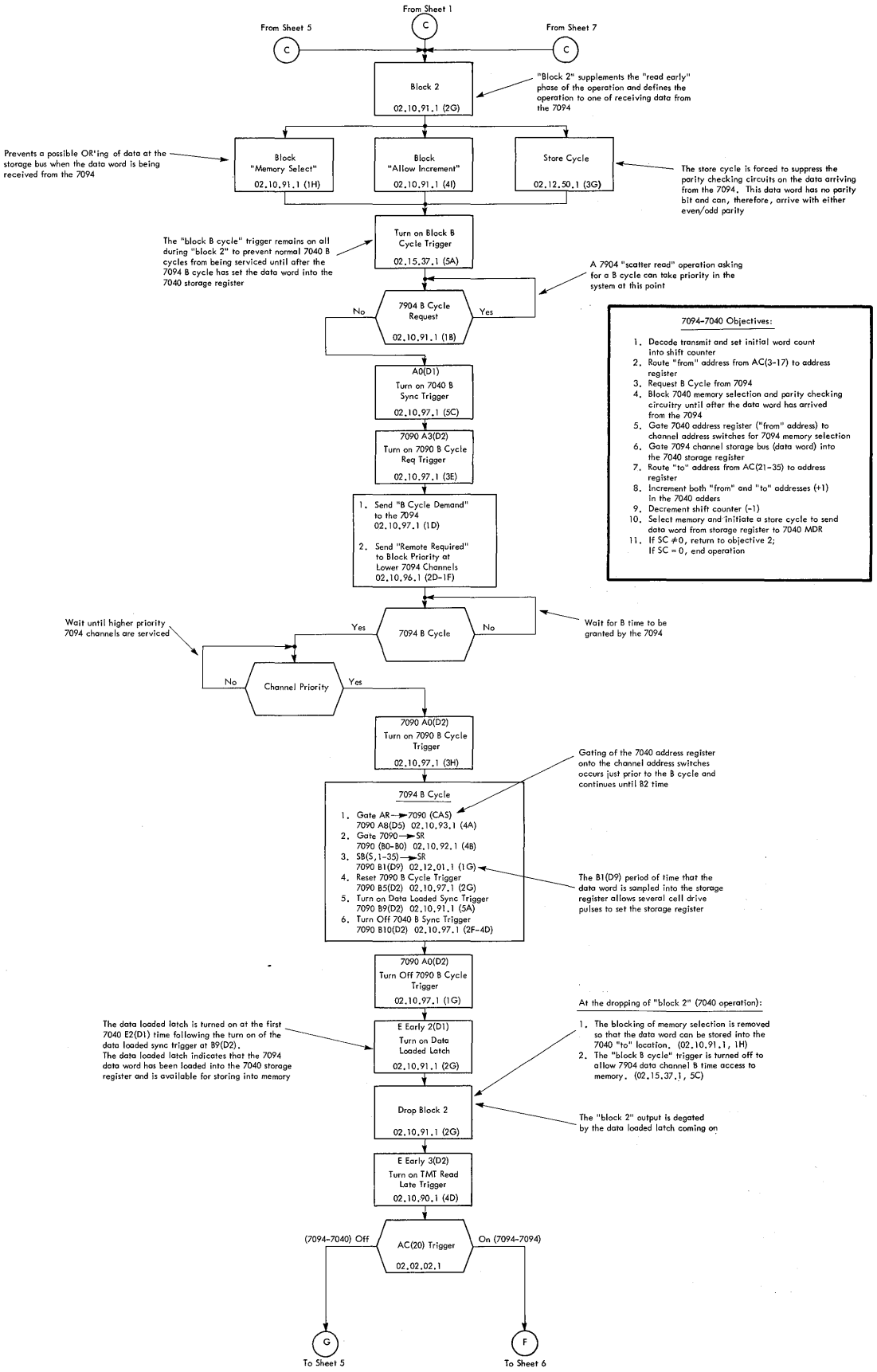


Figure 15. Transmit Flow Chart (Sheet 4 of 9)

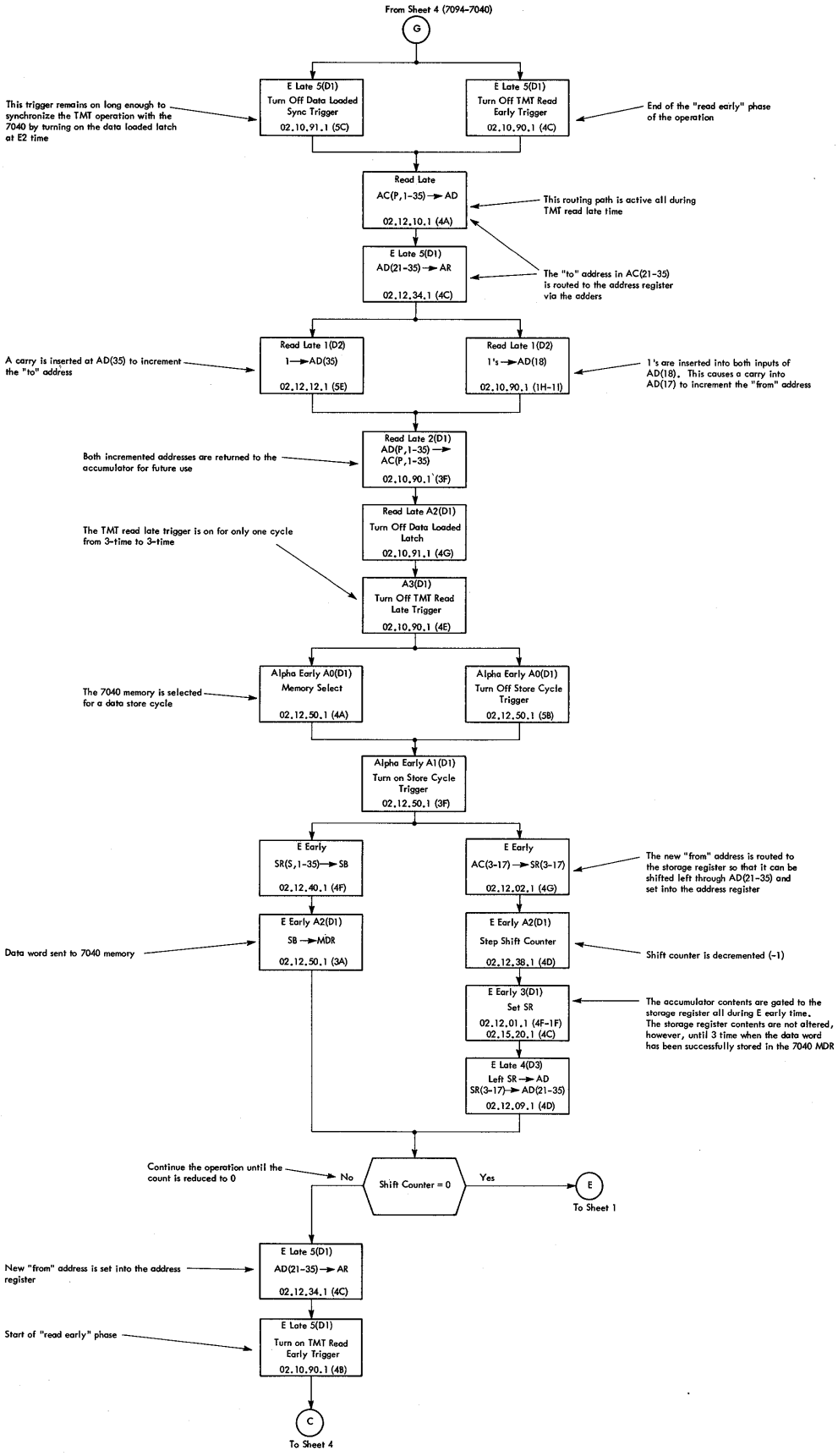


Figure 15. Transmit Flow Chart (Sheet 5 of 9)

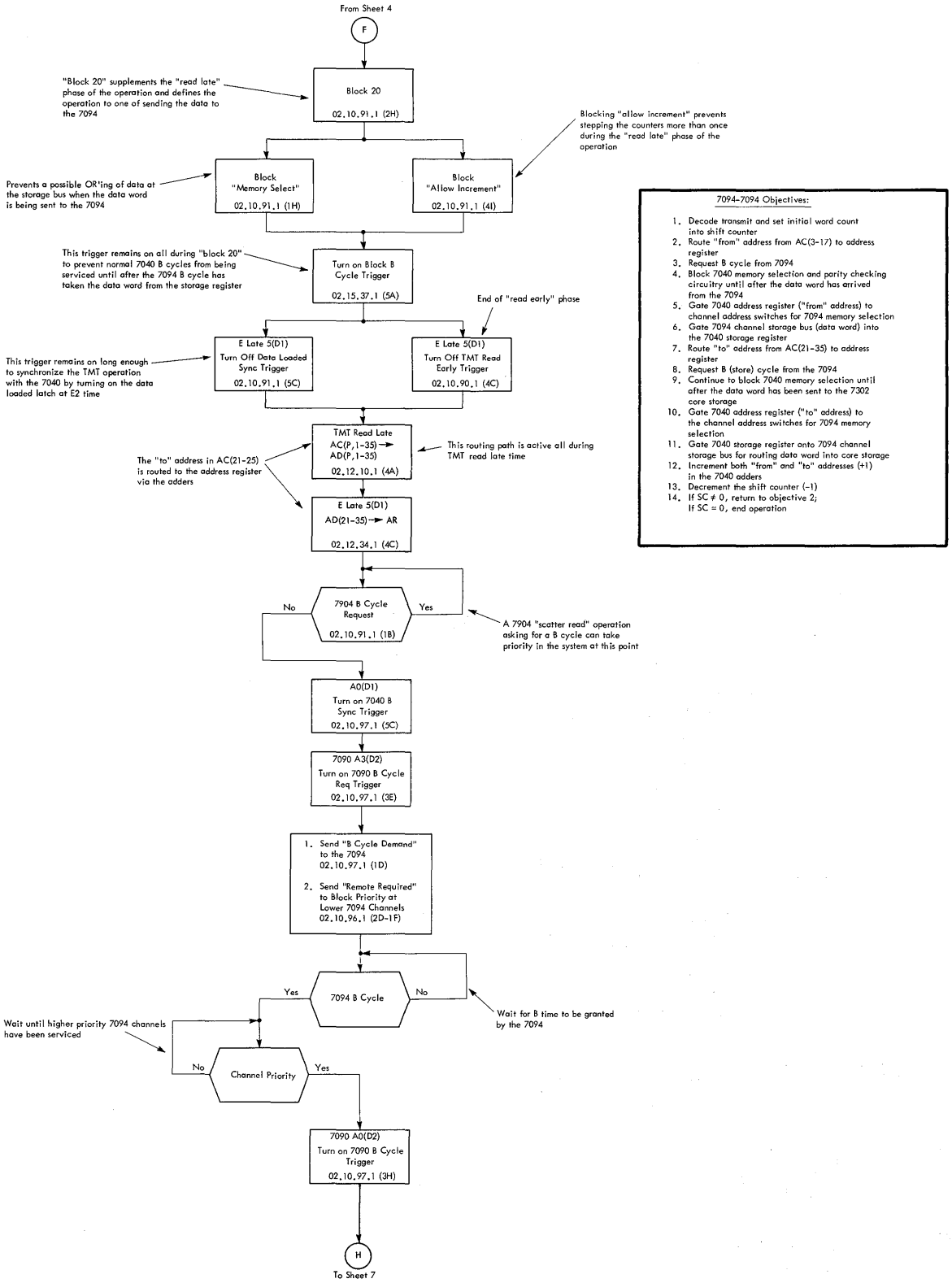


Figure 15. Transmit Flow Chart (Sheet 6 of 9)

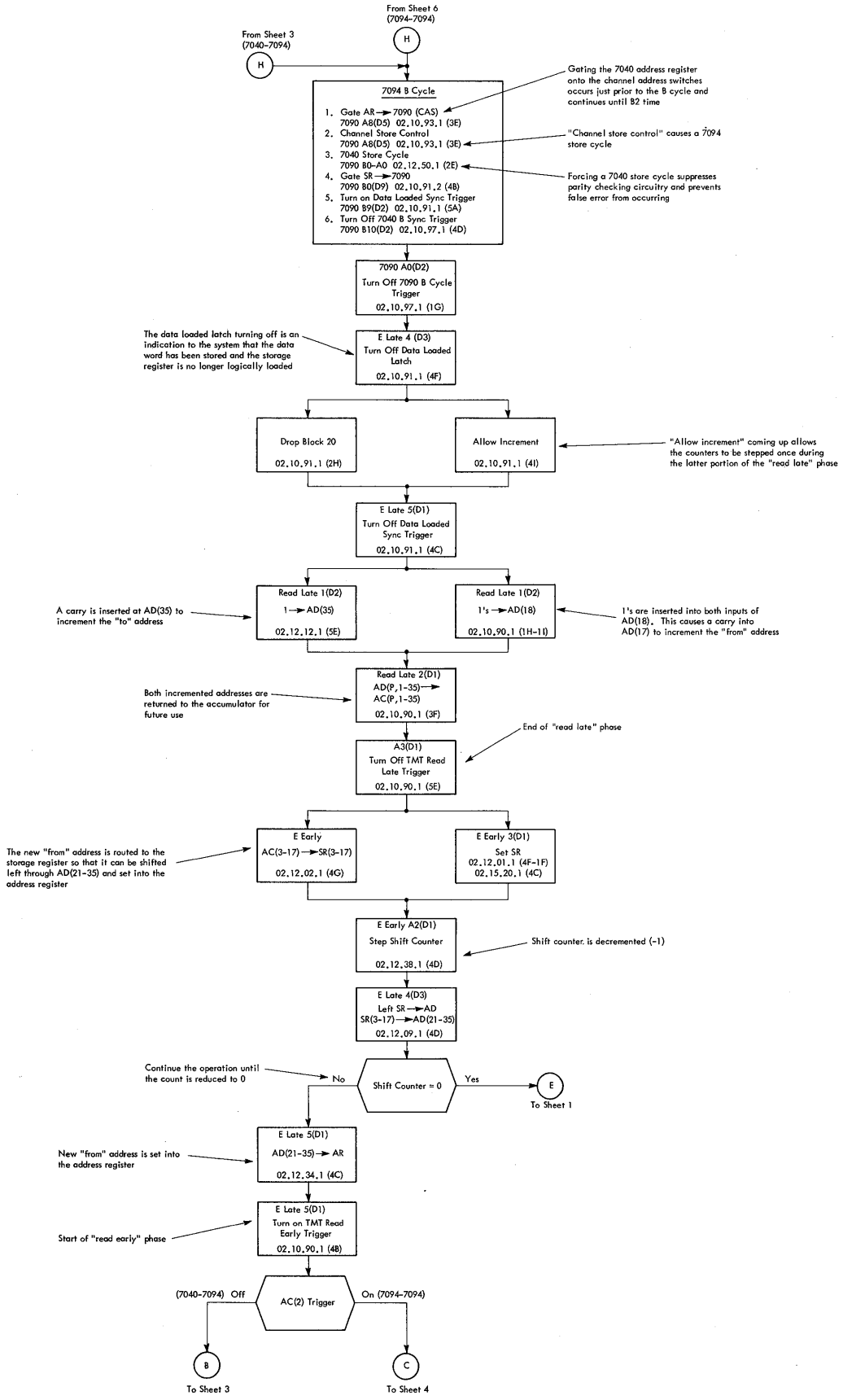


Figure 15. Transmit Flow Chart (Sheet 7 of 9)

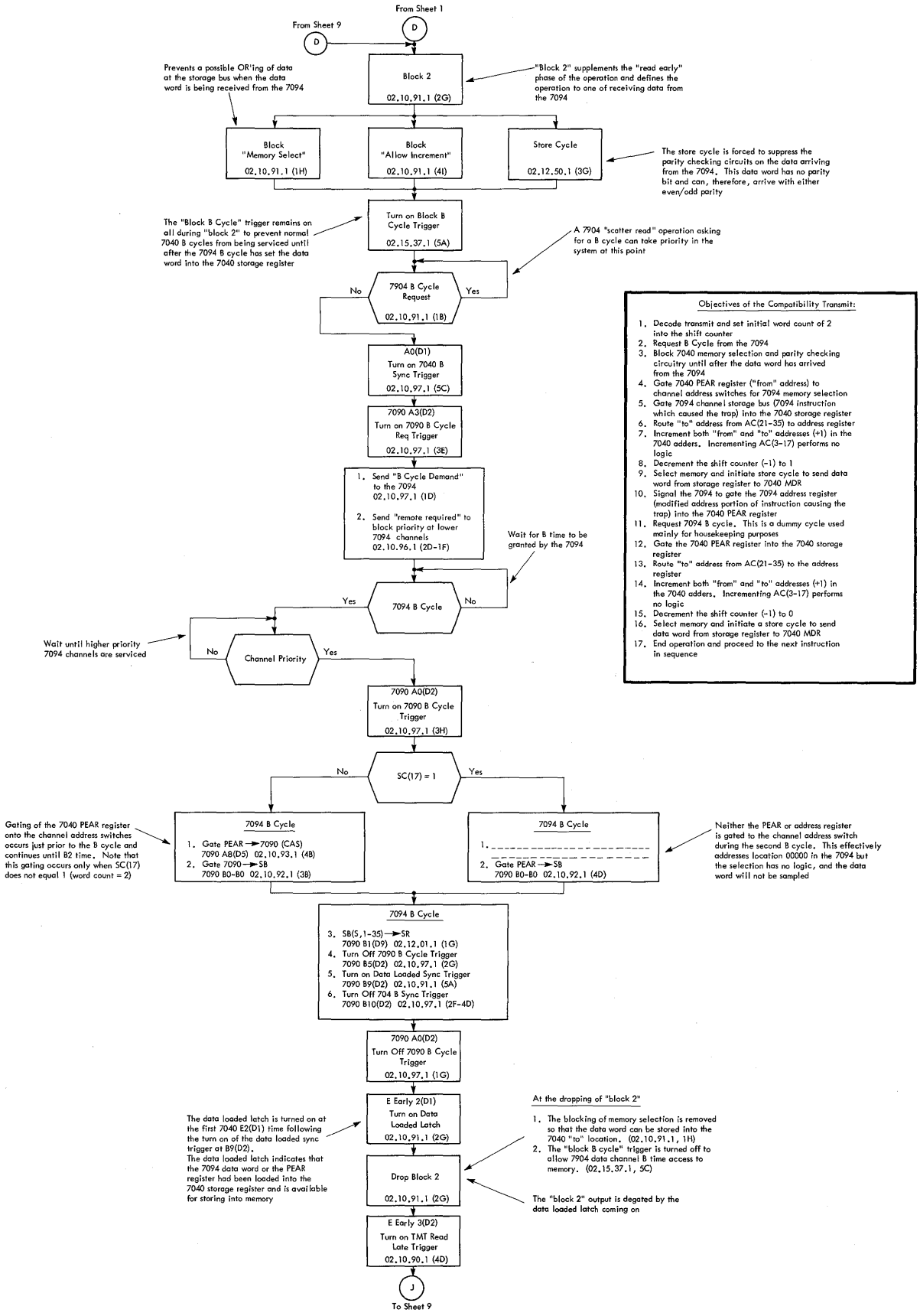


Figure 15. Transmit Flow Chart (Sheet 8 of 9)

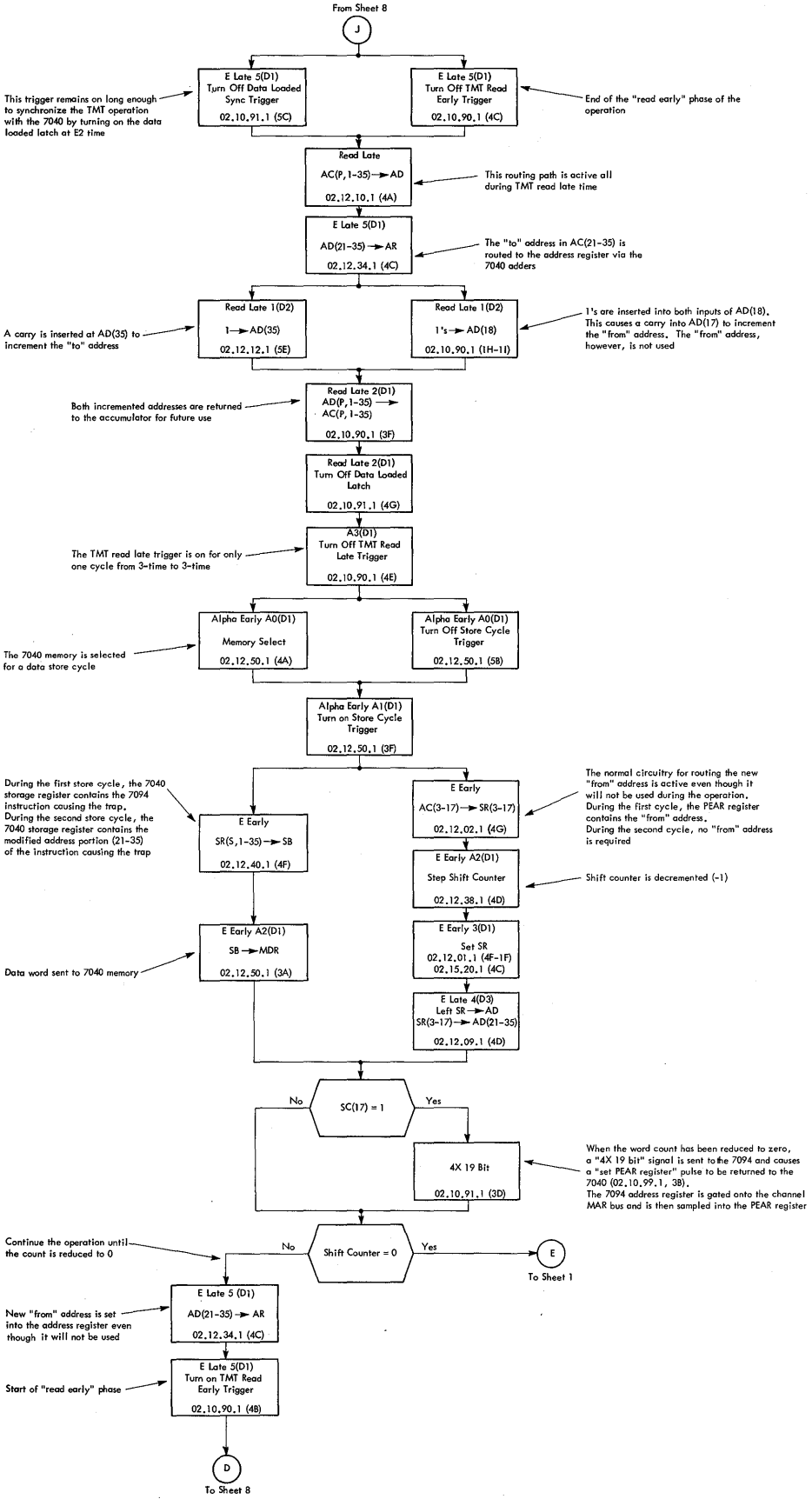


Figure 15. Transmit Flow Chart (Sheet 9 of 9)

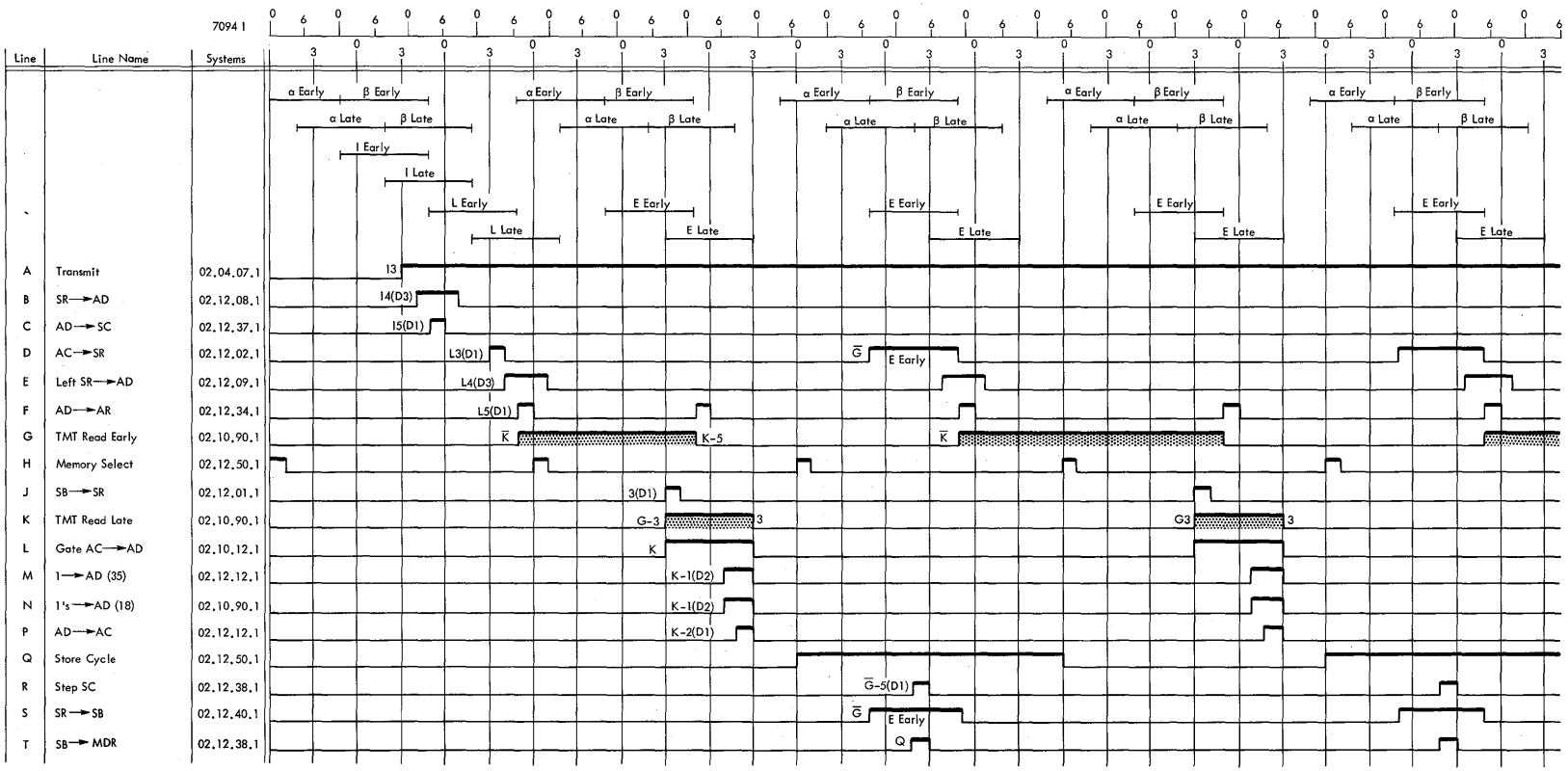


Figure 16. 7040 to 7040 Transmit Timing Chart

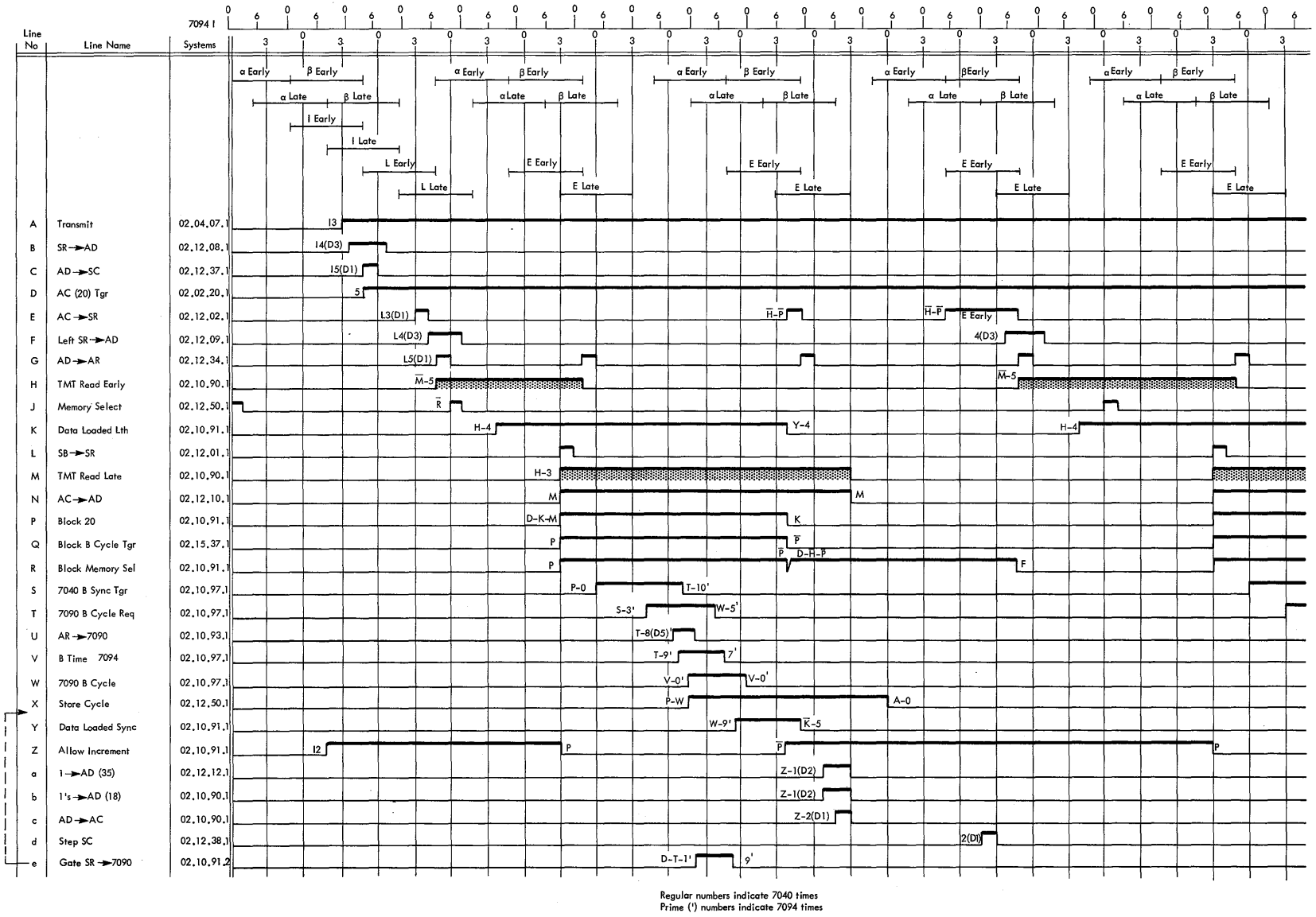


Figure 17. 7040 to 7094 Transmit Timing Chart

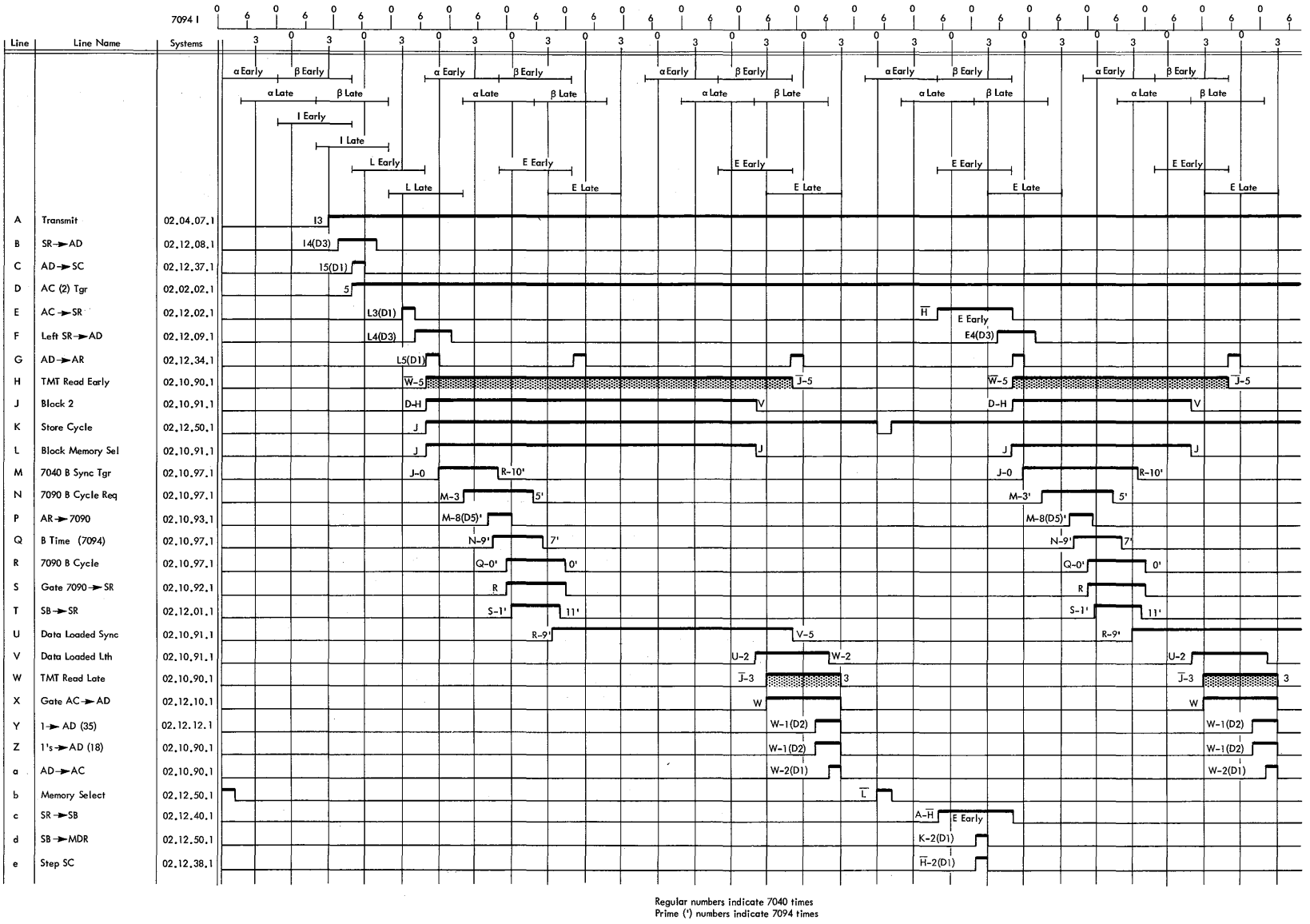


Figure 18. 7094 to 7040 Transmit Timing Chart

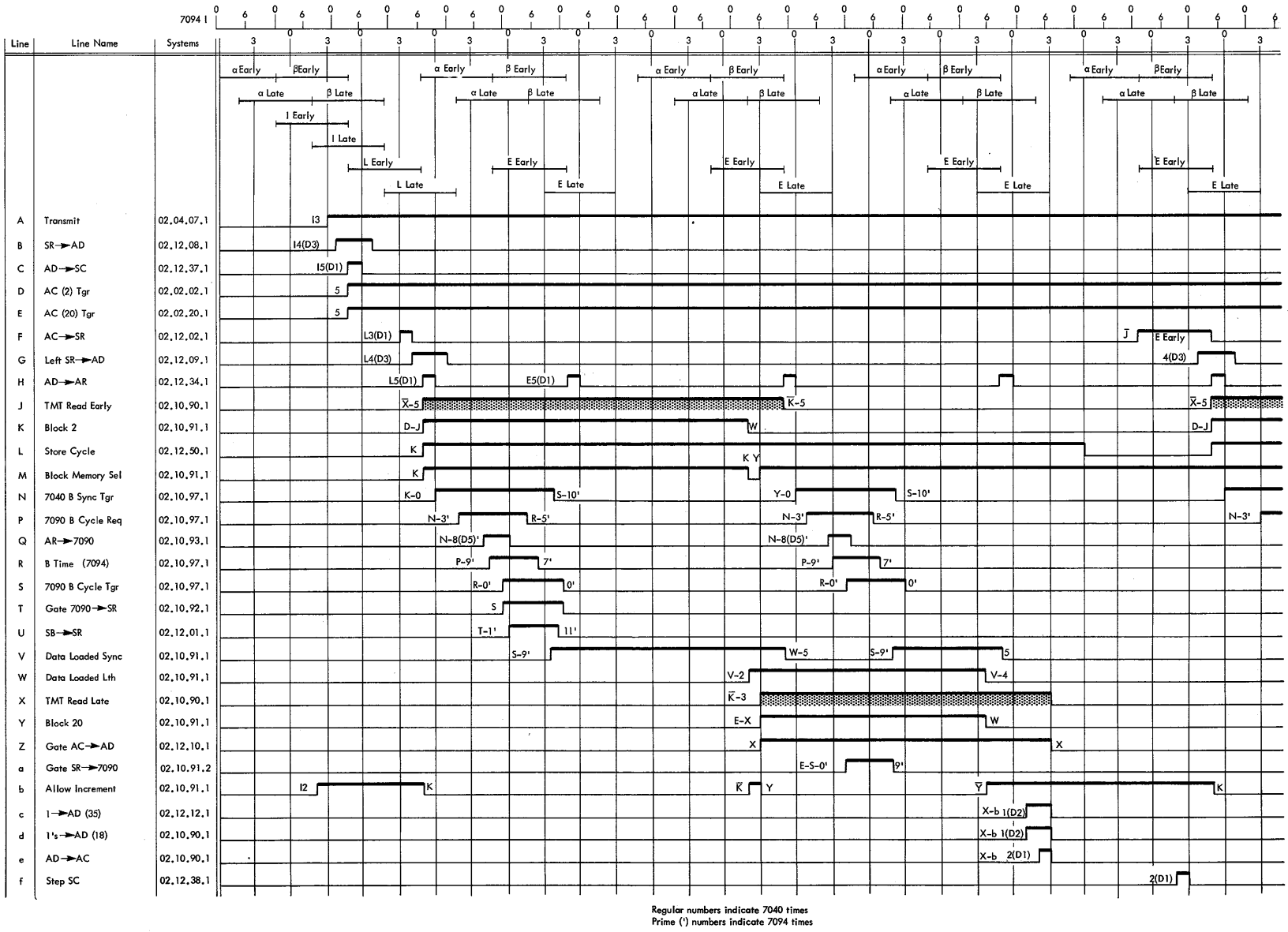


Figure 19. 7094 to 7094 Transmit Timing Chart

Line	Line Name	Systems
A	Transmit	02.04.07.1
B	SR → AD	02.12.08.1
C	AD → SC	02.12.37.1
D	AC (2) Tgr	02.02.02.1
E	AC (19) Tgr	02.10.99.1
F	AC → SR	02.12.02.1
G	Left SR → AD	02.12.09.1
H	AD → AR	02.12.34.1
J	TMT Read Early	02.10.90.1
K	Block 2	02.10.91.1
L	Store Cycle	02.12.50.1
M	Block Memory Sel	02.10.91.1
N	7040 B Sync Tgr	02.10.97.1
P	7090 B Cycle Req	02.10.97.1
Q	PEAR → CAS	02.10.93.1
R	B Time (7094)	02.10.97.1
S	7090 B Cycle	02.10.97.1
T	90 to 40 Tgr	02.10.92.1
U	Gate 7090 → SR	02.10.92.1
V	SB → SR	02.12.01.1
W	Data Loaded Sync	02.10.91.1
X	Data Loaded Lth	02.10.91.1
Y	TMT Read Late	02.10.90.1
Z	Gate AC → AD	02.12.10.1
a	1 → AD (35)	02.12.12.1
b	1's → AD (18)	02.10.90.1
c	AD → AC	02.10.90.1
d	Memory Select	02.12.50.1
e	SR → SB	02.12.40.1
f	SB → MDR	02.12.50.1
g	Step SC	02.12.38.1
h	4X 19 Bit	02.10.99.1
* j	AR → MAR	03.08.15.1
k	Set PEAR Reg	02.10.99.1
l	PEAR → SB	02.10.92.1
m	End-Op Tgr	02.15.39.1

Regular numbers indicate 7040 times
Prime (') numbers indicate 7094 times
* 7094 Circuitry

SC = 2 | SC = 1 | SC = 0

Figure 20. Compatibility Transmit Timing Chart (Count of 2)

- **Data can be read from the 7904 data channel into either the 7040 or 7094.**
- **This data can be selectively "scattered" into either or both computers under control of "chained" IORD commands.**
- **Programs can be executing concurrently on both computers while the scatter read operation is being performed.**

The scatter read feature allows data to be read directly into the 7094 memory from devices (tape or disk) on the 7904 data channel. This feature eliminates the need of first having to read into the 7040 and then transmit that data from the 7040 to the 7094. Direct reading such as this allows 7094 programs to be sent quickly and efficiently to the 7094. Information (data), on the other hand usually requires modification or manipulation and is therefore sent to the 7094 via the 7040.

The IORD scatter read feature can only send data to the 7094 while in multiprocess mode. If the system is not in multiprocess mode, the data will be unconditionally sent to the 7040. This scatter read feature may be used as a normal operation to read data into the 7040 at any time and is not dependent on multiprocess mode.

The two bit positions in the IORD command controlling the new operational features are:

Bit 18 — (chain bit) causes chaining of IORD commands.

Bit 20 — causes data to be sent to the 7094.

Bit 20 in the IORD command performs the same function as AC(20) in the transmit operation discussed previously in this manual. When bit 20=0, each data word is sent to the 7040. When bit 20 equals 1, each data word is sent to the 7094.

Bit 18 performs the chaining function. When bit 18 equals 0, the IORD operates in the conventional manner, i.e., the channel disconnects when the end of record is reached or when WC equals 0. When bit 18 equals 1, the count going to zero causes the channel to accept the next data word as a new IORD command. This new command is loaded into the appropriate channel registers and counters just as if initiated by an RCH instruction from the CPU. This new command also respecifies bits 18 and 20 for data transmission and chaining.

Figure 21 shows an example of how a single tape record could be scattered into both the 7040 and 7094.

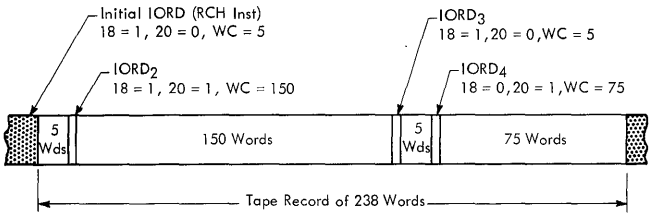


Figure 21. Scatter Read Tape Record

An initial IORD command originating from a CPU reset and load channel (RCH) instruction causes five words to be read into the 7040. The sixth tape word becomes a new IORD command to read 150 words (possibly a logical record) into the 7094. The 157th tape word becomes a new IORD command to read five more words into the 7040. The 163rd word becomes the last IORD command to read the remaining 75 words into the 7094.

The various IORD commands interspersed throughout the tape record do not appear there as if by magic. They have been placed there by programming, and specifically define the logical areas of that particular record.

Note that chaining of commands is confined to one physical record. In the case of Figure 21, chaining could not continue into a second record. Reading of the next record requires another RCH instruction from the 7040 program.

Reset and Load Channel RCH ±054x (I, E)

- **Positions 21-35 of RCH instruction specify address of initial IORD command.**
- **RCH causes loading of IORD command into channel.**
- **An RCH not preceded by an RDS or WRS causes an I/O check.**

The purpose of the RCH instruction is to locate the initial IORD command and set its contents into the channel registers and counters. Figure 22 shows the IORD format.

The operation code of the RCH instruction specifies the particular channel. As a result of decoding the RCH instruction, "POD 54" and "select channel B-E" signals are

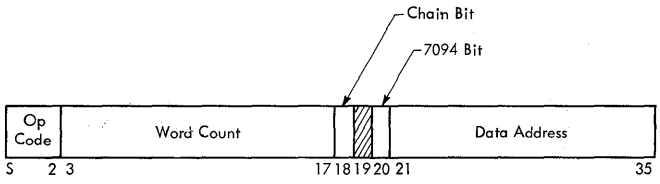


Figure 22. IORD Command Word Format

sent to the channel (Figures 23 and 24). The "POD 54" signal is common to all channels; the "select channel" signal will isolate the operation to the specific channel concerned with the operation.

Address modification can be performed on positions 21-35 of the RCH instruction. Therefore, during I4(D3), SR(21-35) and the specified index register are combined in the main adders. The effective address is set into the address register and memory is selected to read out the IORD command during the following E cycle.

At the end of I time, a test is made in the channel to see if the channel had been previously selected for a read operation. If the channel were not previously read selected, the I/O check trigger is turned on in the CPU. This I/O check only indicates a possible programming error and does not affect the RCH operation. The status of the I/O check can be program tested by the IOT instruction.

Also, at the end of the I cycle, both the channel word counter (CWC) and channel address counter (CAC) are reset in preparation for receiving the channel command during the following E cycle.

Starting at the initial decoding (I3 time) and continuing until the end of the RCH instruction, the data register is held reset and the channel word count is forced to indicate a zero condition (Figure 23).

The data register is held reset during the RCH instruction because (during the E cycle) the data register is used as a routing path from the storage bus for setting the word count (3-17) into the channel word counter. The reset line, therefore, allows the count field to be routed through the upper (+TOO) block without actually setting the data register with unwanted information.

The word count zero condition is forced during the RCH instruction to prevent the immediate generation of a B cycle. Note that this zero condition is not dependent on the actual word counter contents.

Early in the E cycle (E1 time) all positions of the word counter, CWC (3-17), are turned on. This is necessary because the minus (−) output of the +TOO is used from the data register to route the word count. This negative output then resets all word counter positions corresponding to zeroes on SB(3-17).

At E3 time, the IORD command is read from memory, placed onto the storage bus and gated into the word counter and address counter. Note, however, that the address counter value is not routed through the data register.

SB(18 and 20) are routed through DR (18 and 20) and are tested for bits to set the "command chain" and "bit 20" latches respectively (Figure 23).

IORD-Channel Operation

- **Controls reading data on the 7904 data channel.**
- **Data chaining can scatter incoming records to either the 7040 or 7094.**
- **Data transmission to the 7094 requires being in multiprocess mode.**
- **Scatter reading (chaining) may be used as a normal operation into the 7040 independent of multiprocessing.**

The 7904 data channel is limited to one I/O command: IORD. The conventional operation limits this command to one physical record and to consecutive locations in the 7040 memory.

The scatter read feature, however, introduces flexibility into the IORD command so that incoming data can be scattered into various locations in either the 7040 or the 7094. This scattering is under control of IORD commands sandwiched between logical records within the one physical record. This was discussed with an example in the previous section together with Figure 21.

Figure 25 is an overall flow chart showing both the normal IORD operation as well as that of scatter reading data words into either computer. Figure 26 is a timing chart of data being read into the 7094 and chaining to a new IORD command. Figure 27 is a timing chart of a scatter read operation occurring simultaneously with a 7094 to 7094 transmit operation in the 7040.

For ease of explanation and understanding, four operations or phases of the operation will be considered: a conventional IORD operation; data transmissions to the 7094; command chaining; and error and disconnect conditions.

Conventional IORD

- **Assemble incoming characters into complete data words (6 characters per word).**
- **Store data words into 7040 memory under 7040 B time control.**
- **Update channel address counter (CAC) and channel word counter (CWC) for data transferred.**
- **Stop transmission when WC = 0 or at end of record, whichever occurs first.**

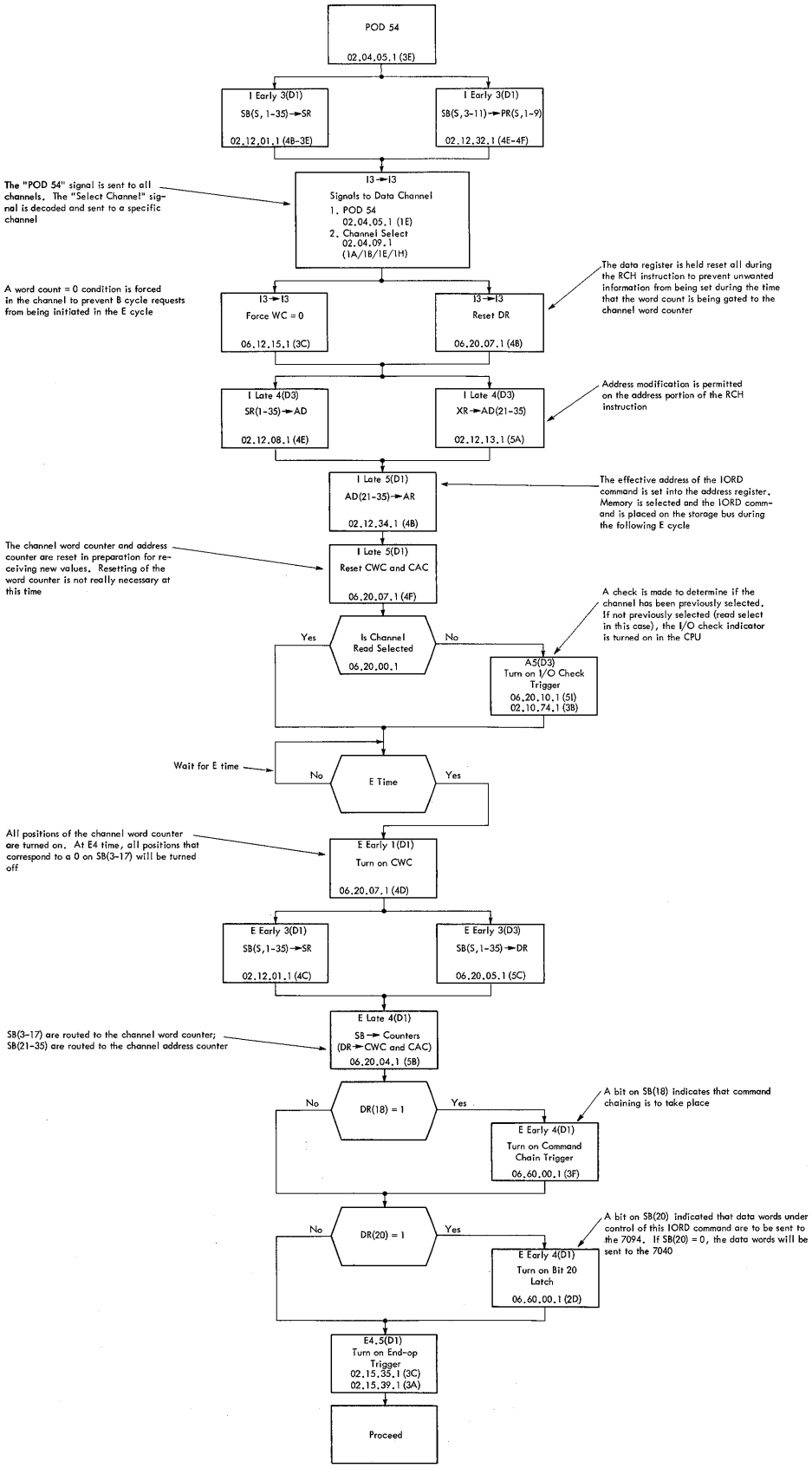


Figure 23. 7040 Reset and Load Channel Flow Chart

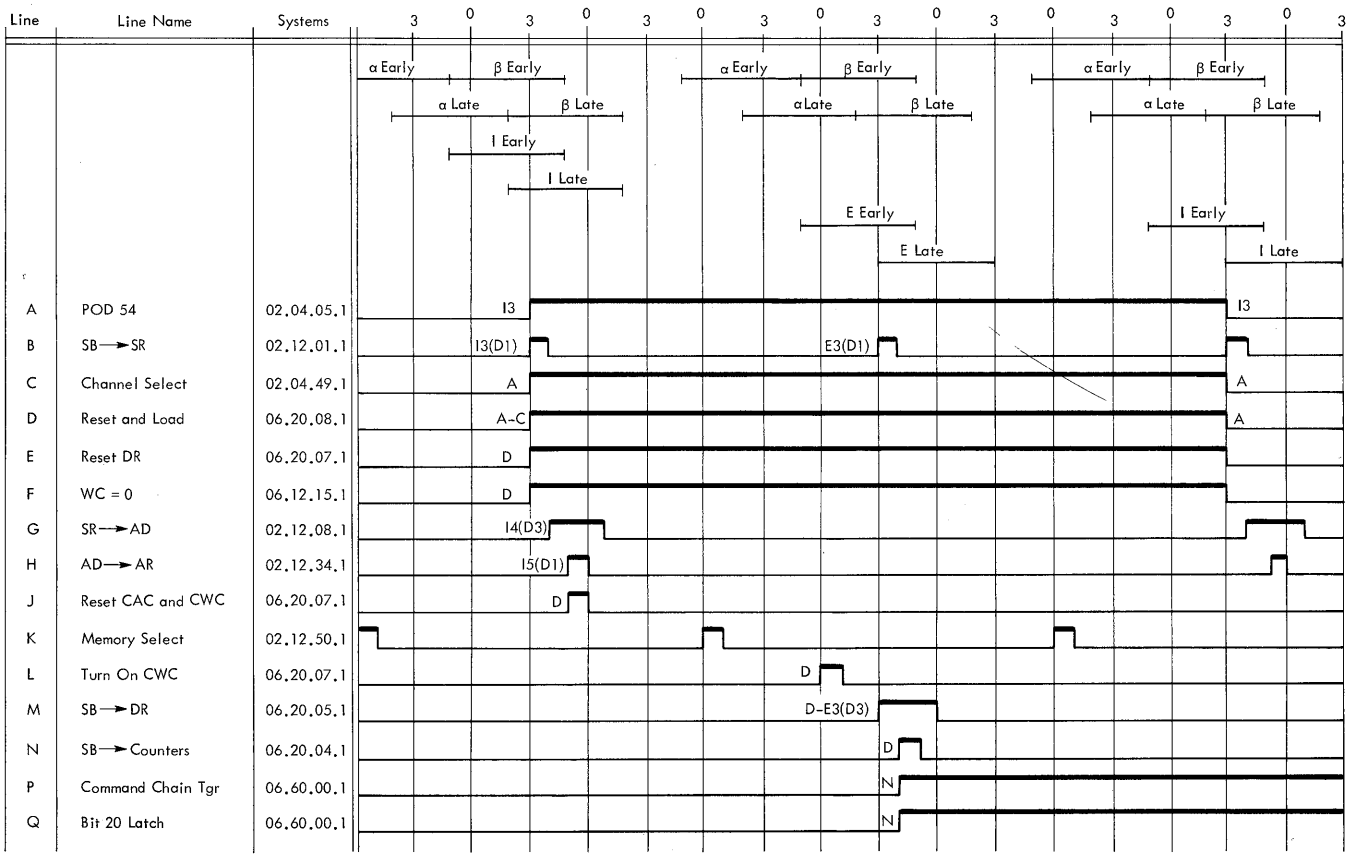


Figure 24. 7040 Reset and Load Channel Timing Chart

A description of the normal IORD operation will serve to point out in the following sections where the new feature departs from conventional operation.

Starting at the top of Figure 25, sheet 1, it is assumed that a previous read select and RCH instruction has started the I/O device moving and has loaded the channel counters. Assume also that the word count is less than the record length and that the number of characters in the record is a multiple of 6.

Nothing happens in the channel until a strobe pulse from the I/O device indicates that a character is available on the input bus. This strobe pulse initiates three 800 nanosecond single shots in sequence as a timing arrangement used for routing the data character, and to perform necessary checks and housekeeping.

Reading the First Character of a Word

At the time of the strobe pulse, a valid character is already available on the input bus and is feeding through the translator circuits. Whether or not translation actually takes place, however, depends on whether the 7904 is in binary or BCD mode of operation.

During this initial pass through the flow chart at SS1 time (Figure 25, sheet 1), the character counter will be set at 1. This counter was reset to 1 during the preceding RDS instruction to the channel. As a result of this count, the first character is gated from the translators and is set into assembly register (ASM) positions 0-5. The 6th character trigger is reset as the result of CC=1 to clear conditions from a previous full word (not applicable for the first character of a record).

During SS2 time, the only logic performed for a read operation is checking the character for correct parity. For now assume that parity is correct.

During SS3 time, a service response is generated (Figure 25, sheet 1) for standard interface operation so that the I/O device can release the present character from the in-bus and prepare for sending another. For operations with TAU, this service response is generated but is not used. The TAU operation does not require a response for the incoming character.

The character counter is stepped to 2 during SS3 time to prepare the channel circuits for the next incoming character.

A test is made (Figure 25, sheet 1) to see if the data register loaded trigger is on. At this point the trigger is off, and the operation returns to await the next strobe pulse.

During this pass through the flow chart, a character has been received, translated, stored in the assembly register, and checked for parity. A response has also been returned to the I/O interface, and the character counter stepped to 2.

End-of-file recognition circuits are also activated for the first character. This condition, however, is not covered in this manual.

Reading the 2nd-5th Characters of a Word

During the next four passes through SS1 time of the flow chart (Figure 25, sheet 1), higher values of the character counter cause the incoming characters to be gated into the assembly register as follows:

CHARACTER COUNTER	ASSEMBLY REGISTER POSITIONS
CC = 1	ASM (0 - 5)
CC = 2	ASM (6 - 11)
CC = 3	ASM (12 - 17)
CC = 4	ASM (18 - 23)
CC = 5	ASM (24 - 29)
CC = 6	ASM (30 - 35)

Each character is checked for parity during SS2 time, and during SS3 time the character counter is incremented to indicate the next character to arrive. At the end of the fifth pass through the flow chart the CC=6.

Reading the 6th Character of a Word

The 6th character is received, translated, and gated (under CC=6 control) into ASM (30-35). During this pass through SS1 time (Figure 25, sheet 1), a 6th character trigger is turned on to cause the full 36-bit data word to be transferred from the assembly register to the data register.

During SS2 time the 6th character is checked for correct parity.

During SS3 time the service response is generated as before but, now, additional functions occur because of the 6th character condition.

The character counter is reset to 1 for receiving the first character of a new word. Note that because the character counter is a binary make-up of three triggers, just stepping the counters would tend to indicate a CC=7 instead of CC=1 as is needed. The normal stepping pulse still occurs, however, but is overridden by the reset pulse.

As each character was received during SS1 time, a parity generator circuit was remembering the total odd/even count of bits being assembled in the assembly register. During SS3 time of this 6th character cycle, the status of the parity generator is set into DR(C) to establish an odd parity for the full data word. This data word was also gated to DR(0-35) during this sixth SS1 time.

Now that the data register is completely set, the data register loaded trigger is turned on (SS3 time) to initiate a B cycle for storing the information.

The assembly register is reset (SS3 time) in preparation for receiving the first character of a new word.

Storing The Data Word (7040 B Cycle)

A full data word is now in the data register (indicated by the data register loaded trigger) and a second data word is beginning to form in the assembly register. The first data word must therefore be stored before the next data word completely fills the assembly register or else an "overrun" condition will occur and data will be lost.

Whenever a read operation finds the data register loaded and a word count not equal to 0, the read operation immediately knows that a B cycle must be requested for storing data. These conditions are indicated at the bottom decision block (Figure 25, sheet 1) and the top decision block in Figure 25, sheet 2.

A B cycle demand is now initiated by the channel. Because there is no "bit 20" indication (this is a normal 7040 operation), the operation proceeds down the flow chart and seeks priority with other 7904 data channels. Once priority is obtained, the channel gates its address counter (memory data address) to MAR for the subsequent 7040 B cycle.

At the time of the 7904 B cycle request, the 7040 could be in multiprocess mode and transmitting (TMT) data to or from the 7094. If this is the case, the 7040 B cycle must be delayed to prevent a conflict in 7040 storage bus usage. The "block B cycle" trigger (Figure 25, sheet 2) being off indicates that the 7040 is not in the process of transmitting data to or from the 7094 and, therefore, allows the 7040 B cycle trigger to be turned on at A5(D2) time.

The 7040 B cycle is concerned mainly with memory controls and gating:

1. Select memory.
2. Set the MAR address.
3. Force a store cycle.

The CPU B cycle, in turn, causes a 7904 B cycle. The CPU has exerted control over memory; now the channel must supply the data. The logical functions performed during the 7904 channel B cycle shown in Figure 25, sheet 2 are as follows:

1. *Gate the DR to SB* – to make the data word available to memory.
2. *Reset the DR* – to allow a new word to be received from the assembly register when necessary.
3. *Turn off the B cycle demand trigger* – to drop the request for further 7040 B cycles.
4. *Step the CAC (+1) and CWC (−1)* – to indicate next memory data address and remaining data transmission.
5. *Turn off DR loaded trigger* – as a logical indication that the DR is available for new data.
6. *Turn off the 7904 B cycle trigger* – to end the cycle.

The operation now returns to sheet 1 of Figure 25. This B cycle has most likely been finished before arrival of the next character at the assembly register. If time were needed, however, to obtain channel priority and B cycle allotment for memory usage, it is possible that one or more characters have already been loaded into the assembly register. In this latter case, character assembly and data storing occur simultaneously.

Channel Disconnect (TAU Operation)

Characters flow into the assembly register under control of the character counter; complete data words continue to be stored under control of the word counter. During one of the B cycles, the word counter is stepped to zero indicating that the last data word has just been stored.

Under these conditions where the word count is less than the record length, the rest of the physical record continues to flow into the channel circuits as usual (Figure 25, sheet 1). The difference, however, is that B cycles are no longer requested for storing the data. Data ceases to flow, and disconnect procedures start as soon as the end-of-record is reached and TAU busy falls.

The fall of TAU busy, indicating the end of the tape operation, turns on the disconnect call and channel disconnect triggers which, in turn, perform the necessary reset functions as shown in Figure 25, sheet 1.

Note that parity checking is performed on all characters arriving after WC=0. The redundancy check trigger can, therefore, be turned on by data not actually stored in memory.

Channel Disconnect (SI Operation)

Channel disconnect on the standard interface (SI) has some differences from TAU disconnect. Consider the case where the word count is less than the record length.

During the B cycle that stored the last data word, the word count stepped to 0. The next character received (indicated by service request) will be the first one of a new data word. Proceeding through SS1 time (Figure 25, sheet 1) we see that a WC=0 condition causes the SI-TI disconnect trigger to be turned on.

This character being received is handled in the normal fashion even though it is not logically used or stored in memory. During SS3 time, the normal service response is returned to the I/O interface. As a result of service response the I/O device drops service request. It is at this point that the stop trigger is turned on and a stop signal is sent across the I/O interface.

This stop signal now prevents the transfer of any additional characters by the I/O device and causes the device to drop command response. The fall of command response causes SI busy to fall (Figure 25, sheet 1) and the channel initiates disconnect procedures (Figure 25, sheet 1).

Note that during SI operations, a channel disconnect does not necessarily wait for an EOR condition as explained previously for TAU.

Data Transmission to the 7094

- **System must be in multiprocess mode with bit 20 in the IORD command.**
- **Characters are assembled into the assembly register as usual.**
- **A 7094 B cycle is requested and the data word is sent to the 7094 via the channel storage bus.**
- **Normal 7040 and 7904 B cycle circuitry is blocked during data transmissions to the 7094.**
- **An "update" 7040 and 7904 B cycle is requested to perform counter updating and channel housekeeping.**

Assume that this is data transmission to the 7094 with no chaining. The system must be in multiprocess mode with a bit in position 20 of the IORD channel command.

Figure 25, sheet 1, shows that the assembly of characters into full data words is the same as for a normal 7040 IORD operation; there is no bit 20 control at this point.

When the data register has been filled with a data word from the assembly register, a B cycle demand is sent to the 7040 CPU (Figure 25, sheet 2). At this point the circuit is tested for both bit 20 and multiprocess mode. For the data word to be sent to the 7094 memory, both of these tests must be passed. The bit 20 signal which arrives from the channel is gated by B cycle demand and is, therefore, only active while the channel is in the process of obtaining B time with the 7040.

It is possible for the 7040 to be simultaneously executing a TMT instruction with the 7094. If this is the

case, 7904 data transmission must be delayed during the period of time when the TMT is actively transmitting a data word to or from the 7094. This is necessary to prevent a conflict in storage bus usage because both operations use the same basic data paths.

The 7904 B sync trigger is turned on to gain control of the 7040 circuits. As a result of this trigger and the "block cycle" latch coming on (Figure 25, sheet 2):

1. *The 7040 B cycle trigger is held reset* – to prevent a 7040 and 7904 B cycle from occurring at this time.
2. *Other 7904 channel B cycle demands are blocked* – to prevent other 7904 channels from obtaining B time while this channel is transferring a data word to the 7094.
3. *The 7040 B sync trigger is blocked from setting* – to prevent a simultaneous 7040 TMT operation from also sending data to the 7094 at this time.
4. *7040 memory selection is blocked* – to prevent data from being OR'ed on the storage bus at this time.
5. *A store cycle is forced* – to prevent the parity circuits from causing a false error when the parity bit is dropped on its way to the 7094.

7094 B Cycle

At this point the 7904 channel has gained access to the 7040 system. The 7040 must now request a B cycle from the 7094 and transfer the data word to the proper location in the 7302 memory.

The 7094 memory address is supplied by the 7904 channel address counter: gate CAC to MAR (Figure 25, sheet 2). This address is available to the 7040 from the time that the 7904 obtains priority until after the data word is stored in the 7094. From the 7040, this address is further gated to the 7094 system MAR as soon as 7094 channel priority is obtained.

The 7040 appears to the 7094 as a data channel. Therefore, the 7040 must request a 7094 B cycle and obtain 7094 channel priority (Figure 25, sheet 2). The 7090 B cycle trigger is turned on with a 7094 A0(D1) pulse to synchronize the two systems. During this 7094 B cycle:

1. *The CAC address is gated to MAR* – to select the proper 7094 address. The CAC had been gated to the 7040; now it is further gated to the 7094.
2. *Channel store control* – forces a store cycle in the 7302 memory.
3. *The 7904 DR is gated to the 7040 SB* – and,
4. *The 7040 SB is gated to the 7094 channel storage bus* – to completely route the data word into the 7302 memory.

Up to this point there has not been a 7040 B cycle; the 7040 B cycle trigger has been purposely blocked. In fact, a 7040 cycle would not be required at all if it were not for channel updating and housekeeping functions.

7040 and 7904 Update B Cycle

Towards the end of the 7094 B cycle, the blocking effect on the 7040 B cycle trigger is removed and the trigger is allowed to turn on at A5(D1) time. Because this is strictly a housekeeping B cycle, memory selection is blocked. The 7040 B cycle, in turn, causes the following actions in the 7904 channel B cycle (Figure 25, sheet 2):

1. *Turns off the B cycle demand trigger* – to prevent further B cycle requesting and also drop bit 20 to the 7040.
2. *Resets the data register* – to free it for the next data word from the assembly register.
3. *Steps the CAC+1* and *CWC−1* – to update these counter contents.
4. *Turns off the DR loaded trigger* – as a logical indication to the 7904 channel that the data word has been stored.

The data word has been stored, the channel has been updated, and the operation now returns to sheet 1, Figure 25, to assemble the next data word. When the word count goes to zero, the channel disconnects as in a normal IORD operation.

Command Chaining

- **Not dependent on multiprocess mode or bit 20.**
- **Provides a means of "scatter" reading data.**
- **At WC = 0, next data word is used as new IORD command word.**
- **Each new IORD command can direct the data into either the 7040 or 7094.**
- **Chaining is limited to one physical record.**

Command chaining provides a means of "scatter" reading data into either/both the 7040 or 7094 systems as explained earlier in Figure 21. The operation is not limited to a multiprocessing operation, but can be used on a stand-alone 7040.

During command chaining, the assembling of data continues according to the normal IORD operation. When the word count is reduced to zero, however, (Figure 25, sheet 1) the channel does not start disconnecting. Instead, this next data word which was transferred from the ASM to the DR during SS1 time is used as a new IORD command.

Because this last data word is not really a data word, the normal B cycle request circuitry is blocked, and the channel registers are reloaded from the contents of the data register. This operation is almost identical to the E cycle of an RCH instruction.

As shown at the top of Figure 25, sheet 2, when the DR is loaded, WC=0, and the command chain trigger is on, the normal B cycle path is by-passed. Instead, the chain sync trigger is turned on and the channel proceeds with chain updating. The functions performed during this phase of the operation are:

1. *The CAC and CWC are reset* – to clear the counters for new values. Reset of the CWC is not necessary but occurs with the CAC reset.
2. *The CWC is set to all 1's* – because of the complement method of setting the counter from the data register.
3. *The bit 20 latch and command chain trigger are reset* – to accept new setting from the new IORD.
4. *Bits in DR (18 and 20) are tested* – to set the command chain trigger and "bit 20" latch for the next operation.
5. *DR (3-17) and DR (21-35) are gated* – to the CWC and CAC for data control of the next logical record.
6. *The DR is reset and the DR loaded trigger is turned off* – to clear the circuitry for the next data word.

With the new IORD command in the channel, the operation returns to Figure 25, sheet 1 to continue assembling data for the next storage cycle.

Note that if an RCH initially loads a WC=0, the first data word received from the I/O device becomes the active IORD command.

Error and Disconnect Conditions

Data Overrun – I/O Check with Chaining

If the 7904 cannot be serviced fast enough by the system, it is possible that a new complete data word will be assembled in the assembly register before the previous word has been stored from the data register. In this case, data has been lost and the CAC and CWC are no longer in proper relationship to the incoming data.

Detection of the overrun condition is indicated by the "transmission loss" indicator in the channel and the I/O check indicator in the CPU (Figure 25, sheet 1). The CPU indicator can be tested by means of the IOT instruction.

The overrun condition, however, is completely ignored by the channel circuits. Therefore, with the CWC out of synchronization with incoming data, it is possible for the channel to accept a word of actual data and use it as a new IORD command. In this case it is impossible to predict the channels course of action without knowing the bit make-up of the data word.

Redundancy Check – Enabled with No Chaining

Each character received from the I/O device and each word sent to the 7040 is checked for correct parity. An error detected in either case causes a redundancy check trigger to be turned on (Figure 25, sheet 1). In this case, data transmission is blocked to the 7040 or 7094 by forcing word count=0.

For a tape operation, characters continue to be received from TAU until the tape reaches the end-of-record gap. At this time, TAU busy falls, the channel disconnects, and the redundancy trap begins.

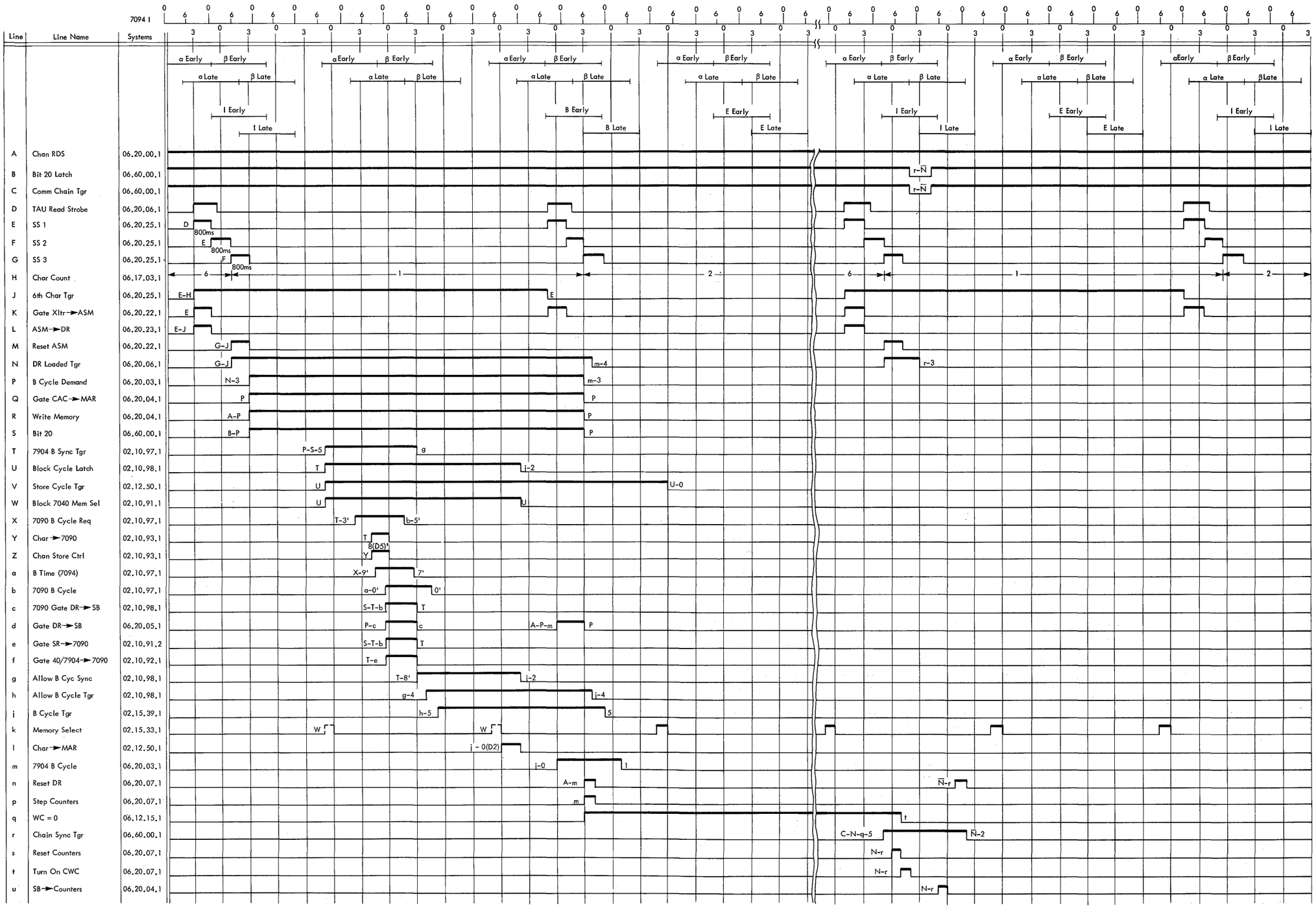
For a standard interface operation, a redundancy check causes a "stop" signal to be sent to the I/O device. The device stops the flow of characters across the interface and drops command response at the appropriate time. The fall of command response initiates a channel disconnect which allows the redundancy trap to begin.

Redundancy Check – Enabled with Chaining

A redundancy check detected during a chaining operation forces a word count=0, as explained above. With a WC=0, the channel starts a chain update operation as soon as the data register is loaded. This chaining operation as well as any subsequent chaining operation is invalid and will not affect the channel operation because the redundancy check is maintaining a word count=0 condition.

Figure 25. 7904 Data Channel Flow Chart (Sheet 1 of 2)

Figure 25. 7904 Data Channel Flow Chart (Sheet 2 of 2)



Regular numbers indicate 7040 timings
Prine (') numbers indicate 7094 timings

Figure 26. 7904 to 7094 Scatter Read with Chaining Timing Chart

Line	Line Name	Systems
A	Transmit	02.04.07.1
B	SR→AD	02.12.18.1
C	AD→SC	02.12.37.1
D	AC(2) Tgr	02.02.02.1
E	AC(20) Tgr	02.02.20.1
F	AC→SR	02.12.02.1
G	Left SR→AD	02.12.09.1
H	AD→AR	02.12.34.1
J	TMT Read Early	02.10.90.1
K	Block 2	02.10.91.1
L	Store Cycle Tgr	02.12.50.1
M	Block Mem Sel	02.10.91.1
N	Select Memory	02.15.33.1
P	7040 B Sync Tgr	02.10.97.1
Q	7090 B Cyc Req	02.10.97.1
R	AR→7090	02.10.93.1
S	B Time (7094)	02.10.97.1
T	7090 B Cycle Tgr	02.10.97.1
U	Gate 7090→SR	02.10.92.1
V	SB→SR	02.12.01.1
W	Data Loaded Sync	02.10.91.1
X	Data Loaded Lth	02.10.91.1
Y	TMT Read Late	02.10.90.1
Z	Block 20	02.10.91.1
a	Gate AC→AD	02.12.10.1
b	Gate SR→7090	02.10.91.2
c	Allow Increment	02.10.91.2
d	1→AD(35)	02.12.12.1
e	1's→AD(18)	02.10.90.1
f	AD→AC	02.10.90.1
g	Step SC	02.12.38.1
h	Channel RDS	06.20.00.1
j	Bit 20 Latch	06.60.00.1
k	TAU Read Strobe	06.20.06.1
l	SS 1	06.20.25.1
m	SS 2	06.20.25.1
n	SS 3	06.20.25.1
p	Char Count	06.17.03.1
q	6th Char Tgr	06.20.25.1
r	Gate XItr→Asm	06.20.22.1
s	ASM→DR	06.20.23.1
t	Reset ASM	06.20.22.1
u	DR Loaded Tgr	06.20.06.1
v	B Cycle Demand	06.20.03.1
w	Gate CAC→MAR	06.20.04.1
x	Write Memory	06.20.04.1
y	Bit 20	06.60.00.1
z	7904 B Sync Tgr	02.10.97.1'
aa	Block Cycle Lth	02.10.98.1
bb	Char→7090	02.10.93.1
cc	Chan Store Ctrl	02.10.93.1
dd	7090 Gate DR→SB	02.10.98.1
ee	Gate DR→SB	06.20.05.1
ff	Gate SR→7090	02.10.91.2
gg	Gate 40/7904→7090	02.10.92.1
hh	Allow B Cycle Sync	02.10.98.1
jj	Allow B Cycle	02.15.39.1
kk	7040 B Cycle Tgr	02.15.39.1
ll	Char→MAR	02.12.50.1
mm	7904 B Cycle	06.20.03.1
nn	Reset DR	06.20.07.1
pp	Step Cntrs	06.20.07.1

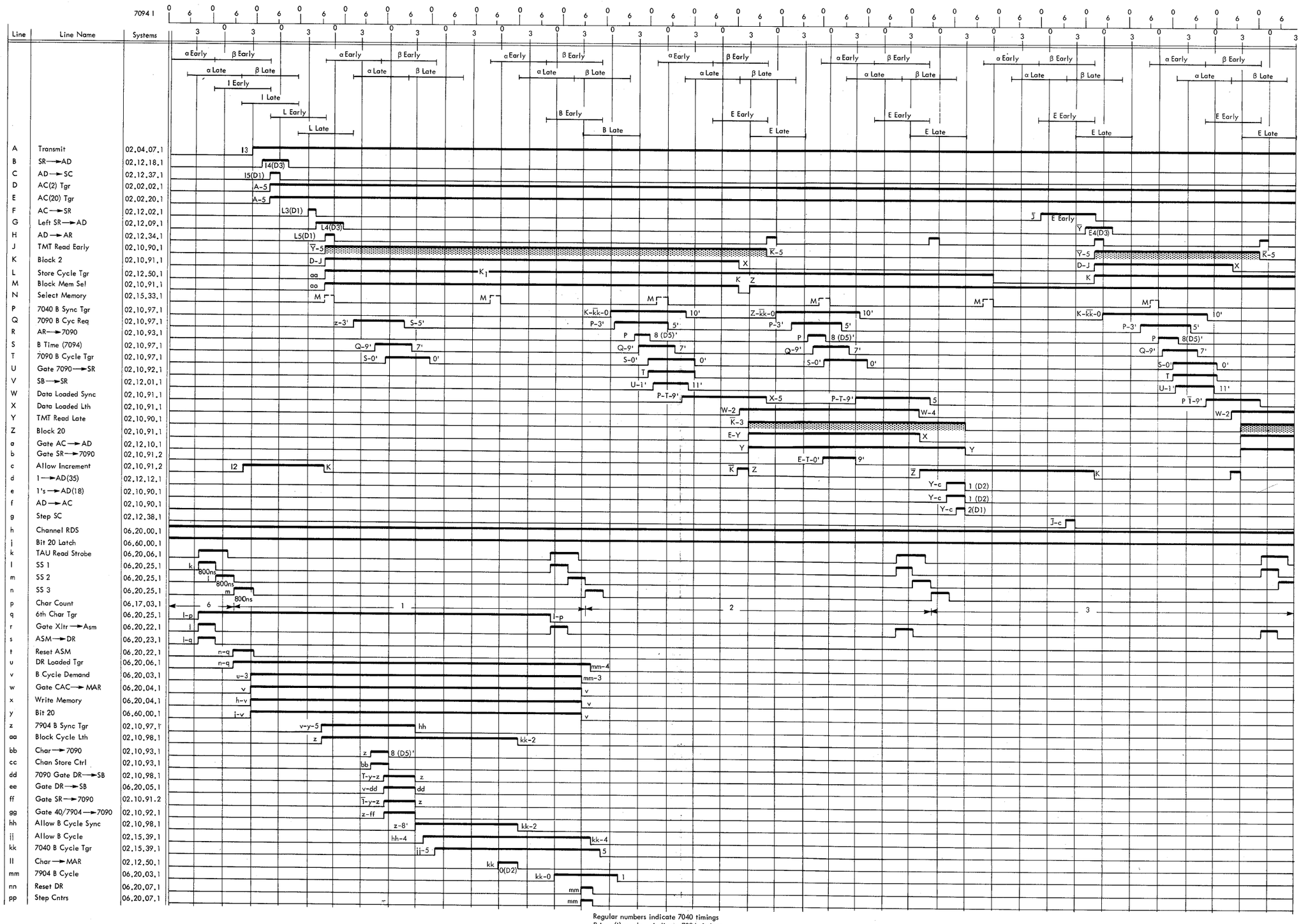


Figure 27. Simultaneous Transmit and Scatter Read Timing Chart

BTT—Multiprocess Channel

- **BTT(M) is an "I/O call" from the 7094 to the 7040.**
- **The 7040 will not trap unless enabled for this type of trap.**
- **This BTT(M) (I/O Call) trap request can be stacked in the 7040 if not enabled.**
- **The BTT(M) never causes a 7094 program skip.**
- **Channel selection requires physical wiring changes for the multiprocess channel.**

Execution of the BTT instruction is designed to act as an I/O call from the 7094 to the 7040 system. If the 7040 is enabled, the program traps and answers the demands of the 7094. At the time of the BTT trap, bit 15 is stored in location 00034_8 of the 7040 to identify the type of trap. When the 7040 trap occurs, further traps are inhibited until a new enable (ENB), or restore channel traps (RCT) instruction is executed by the 7040.

A 7094 BTT instruction issued to the multiprocess channel never causes a program skip; i.e., it always operates as though the BTT indicator were ON in the channel.

In the following case assume a pure multiprocess operation with the computer out of both HIP and exempt modes. Assume also that channel E is the multiprocess channel (Figures 28 and 29).

The BTT instruction (+0760 . . . c000) is set into the program register and storage register at I6 time (Figure 29). The address portion of the instruction is set into the address register with an I7 CP set pulse and is then routed to the shift counter via the index adders at I10 time. The shift counter now contains SR(28-35) which is used for class and unit address decoding.

Because of the BTT instruction format, the following decoding is required:

POD 76
SOD 00
Unit Address (SC 14-17) = 0
Class Address (SC 10-13) = 0
PR(S) - - Plus

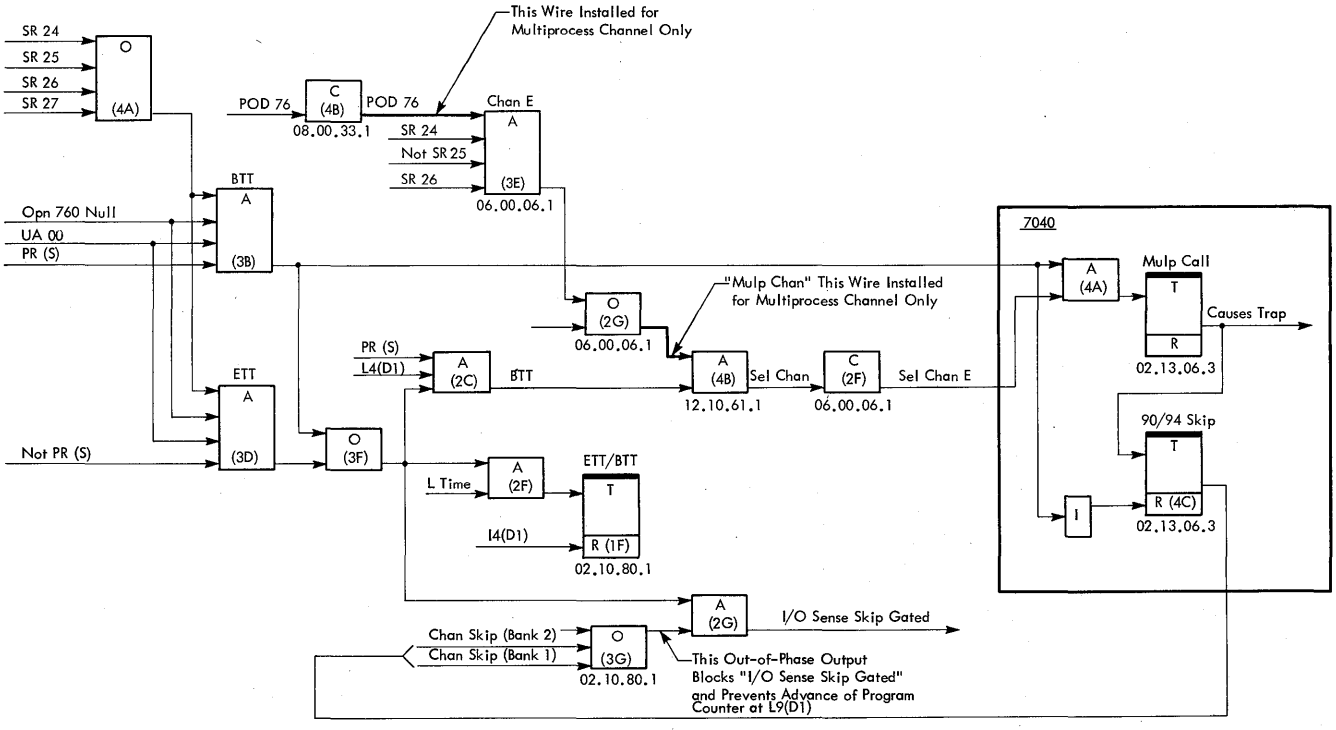


Figure 28. BTT Condensed Logic for Multiprocess Channel

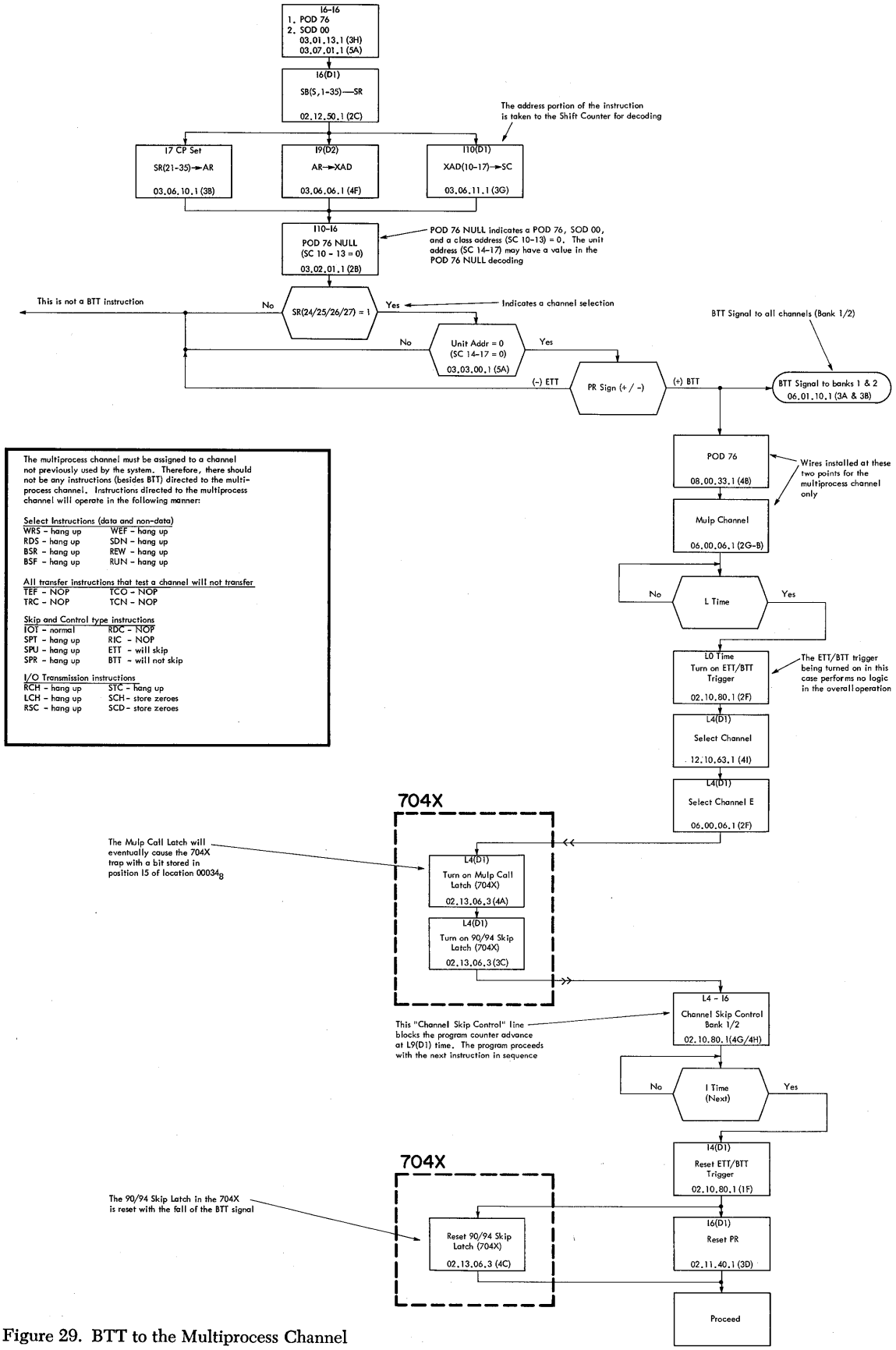


Figure 29. BTT to the Multiprocess Channel

The channel indication is located and decoded from SR(24-27). Note that with the above decoding (Figures 28, 29) a BTT signal is sent to all channels on banks 1 and 2. The only channel to respond actively will be the channel which also receives the channel select pulse.

Channel selection results from proper machine wiring. Note that selection of the multiprocess channel (channel E) results from the POD 76 wire from the convert block (Systems 08.00.33.1, 4B). The line is wired for the multiprocess channel only and produces "mulp chan" from the OR circuit (Systems 06.00.06.1,2G). The out-of-phase output of this OR circuit (pin A) is *removed* for the multiprocess channel. The in-phase output (pin B), however, *is* wired for the multiprocess channel. This latter output finally AND's with BTT decoding to form an L4(D1) "sel chan E" pulse to the 7040 (Figures 28, 29).

The combination of BTT and "sel chan E" in the 7040 turns on a mulp call latch which remembers the I/O call request from the 7094. The latch output eventually causes a 7040 trap with a bit stored in position 15 of location 00034_8. See section on Trapping.

Design of the BTT(M) operation is such that a program skip will not occur. Suppression of the skip at L9(D1) time requires a "chan skip ctrl" signal from the channel. This signal is produced by a 90/94 skip latch which is also turned on in the 7040. This latch stays on for the remainder of the BTT(M) operation and is turned off by the fall of the BTT signal from the 7094.

End of operation results from the normal BTT end-op circuitry (Systems 08.00.01.1,3C), and the 7094 proceeds with the next sequential instruction.

The ETT/BTT trigger (Systems 02.10.80.1) was turned on at L0 time but performs no logic in this operation with the multiprocess channel.

BTT (Multiprocess, HIP, and Exempt Modes)

- **BTT instruction operates in normal manner when the 7094 is operating stand-alone (no multiprocess references).**
- **In multiprocess mode, BTT operates in normal manner to attached data channels, and as a BTT(M) to multiprocess channel.**
- **In HIP mode, BTT operates as a BTT(M) to multiprocess channel, and causes HIP halt to other channel references.**
- **In HIP and exempt modes, the BTT:**
  Acts as a BTT(M) to the multiprocess channel.
  Acts as a normal BTT to 7094 exempt channels.
  Causes a HIP halt to other channel references.

Operation of the BTT instruction in the various system modes is summarized in Figure 30.

Figure 31 is an expanded flow chart showing the 7094 BTT instruction under all three modes of operation.

The multiprocess channel is installed as a channel which has not yet been used prior to this time as a part of the overall system. Because of this, any BTT instruction issued to the multiprocess channel is meant to be an I/O call to the 7040. Under these conditions, the 7040 traps and the 7094 continues with the next sequential instruction.

When only in multiprocess mode, the BTT instruction operated normally for all channels other than the multiprocess channel. If the BTT indicator is off, the program skips one instruction; if the BTT indicator is on, the program continues with the next sequential instruction. Actually the 7094 has no active signal that indicates being in multiprocess mode. The operation is therefore the same as under stand-alone conditions.

When the system is in both multiprocess and HIP modes, a HIP-halt occurs for a BTT instruction directed at any channel other than the multiprocess channel. A BTT to the multiprocess channel operates as an I/O call to the 7040.

When the system is in multiprocess, HIP, and exempt modes, a BTT instruction will:

1. Operate as a normal BTT when issued to an exempt channel.
2. Cause a HIP-halt when directed to a channel other than an exempt or multiprocess channel.
3. Operate as an I/O call when issued to the multiprocess channel.

Channel	Stand-alone	Multiprocess Mode		
			HIP Mode	
				Exempt Mode
Multiprocess	I/O Call	I/O Call	I/O Call	I/O Call
Exempt	Normal	Normal	HIP-halt	Normal
Other	Normal	Normal	HIP-halt	HIP-halt

Figure 30. BTT Instruction Mode Summary

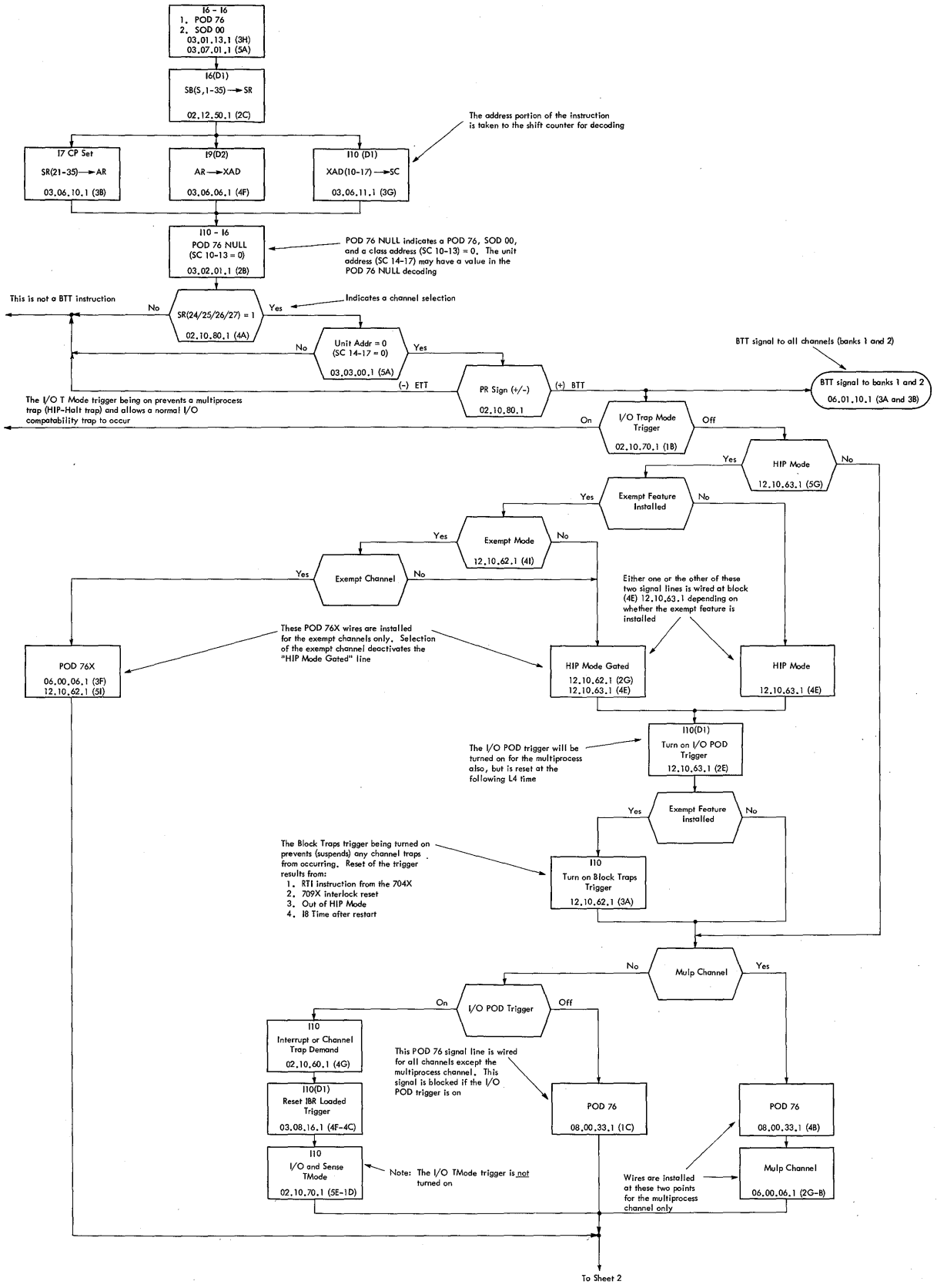


Figure 31. Overall BTT Flow Chart (Sheet 1 of 3)

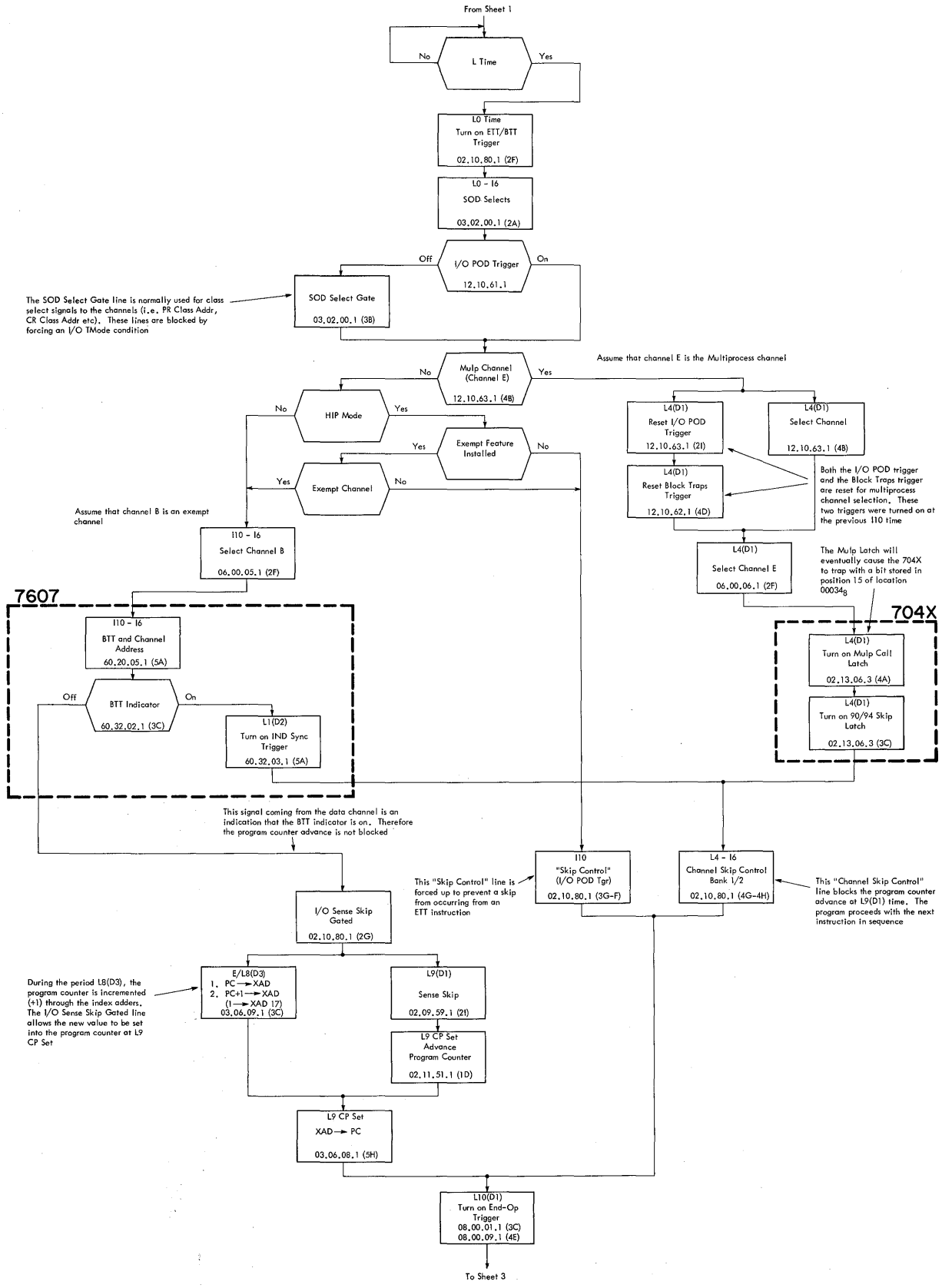


Figure 31. Overall BTT Flow Chart (Sheet 2 of 3)

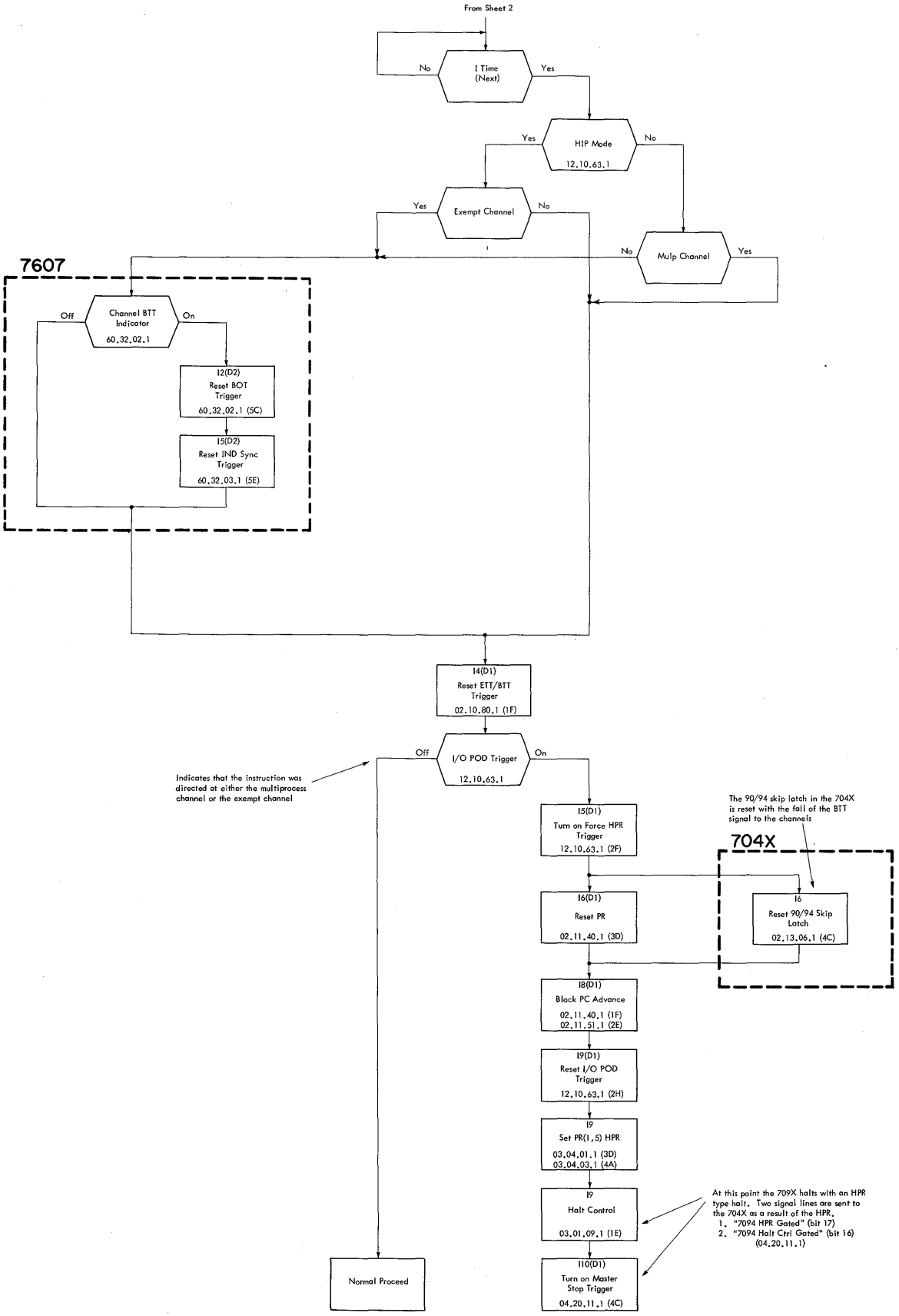


Figure 31. Overall BTT Flow Chart (Sheet 3 of 3)

Reset and Load Channel (POD 54)

- **All POD 54 instructions operate in a normal manner when not in HIP mode or when selecting an exempt channel.**
- **When a HIP halt condition exists, the POD 54 instruction is forced to end-op in L time. No "proceed to E" is received from the channel.**
- **If indirect addressing is specified, the IA cycle occurs before the HIP halt.**

The following description includes all of the POD 54 instructions: reset and load channel (RCH), and load channel (LCH) instructions for the 7607 data channel; reset and start channel (RSC), and start channel (STC) instructions for the 7909 data channel.

During the initial 7094 I cycle, the storage bus is gated into the storage register, program register, and tag register as a normal I time function (Figures 32, 33).

At I7 time, SR(21-35) are routed to the address register. During I9(D2), address modification takes place with the specified index register (if any), and the new address is returned to the address register. If a HIP halt occurs, this address will never make a direct reference to core storage. However, being in the address register makes this effective address available to the 7040 program during the special compatability transmit instruction.

POD 54 and other PR decoding immediately create a "select channel" signal to the attached channel (Figure 32). If this selected channel is an exempt channel, HIP mode gated is prevented from occurring and, therefore, allows the POD 54 instruction to complete its execution in the normal manner.

If the selected channel is not exempt, HIP mode gated allows the I/O POD trigger to be turned on at I9 time. This trigger in turn deactivates the select channel signal (Figure 32). The select channel signal is therefore active to the channel for a period of approximately three clock pulses but is not of sufficient duration to produce any effects.

If an indirect address is specified, the POD 54 instruction proceeds immediately to an E(IA) cycle and performs all of the normal indirect addressing functions. At the end of the IA cycle the address register contains the new, modified, effective address and is again available to the 7040 program during the special compatability transmit instruction.

If a HIP halt condition exists, the POD 54 instruction will never take a normal E cycle because a "proceed to E" signal cannot be returned from a data channel. Because the instruction can never end-op from channel conditions, an end-op is forced by the I/O POD trigger during the L cycle.

During the I cycle that follows, the force HPR trigger is turned on and bits are forced into PR(1,5). The storage bus is prevented from being set into the program register during this next I cycle, and the 7094 proceeds to execute the forced HPR (HIP halt). As a result of the HIP halt, identification bit signals are sent to the 7040 for future storing into the decrement of location 00034_8.

The HIP halt condition also blocks the normal program counter advance at I8 time of the HPR. This blocking insures that the program counter value reflects the location of the POD 54 instruction +1.

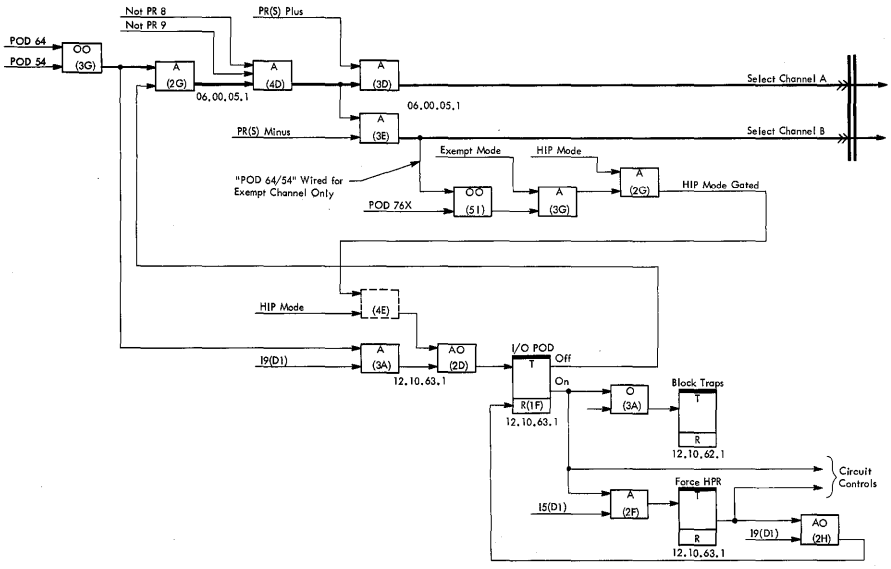


Figure 32. POD 54/64 Condensed Logic

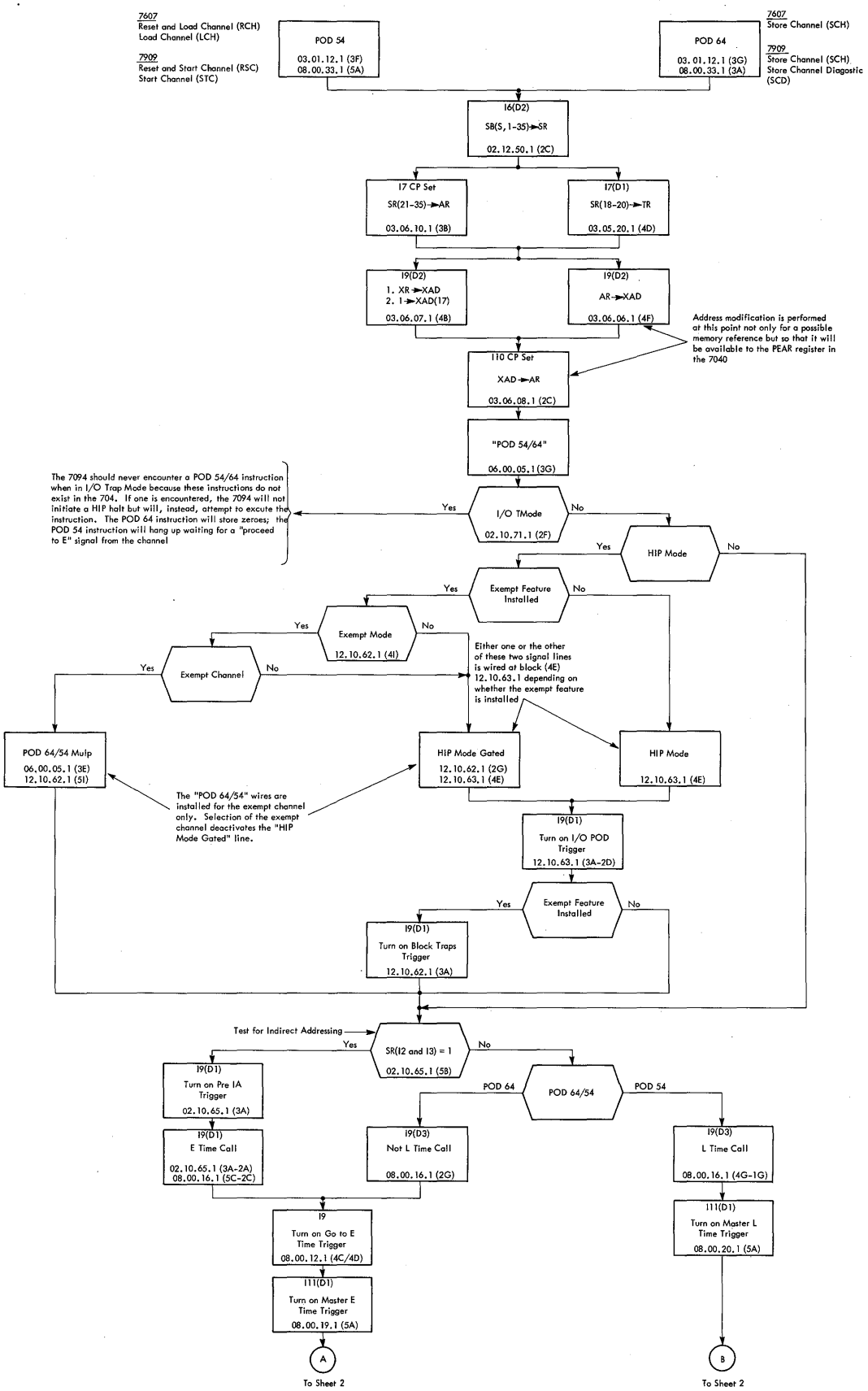


Figure 33. POD 54/64 Flow Chart (Sheet 1 of 2)

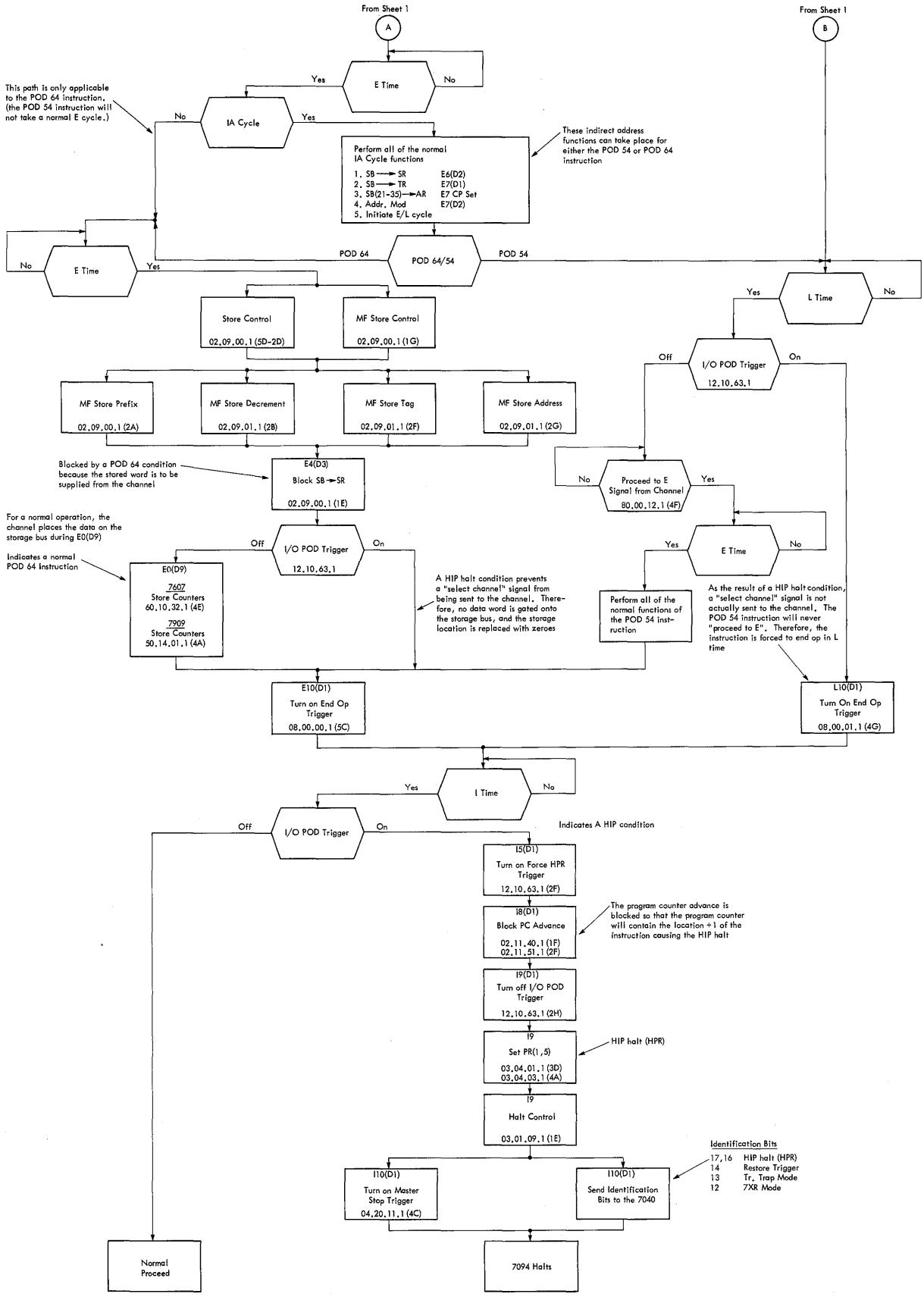


Figure 33. POD 54/64 Flow Chart (Sheet 2 of 2)

Store Channel (POD 64)

- **The POD 64 instructions operate in a normal manner when not in HIP mode or when selecting an exempt channel.**
- **When a HIP halt condition exists, the POD 64 instruction stores zeros into the specified core storage location.**
- **If indirect addressing is specified, an IA cycle occurs before the normal E cycle and HIP halt.**

The following description includes all of the POD 64 instructions: store channel (SCH) for both the 7607 and 7909 data channels, and store channel diagnostic (SCD) for the 7909 data channel.

During the initial I cycle, the storage bus is gated into the storage register, program register and tag register as explained previously for POD 54 instructions (Figure 33). Address modification is performed and the effective address is placed into the address register.

If indirect addressing is specified, the POD 64 instruction goes immediately to an E(IA) cycle and performs all of the normal indirect addressing functions. At the end of the IA cycle, the address register contains the new, modified, effective address.

As explained in the preceding section, program register decoding creates an initial select channel signal to the attached channels (Figure 32). If the selected channel is exempt, the POD 64 instruction completes its execution in the normal manner.

If the selected channel is not exempt, the I/O POD trigger is turned on at I9 time. This trigger, in turn, deactivates the "select channel" signal (Figure 32). Because this signal is only active a short period of time (approximately three clock pulses), it produces no effect in the data channel.

The POD 64 instruction proceeds into the normal required E cycle. During this time, a core storage location is selected as specified by the address register. If a HIP condition exists, information will not be placed on the storage bus by the channel (there is no channel selected) and zeros replace the addressed core storage location.

At this point the core storage location contains zeros. The 7040 program, however, has knowledge of this location through use of the special compatability transmit instruction and can supply the necessary data before restarting the 7094.

The POD 64 instruction ends-op through normal circuitry during the E cycle. During the I cycle that follows, the force HPR trigger is turned on and the 7094 comes to a HIP halt. Identification bits are sent to the 7040 for future storing into the decrement portion of location 00034_8.

The HIP halt condition blocks the program counter advance at I8 time of the HPR to insure that the program counter reflects the location of the POD 64 instruction +1.

Enable (ENB)

- **The enable instruction always completes its normal execution.**
- **A HIP halt (when in HIP mode) occurs immediately following execution of the enable instruction.**
- **If indirect addressing is specified, the IA cycle occurs before the normal E cycle and HIP halt.**
- **Exempt mode has no effect on the enable instruction when in HIP mode.**

During the initial 7094 I cycle, the storage bus is gated into the storage register, program register, and tag register as a normal I time function. Address modification is performed and the effective address is placed in the address register (Figures 34 and 35).

If indirect addressing is specified, the enable instruction goes immediately to an E(IA) cycle and performs all of the normal indirect addressing functions. At the end of the IA cycle, the address register contains the new, modified, effective address.

Program register decoding (Figure 35), sends an enable signal to all channels for control gatings during the normal E cycle. An early E time pulse resets the enabling triggers in all channels including the multiprocess channel (7040). This reset disables the MPT trap in the 7040.

A late E time pulse sets the channel enable triggers as indicated by the bit mask read from storage. Note that when running existing 7094 programs in HIP mode, there will be no multiprocess channel (7040) enabling bit present in the mask because the 7040 is, in effect, a channel which never existed before.

The bit mask provides the only channel control of the enable instruction. The bit mask does not cause a select channel signal to be sent to a particular channel. Because of this, exempt mode has no effect on the enable operation. If the system is in HIP mode, a HIP halt occurs regardless of the status of exempt mode.

The restore trigger in the 7094 is turned on at E1 time. The I/O POD and block traps triggers were both turned on at I9 time; therefore, any waiting data channel traps are not allowed to occur following the enable instruction. Instead, the 7094 initiates a HIP halt and trap to the 7040.

The enable instruction ends-op through normal circuitry during the E cycle. During the following I cycle, the force HPR trigger is turned on and the 7094 comes to a HIP halt. Identification bits are sent to the 7040 for future storing into the decrement of location 00034_8.

The HIP halt condition blocks the program counter advance at I8 time of the HPR to ensure that the program counter reflects the location of the enable instruction +1.

Restore Channel Traps (RCT)

- **The RCT instruction always completes its normal execution.**
- **A HIP halt (when in HIP mode) occurs immediately following execution of the RCT instruction.**
- **Exempt mode has no effect on the RCT instruction when in HIP mode.**

During the 7094 I cycle, the storage bus is gated into the storage register, program register, and tag register as a normal I time function (Figure 35). Address modification is performed (even though it might seem illogical for an RCT instruction), and the effective address is placed into both the address register and the shift counter. Note that address modification in cases of POD 760 instructions can actually change the instruction itself.

The RCT instruction goes into an L cycle and turns on the restore trigger. This is the only logical function of the RCT. Because the full RCT decoding is not available until the beginning of L time, the I/O POD and block traps triggers are not turned on until L1 time. Any waiting data channel traps are not allowed to occur following the RCT instruction; instead, the 7094 initiates a HIP halt and trap to the 7040.

The RCT instruction does not cause a "select channel" signal to be sent to a particular channel. Because of this, exempt mode has no effect on the RCT operation. If the system is in HIP mode, a HIP halt occurs regardless of the status of exempt mode.

During the following I time, the force HPR trigger is turned on and the 7094 comes to a HIP halt. Identification bits are sent to the 7040 for future storing in the decrement portion of location 00034_8.

The program counter advance is blocked at I8 time of the HPR to insure that the program counter reflects the location of the RCT instruction +1.

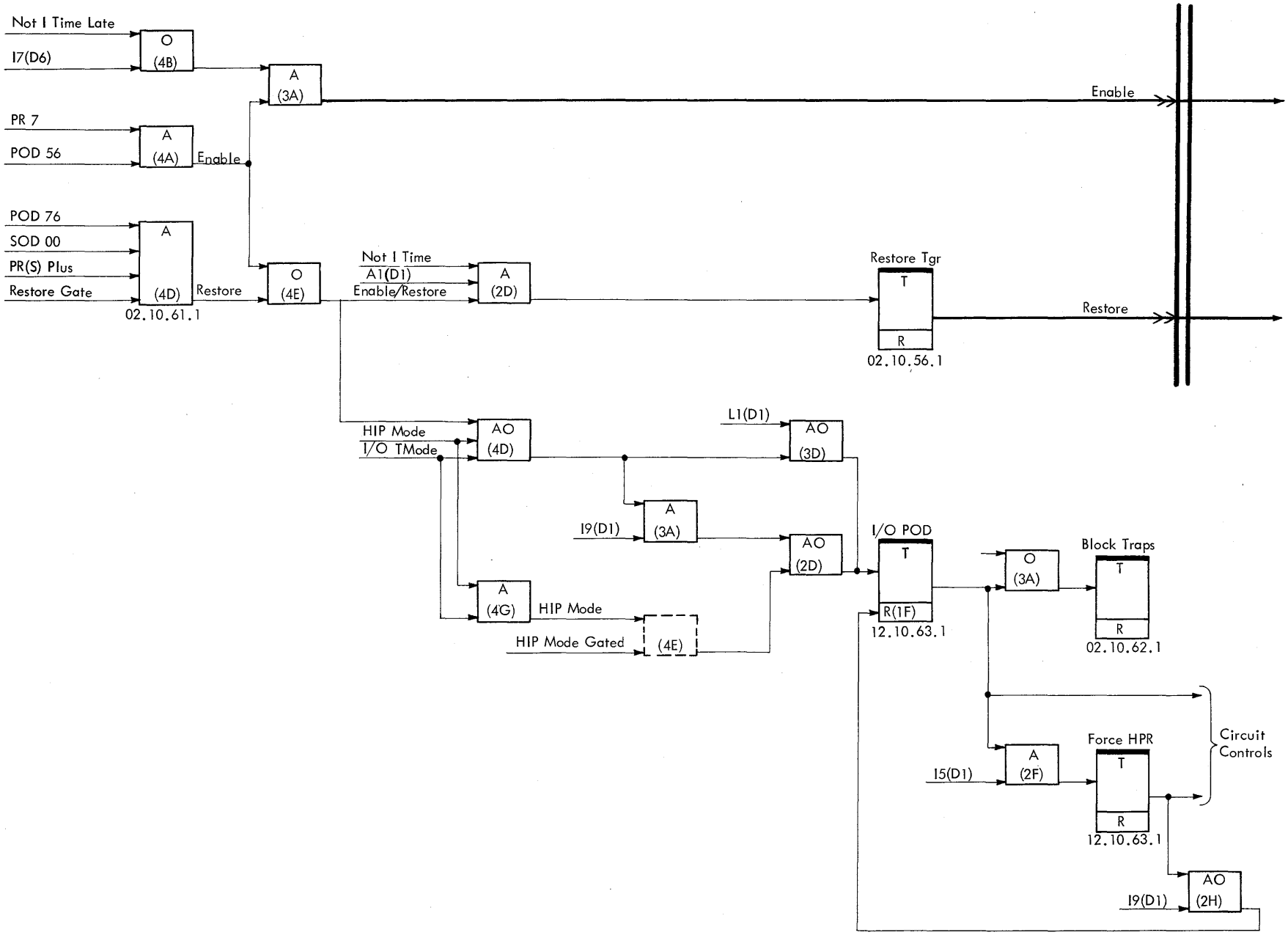


Figure 34. ENB/RCT Condensed Logic

Figure 35. ENB/RCT Flow Chart

7040 Trap to 7094

- **The system must be in multiprocess mode.**
- **Two trap instructions:**
  MPT — Multiprocess Trap
  MFT — Multiprocess Forced Trap
- **7094 trap locations 00003 and 00004 are used (direct data trap cells).**

The 7040 communicates with the 7094 by means of trapping. Two 7040 instructions are provided to trap the 7094 when the system is in multiprocess mode.

The multiprocess trap (MPT) is considered a "polite" trap because it requires enabling and I time in the 7094 for completion. On the other hand, the multiprocess forced trap (MFT) does not require enabling, and can force the 7094 to end operation prematurely during an L cycle.

Both of these trap instructions appear as a direct data trap in the 7094 by using locations 00003 and 00004. A bit in position 18 of location 00003 identifies the multiprocess trap.

Multiprocess Trap MPT —1775 (I, L)

- **The MPT instruction will initiate a 7094 trap only if the system is in multiprocess mode.**
- **An MPT trap (when not enabled) remains stacked until enabled by a 7094 ENB instruction or by a 7040 start and enable instruction.**
- **An MPT trap remains stacked until the 7094 is in a restore condition.**
- **An MPT trap (when enabled in multiprocess mode) traps the 7094 to location 00003 to store the program counter; and location 00004 for instruction execution.**
- **A bit in position 18 of location 00003 of the 7094 identifies the MPT/MFT trap.**
- **An MPT instruction causes a 7040 program skip if there is no stacked trap.**
- **An MPT instruction executes the next sequential instruction (no program skip) if a stacked trap condition exists.**
- **Leaving multiprocess mode does not nullify a stacked MPT trap. The stacked trap will be executed when:**

  The 7094 becomes restored for accepting subsequent traps, or,

  The MPT is enabled by a 7094 enable instruction issued to the multiprocess channel, or by a 7040 start and enable (SRC . . . 04) instruction executed in multiprocess mode.
- **A stacked MPT trap is reset by execution of an MFT trap.**
- **An MPT instruction executed out of multiprocess mode does not initiate an MPT trap to the 7094; however,**

  A 7040 program skip *does occur* if there is no stacked trap condition.

  A 7040 program skip *does not* occur if a stacked trap condition exists.

The MPT instruction is a means by which the 7040 can interrupt and modify the course of the 7094 program.

During the 7040 I cycle of the MPT instruction, the storage bus is gated into the storage register and program register at I3 time as a normal I time function (Figure 36, sheet 1; Figure 37).

At I5(D1) the shift counter is set from AD(28-35) and the address register is set from AD(21-35). These settings occur as normal POD 7X functions but perform no logic in the overall trap operation.

The 7090 MPT trap trigger is not turned on in the 7040 until late in the L cycle (L5 time). The purpose of this is to allow the previous status of the trigger to be tested at L3 time so that a possible program skip can be initiated as determined by a "stacked" trap condition. If the MPT trigger is off, program counter skip circuitry is initiated at L3 time (Figure 36, sheet 1; Figure 37). If the MPT trigger is on (stacked trap condition), program counter skip circuitry is not initiated and the 7040 program proceeds with the next sequential instruction. Note that these two skip conditions are not dependent on multiprocess mode and occur every time the MPT instruction is executed.

Once the MPT condition has been set up, trapping is suspended until enabled to trap the 7094. Enabling can be accomplished in one of two ways:

1. By a 7094 enable (ENB) instruction with a CW bit corresponding to the multiprocess channel.
2. By a 7040 start and enable (SRC . . . 04) instruction.

With the system properly enabled, the 7040 trap sync trigger is turned on by a 7094 A5(D2) pulse and a

7040/44 trap demand signal is sent to the 7094. The 7094 clock pulse provides proper synchronization of the trap demand with the 7094 circuitry.

Before proceeding into the 7094 operation, note that the 7040 MPT trigger and trap sync trigger remain on until reset by a 7094 E6(D2) pulse of the 7094 STR operation.

The 7040/44 trap demand signal enters the 7094 and turns on the pre-interrupt trigger with the first A10(D1) pulse (Figure 36, sheet 2). If the 7094 is in an I cycle at this time, the IBR loaded trigger is reset and overlap is nullified so that the trap demand may interrupt the program sequence following the current instruction.

The MPT is normally serviced during the following I time after being received. The following special conditions, however, as indicated by the Note 1 decision block (Figure 36, sheet 2) can delay this servicing:

1. If the computer is in the process of executing a POD 34 (CAS/LAS) instruction, the program counter is incremented to anticipate a skip 1 or 2 condition and therefore does not indicate the correct value to be stored in trap location 00003.

2. An execute (XEC) instruction also presents the problem of a correct value being in the program counter. The instruction being executed might cause the program counter value to be changed because of a successful transfer or skip condition.

3. Any trap demand arriving during the execution of a read select (SOD 02), write select (SOD 06), enable (ENB), or restore channel traps (RCT) instruction is delayed for one instruction. For the read or write select, this delay prevents a possible I/O check by allowing a reset and load channel (RCH) instruction to be executed before entering the trap subroutine. For the enable instruction, or the restore channel traps instruction, the delay allows a transfer to be completed back to the main program.

4. A floating-point trap also assumes priority over an interrupt trap. This condition is not considered in the Note 1 decision block, however, but is a gating factor when turning on the interrupt trigger during the next I cycle.

5. The restore trigger must be ON to allow the interrupt to proceed. If this trigger is OFF, it may be turned on only by a 7094 enable or restore channel traps instruction.

At this point a decision block tests whether or not the 7094 is halted at the time of the "trap demand" (Figure 36, sheet 2). Both conditions should be considered.

If the 7094 is in the process of executing a program at the time of the MPT trap, the current instruction is allowed to be completed in the normal manner before the trapping circuits are activated.

Because of the multiprocess programming techniques used, however, the 7094 will most likely be stopped with either a halt and transfer (HTR) or HIP halt (HPR) when the MPT trap occurs. In these halt cases it is necessary to restart the 7094 to allow the trap to take effect. When the 7094 halt occurs, the L time trigger turns on but its output is immediately blocked by the B cycle interrupt trigger. Restarting the computer therefore requires turning off the master stop and B interrupt triggers. When this happens, the halt instruction (HTR/HPR) completes its operation in L time and allows the program to continue.

At L10 time either the program counter or address register is gated to MAR as a normal end-op function (Figure 36, sheet 2). This memory reference performs no logic at this time because the instruction being fetched will not actually reach the storage register or program register. Instead, an STR instruction will be forced into the program register to complete the trap (interrupt) operation.

The actual trap operation does not start until the mid-point in the I cycle following completion of the current instruction. Between I0-I6 time, the current instruction is completing its operation and the program counter is updated to its correct value in the case of successful transfers. If the computer has been started from a halt (HTR) condition, the conditions met trigger is turned off to prevent updating the program counter from the address register.

The interrupt trigger is turned on at I5 time and the STR is forced into the program register at I6 time (SB to PR circuitry is blocked). Note that turning on the interrupt trigger at 5 time allows the interrupt type trap to assume priority over data channel traps which are not tested for until I6 time.

Execution of the trap in the 7094 has three main objectives:

1. To store the contents of the 7094 program counter (location of the next instruction to be executed in the computer program) into the address portion of location 00003.

2. To store an identification bit in position 18 of location 00003.

3. To trap the computer program and cause execution of the instruction contained in location 00004.

In accomplishing the first objective, the program counter is blocked from the normal stepping at I8 time of the forced STR. In this manner, the address of the next instruction to be executed in the main program is retained in the program counter.

The address register is reset at I9 time and bits are forced into AR(16, 17) to provide a memory address of 00003.

Storing the program counter occurs during the following E cycle. The program counter value is routed through the index adders and is set into SR(21-35). Positions SR(S, 1-20) are cleared. Double complementing of the program counter and index adder outputs effectively places the true value in the storage register (Figure 36, sheet 4). The storage register is gated onto the storage bus during E4(D3).

As a second objective, a bit is stored in position 18 of memory location 00003. This bit indication is OR'ed to SB(18) by circuitry resulting from the MPT trap. MF store tag is also activitated by essentially the same circuitry to cause the tag portion of location 00003 to be destroyed on readout.

MF store decrement controls resulting from existing STR circuitry also destroy positions 3-17 on readout. It is possible (though remote) that a direct data trap may also occur simultaneously from a 7094 data channel. In this case additional bits would also be stored in the decrement portion of location 00003. Only the prefix portion of location 00003 remains unchanged.

The third objective is accomplished by resetting the program counter at E6 time (its contents have already been sent to storage) and forcing a bit into PC(15). Gating the program counter to MAR causes the instruction at location 00004 to be fetched and the 7094 program proceeds.

During the E cycle of the STR operation, 7094 condition lines and clock pulses are sent to the 7040 to reset the MPT trap trigger and trap sync trigger as shown in Figure 36, sheet 4; Figure 37.

At I1 time of the cycle following the STR, the interrupt trigger is reset and the trap sequence is completed. Note that the restore trigger has been turned off so that all other direct data or data channel traps are suspended until again restored by the 7094 trap subroutine.

Multiprocess Forced Trap MFT —1776 (I, L)

- **The MFT instruction initiates a 7094 trap only if the system is in multiprocess mode.**
- **An MFT trap, when in multiprocess mode, traps the 7094 to location 00003 (to store the program counter) and location 00004 (for instruction execution).**
- **A bit in position 18 of location 00003 identifies the MPT/MFT trap.**
- **No enabling is required.**
- **MFT traps can not be stacked.**
- **An MFT instruction never causes a program skip.**
- **An MFT trap does not require the 7094 to be in a restore state.**
- **An MFT trap initiated during a 7094 L cycle causes that 7094 operation to end-op at the end of that particular L cycle.**
- **An MFT trap resets a stacked MPT trap.**

The MFT instruction is a means by which the 7040 program can "forcibly" trap the 7094 computer.

The MPT instruction requires enabling conditions to complete the trap and might, therefore, be considered as a "polite" trap. The MFT trap requires no enabling or 7094 end-op conditions and is, therefore, a true forced trap. This "forced" trap feature allows the 7040 control program to regain control of a 7094 program which has either become "hung" in a program loop or has executed an I/O instruction to a non-existent device and has no way of ending-op.

All of the initial I time functions described previously for the MPT trap also apply for the MFT trap (Figure 36, sheet 1; Figure 37).

During the L cycle in the 7040 no test is made for a stacked trap condition. The program counter continues to indicate the address of the next sequential instruction. Also, no enabling is needed to initiate the forced trap to the 7094. Note (Figure 36, sheet 1) that an MFT initiates an additional signal to the 7094, "40/44 end-op." It is this signal line which allows the 7040 to unconditionally interrupt the 7094.

In the 7094 the MPT trap required the restore trigger to be on to continue (Figure 36, sheet 2); the MFT bypasses this condition with no concern as to whether the 7094 restore trigger is on or off.

If the 7094 is halted at the time of the trap, the operation is as explained in the previous MPT section. If the 7094 is in the process of executing a program, however, the MFT causes an end-op condition at the end of the current instruction or at the end of the next L cycle — whichever occurs first (Figure 36, sheet 2). For example, a trap received during an ADD (I, E), DST (I, E, E), etc. allows the instruction to finish correctly. A trap received during an LRS (I, L, L, L, L) or MPY (I, E, L, L, L), for example, could cause the instruction to end-op prematurely with incorrect results in the CPU registers.

The remainder of the MFT trap operation in the 7094 is the same as explained previously for the MPT trap.

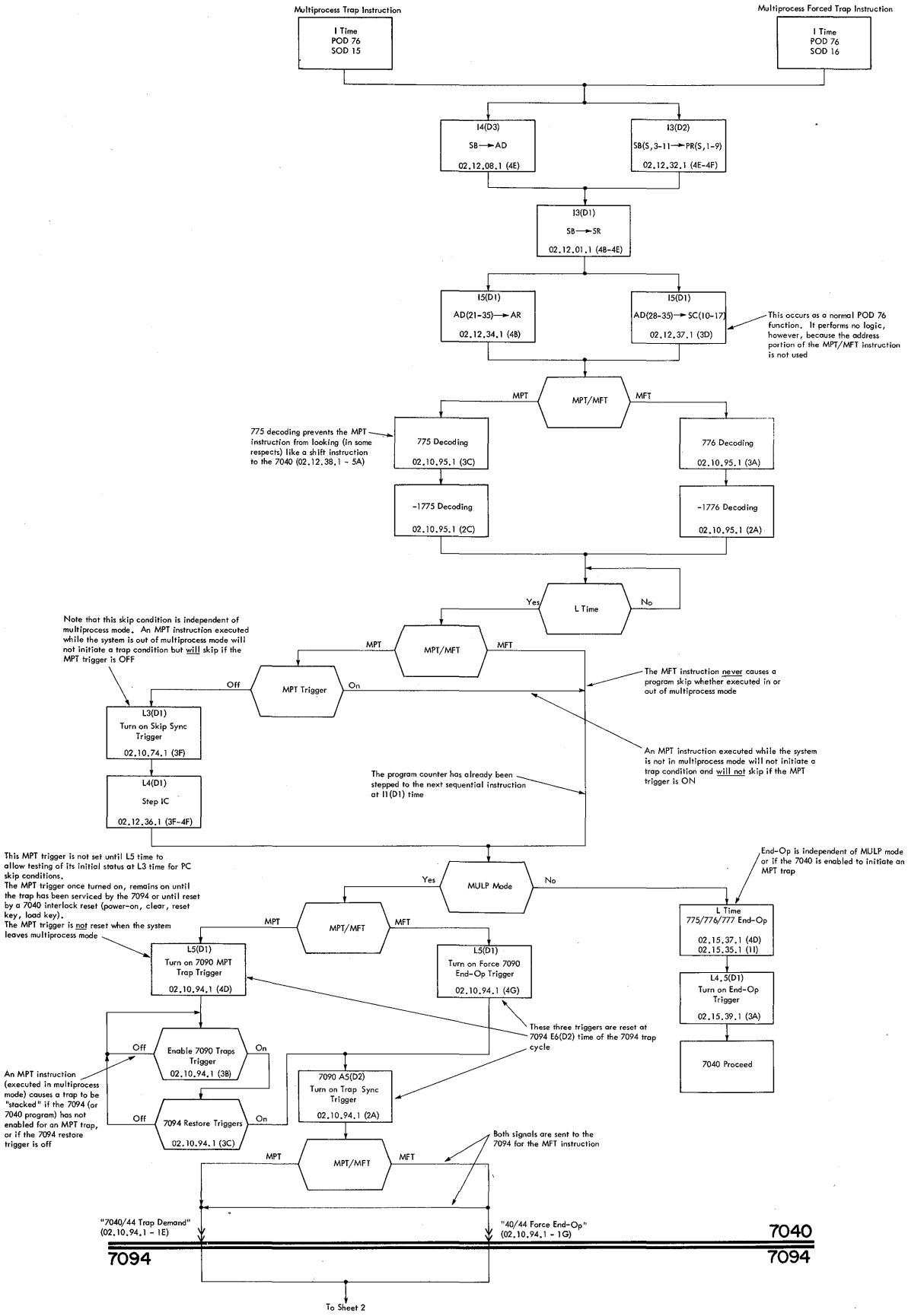


Figure 36. MPT and MFT Flow Chart (Sheet 1 of 4)

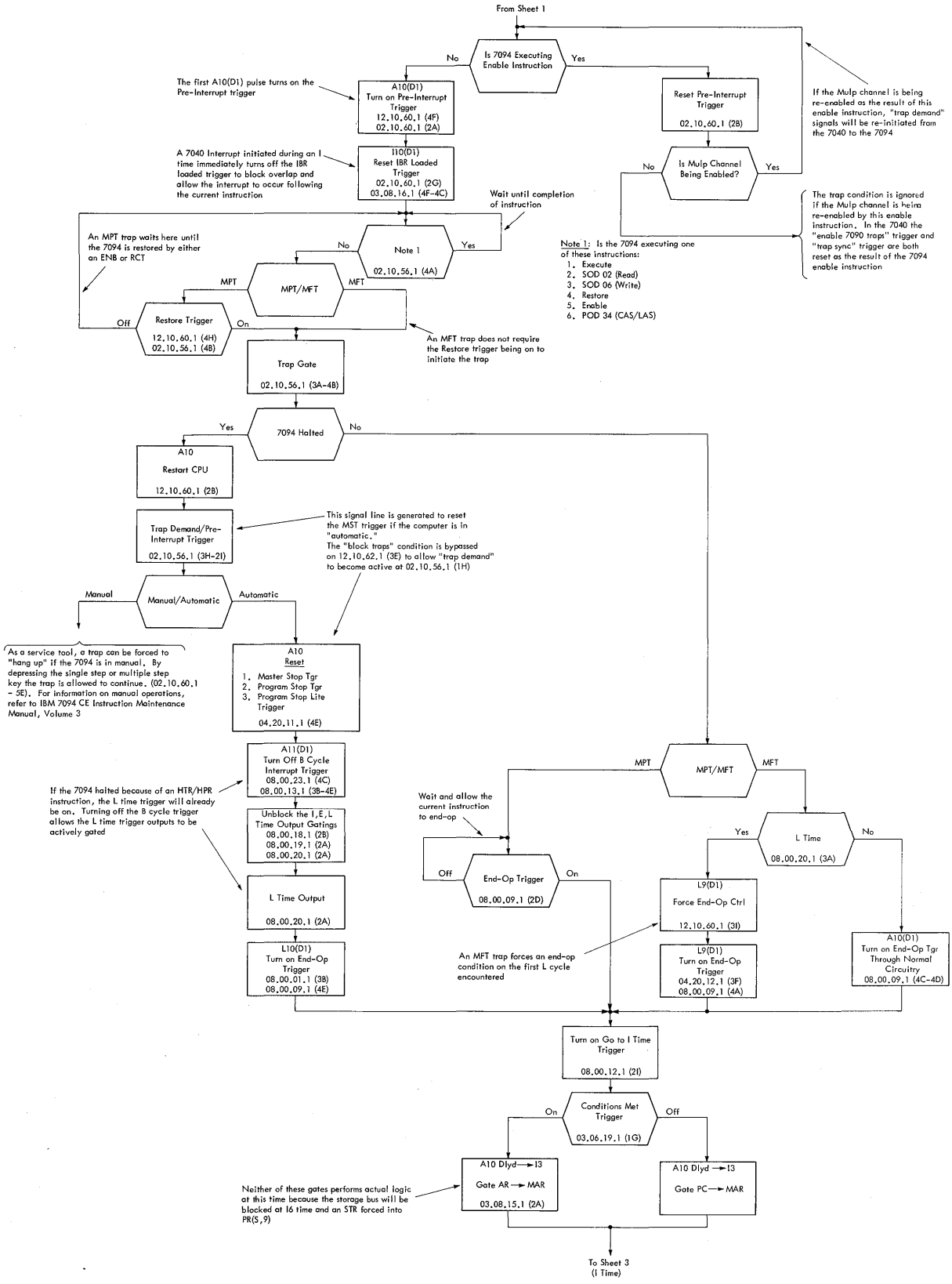


Figure 36. MPT and MFT Flow Chart (Sheet 2 of 4)

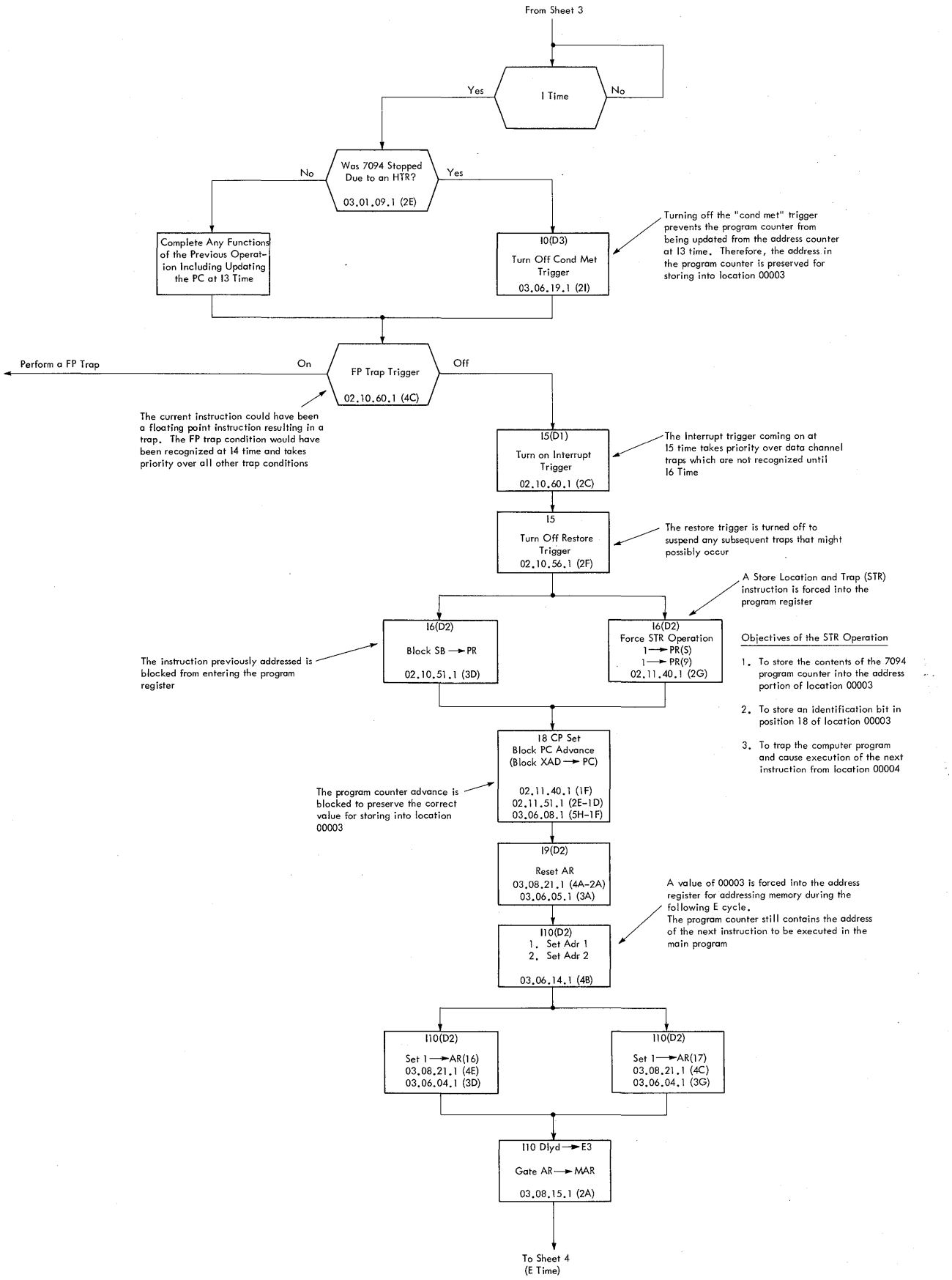


Figure 36. MPT and MFT Flow Chart (Sheet 3 of 4)

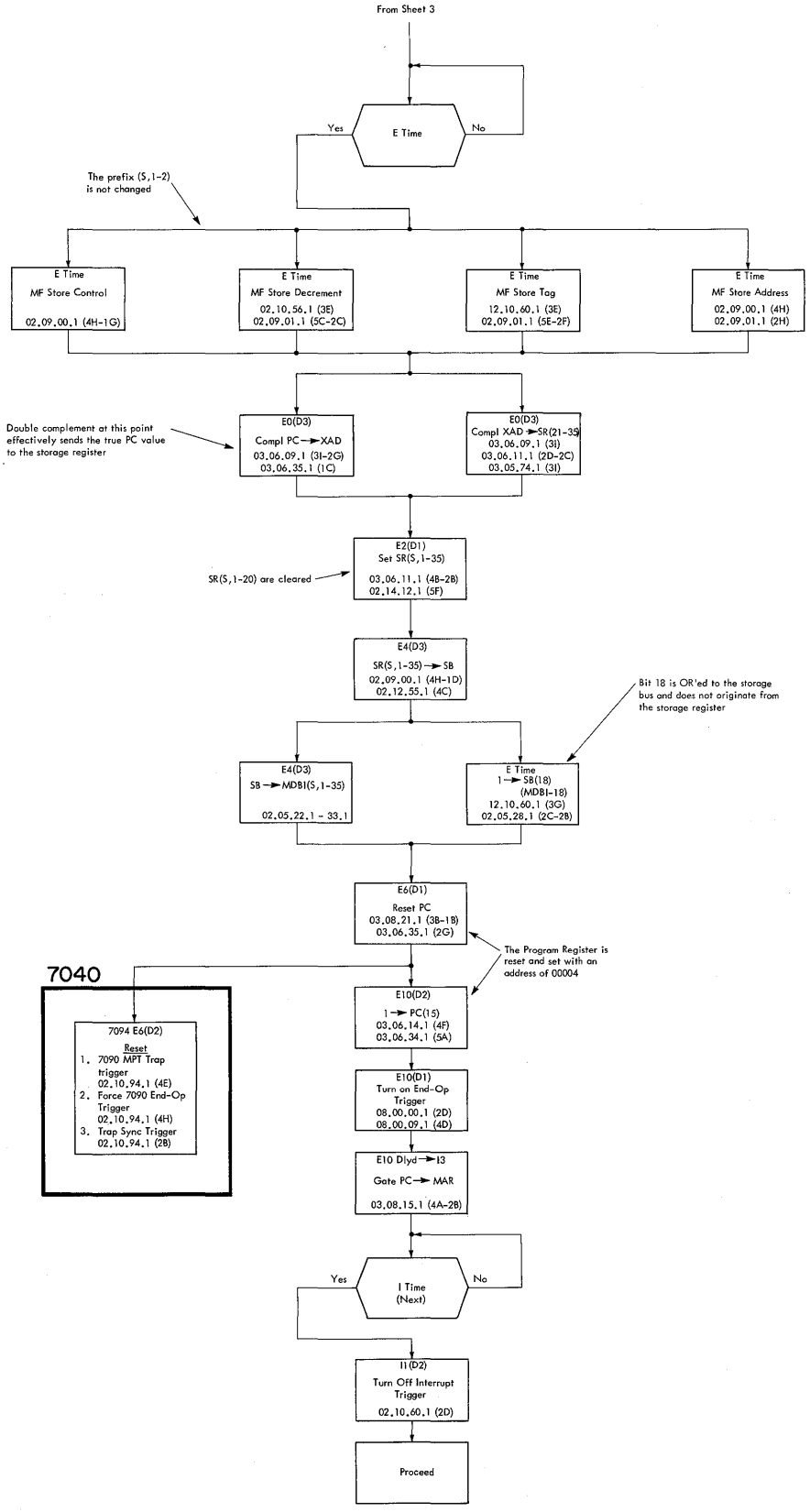


Figure 36. MPT and MFT Flow Chart (Sheet 4 of 4)

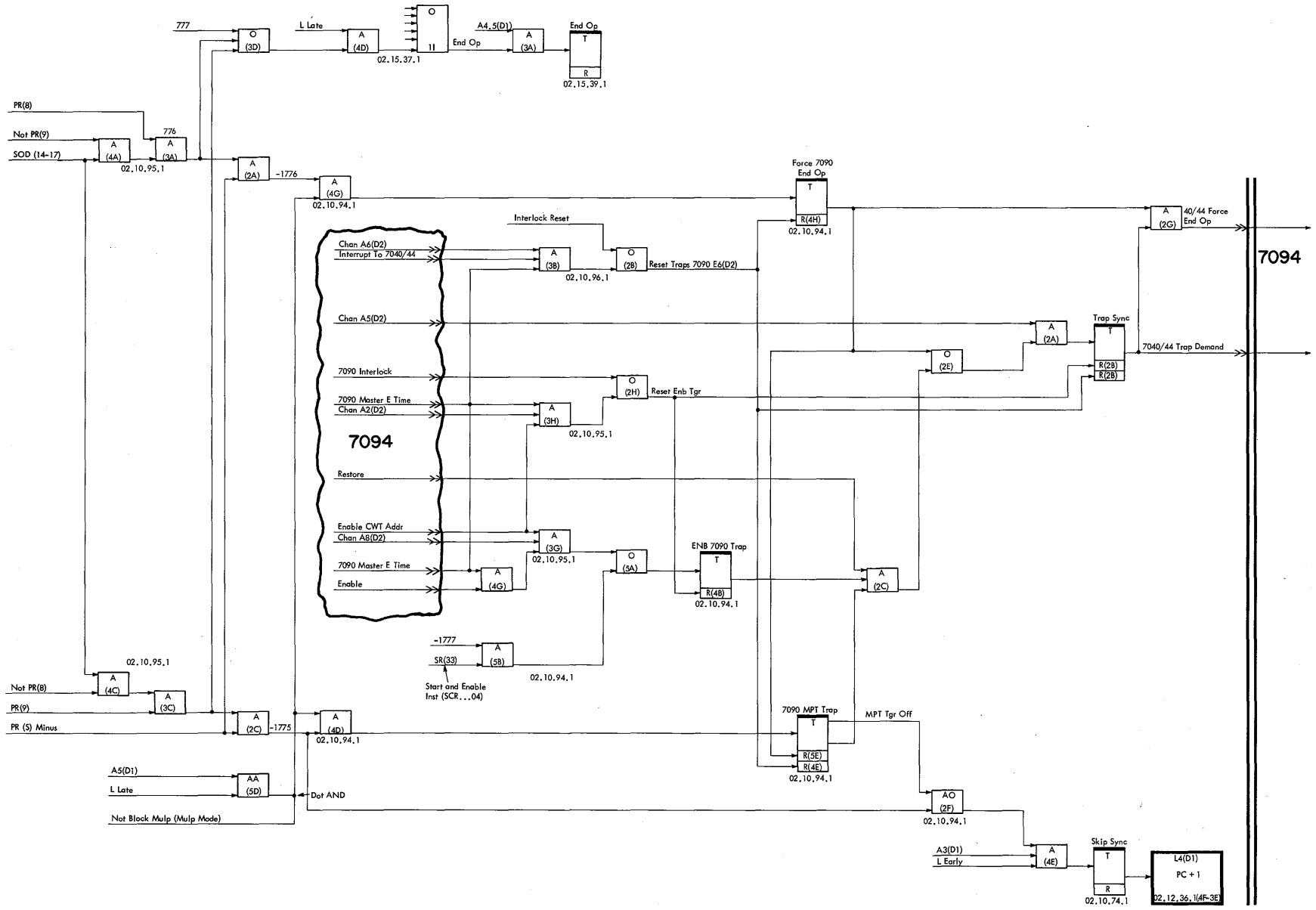


Figure 37. 7040 MPT/MFT Condensed Logic

7094 Trap to 7040

- **The multiprocess trap uses location 00034_8 for storing the 7040 instruction counter and identification bits; and location 00035_8 for the next instruction to be executed.**
- **The 7040 must be enabled to allow completion of the multiprocess trap. (Requires an enable instruction with a bit in position 26 of the enable mask.)**
- **The 7040 must be in multiprocess mode to allow completion of the multiprocess trap.**
- **A trap will be stacked if the 7040 is in multiprocess mode and is not enabled.**
- **Stacked trap indications are held reset when not in multiprocess mode.**
- **The multiprocess trap is one priority level higher than a direct data trap.**
- **A pre-interrupt memory protect trap results from a multiprocess trap request if the 7040 is in memory protect mode.**
- **An IT blast trap resets all multiprocess trap conditions in the 7040.**

The 7094 communicates with the 7040 by means of trapping. Except for the identification bits and new 7040 trap locations (00034_8 and 00035_8), the multiprocess trap operates similarly to the present 7040 direct data or data channel trap.

The 7094 can trap the 7040 for a variety of reasons. As a result of the trap, identification bits are stored into the decrement portion of location 00034_8. These bits and their corresponding meanings are shown in Figure 38.

Various combinations of bits are possible. Figure 39, even though not all inclusive, shows various combina-

ITEM	BIT (S)	CONDITION (S)	COMMENTS
I	12	Load cards	Load cards key pressed on the 7151 console while in HIP mode *
II	13	Load tape	Load tape key pressed on the 7151 console while in HIP mode *
III	14	Enter instruction I/O	Results from an I/O instruction entered into the computer through the 36 console input switches as a manual enter instruction while in HIP mode *
IV	15	I/O call	BTT instruction executed by the 7094 to the direct couple channel
V	16	Halt	Results from an HTR halt in the 7094
VI	16, 17	HPR/DVH/ VDH/FDH/ DFDH	Results from a stop in the 7094 caused by either an HPR or DVH/VDH/FDH/DFDH instruction. The HPR could be the result of a HIP mode stop. (Bit 17 never ocurrs alone)
VII	12 with V or VI	Halt and 7-XR mode	The 7094 was in 7 index register mode (not multiple tag mode) when it trapped the 7040 #
VIII	13 with V or VI	Halt and transfer trap mode	The 7094 was in transfer trap mode when it trapped the 7040 #
IX	14 with V or VI	Halt and channel trap control on	The 7094 was enabled and able to accept a channel trap at the time it trapped the 7040. (Trap control indicator is on at the 7151 console) #

* These traps facilitate maintenance from the 7151 console when the 7094 configuration does not contain I/O equipment. As a result of the trap, a pseudo-operation is performed from the 7040

Multiple conditions of items VII, VIII and IX are possible (i.e. bits 12, 13 and 14 together with V or VI)

Figure 38. 7040 Trap Condition Bits

NORMAL CONDITIONS CAUSING THE TRAP	Manual Load Cards	7XR Mode	Manual Load Tape	Transfer Trap Mode	Enter I/O Instruction	Restore Trigger	Mulp Call (BTTM)	HTR/DVH/VDH/FDH/DFDH/HPR (or HIP halt)	DVH/VDH/FDH/DFDH/HPR (or HIP halt)
	12		13		14		15	16	17
HTR								✓	–
DVH/VDH/FDH/DFDH/HPR (or HIP halt)								✓	✓
BTTM							✓	–	–
BTTM followed by an HTR							✓	✓	–
BTTM followed by an HPR							✓	✓	✓
Manual I/O instruction in HIP mode					*✓	–	–	–	–
Manual I/O instruction of a BTTM					✓	–	✓	–	–
BTTM followed by an HTR, Restore Tgr					–	✓	✓	✓	–
BTTM followed by an HPR, Restore Tgr					–	✓	✓	✓	✓
Manual Load Tape in HIP mode			*✓	–	–	–	–	–	–
HTR and Trap mode				✓	–	–	–	✓	–
DVH/VDH/FDH/DFDH/HPR (or HIP halt) and Trap Mode				✓	–	–	–	✓	✓
BTTM followed by an HTR, Trap mode				✓	–	–	✓	✓	–
BTTM followed by an HPR, Trap mode				✓	–	–	✓	✓	✓
HTR, Trap mode, Restore Tgr				✓	–	✓	–	✓	–
HPR, Trap mode, Restore Tgr				✓	–	✓	–	✓	✓
BTTM followed by an HTR, Trap mode, Restore Tgr				✓	–	✓	✓	✓	–
BTTM followed by an HPR, Trap mode, Restore Tgr				✓	–	✓	✓	✓	✓
Manual Load Cards in HIP mode	*✓	–	–	–	–	–	–	–	–
HTR and 7XR mode		✓	–	–	–	–	–	✓	–
DVH/VDH/FDH/DFDH/HPR (or HIP halt) and 7XR mode		✓	–	–	–	–	–	✓	✓
BTTM followed by an HTR, 7XR mode		✓	–	–	–	–	✓	✓	–
BTTM followed by an HPR, 7XR mode		✓	–	–	–	–	✓	✓	✓
HTR, 7XR mode, Restore Tgr		✓	–	–	–	✓	–	✓	–
DVH/VDH/FDH/DFDH/HPR (or HIP halt), 7XR mode, Restore Tgr		✓	–	–	–	✓	–	✓	✓
BTTM followed by an HTR, 7XR mode, Restore Tgr		✓	–	–	–	✓	✓	✓	–
BTTM followed by an HPR, 7XR mode, Restore Tgr		✓	–	–	–	✓	✓	✓	✓
HTR, 7XR mode, Trap mode		✓	–	✓	–	–	–	✓	–
DVH/VDH/FDH/DFDH/HPR (or HIP halt), 7XR mode, Trap mode		✓	–	✓	–	–	–	✓	✓
BTTM followed by HTR, 7XR mode, Trap mode		✓	–	✓	–	–	✓	✓	–
BTTM followed by HPR, 7XR mode, Trap mode		✓	–	✓	–	–	✓	✓	✓
HTR, 7XR mode, Trap mode, Restore Tgr		✓	–	✓	–	✓	–	✓	–
DVH/VDH/FDH/DFDH/HPR (or HIP halt), 7XR mode, Trap mode, Restore Tgr		✓	–	✓	–	✓	–	✓	✓
BTTM followed by an HTR, 7XR mode, Trap mode, Restore Tgr		✓	–	✓	–	✓	✓	✓	–
BTTM followed by an HPR, 7XR mode, Trap mode, Restore Tgr		✓	–	✓	–	✓	✓	✓	✓

* Requires that the system be in both multiprocess and HIP modes.

The other conditions will most likely occur while operating in multiprocess mode but this is not a necessary condition.

Figure 39. 7040 Trap Condition Bit Combinations

tions of bits which can be expected, as well as their probable cause.

Six basic condition signals originate from the 7094 and cause corresponding triggers to be turned on in the 7040 (Figures 40 and 41). These triggers, in turn, cause identification bits to be set into the decrement portion of location 00034_8 at the time of the 7040 trap.

Some of the 7094 identification bit signals can only occur while the system is in HIP mode while others are independent of both HIP and multiprocess modes of operation.

Note: these signals are honored by the 7040 *only* if the system is in multiprocess mode.

An indicator for the enable mulp trap trigger (Systems 02.13.06.2) is located at 01D3B09-9.

Load Cards or Tape

Depression of either the load cards or load tape keys on the 7151 console causes a 7040 trap if the system is, at the same time, in both multiprocess and HIP modes. Depression of either key in HIP mode does not turn on the auto load trigger (Figure 41). Therefore an auto load sequence is not initiated to the channel A card reader or tape unit A1. Note that all of the HIP mode gating used in these operations can not be affected by exempt mode. The load cards and load tape triggers are turned on at A3(D1) time, and off at A8(D1) time to provide an A3(D3) pulse to the 7040 for setting the corresponding identification bit triggers. An output from either trigger is sufficient to initiate the multiprocess trap. Note that a load cards or load tape type of trap does not cause a HIP halt (HPR). Therefore, identification bits 16 and 17 will not be present with bit 12 or 13.

Enter I/O Instruction

Execution of an I/O instruction during a manual enter instruction operation can cause a multiprocess trap. Initiation of the trap requires that the system be in HIP mode and that the I/O instruction is directed to a non-exempt channel (Figures 40 and 41).

Depression of the enter instruction key turns on the enter instruction trigger. Normal manual enter instruction circuitry prevails. The master stop trigger is turned off and the 7094 starts an I cycle to route the instruction from the console keys into the program register. The I/O instruction is detected late in the I cycle and, if in HIP mode, turns on the I/O POD trigger. The output of the I/O POD trigger AND'ing with that of the enter instruction trigger sends a signal to the 7040 to eventually set an identification bit into position 14 of trap location 00034_8.

Note that the enter I/O instruction type of trap does not cause a HIP halt. The I/O POD trigger is reset at I4 time (Systems 12.10.63.1,5E) to prevent the force HPR trigger from being turned on at I5 time (Systems 12.10.63.1,2F). Because there is no HIP halt (HPR), no additional identification bits are sent to the 7040 other than bit 14. An exception to this, however, occurs when a BTT(M) instruction is executed from the console keys. In this case a HIP halt does not occur but the I/O POD trigger does come on for a period of time until L4 time. Because of this both identification bits 14 and 15 are sent to the 7040.

Normal I/O Instruction

A multiprocess trap can occur when the 7094 program tries to execute an I/O instruction to a non-exempt channel while in HIP mode. Under these conditions, the I/O POD trigger is turned on and the 7094 stops with a HIP halt (HPR).

When the master stop trigger is turned on at I10(D1) time of the HPR, identification bit signals 16 and 17 are sent to the 7040 (Figures 40, 41). This same I10(D1) pulse also samples the outputs of the restore, transfer trap mode and multiple tag mode triggers to initiate identification bits 14, 13, and 12 respectively. Each identification bit signal line turns on a corresponding trigger in the 7040 to be remembered until trap time.

If a BTT(M) instruction is being executed, the I/O POD trigger is turned on initially but is reset at L4(D1) time to prevent forcing a HIP halt. A select channel signal is sent to the 7040 to cause setting an identification bit 15.

Normal Halt Instructions

Any one of the 7094 halt instructions (HTR, HPR, DVH, VDH, FDH, DFDH) will initiate a trap signal and identification bits to the 7040 without regard to multiprocess, HIP, or exempt modes of operation (Figures 40, 41).

The same pulse that turns on the master stop trigger initiates the trap signals to the 7040. The outputs of the restore, transfer trap mode, and multiple tag mode triggers are also tested for corresponding identification bits. In addition, bit 16 results if the halt is an HTR; bits 16 and 17 result if the halt is either an HPR, DVH, VDH, FDH, or DFDH. Because of the instruction decoding, bit 17 can never occur alone.

7040 Trap Operation

The six identification bit triggers in the 7040 (Figure 40) are listed as follows:

IDENTIFICATION BIT	TRIGGER NAME	7094 SIGNAL
12	Load Card Bit	Load Card Bit
13	Load Tape Bit	Load Tape Bit
14	Ent I/O Inst	Ent I/O Inst
15	Mulp Call	7090 Channel Address BTT (Mulp)
16	Mulp Stop	7090/94 Halt Control
17	Mulp Norm End	7090/94 HPR Gated

Each of these triggers is turned on with the corresponding 7094 signal. None of these signals are steady levels; they are all pulses of limited duration (Figure

41). These triggers remain on to remember the trap request until the multiprocess trap is completed, or an IT blast occurs, or the computer is manually reset or cleared. Note that all of these triggers are held reset if the system is not in multiprocess mode.

Any of the above conditions (except for bit 17) can initiate a 7040 trap. Before the trap can proceed, however, certain preliminary conditions must be satisfied:

1. The trap control trigger must be on.
2. The channel trap control trigger must be on.
3. An IT overflow trap is not being requested.
4. The system is in multiprocess mode.
5. The 7040 is enabled to process a multiprocess trap.
6. An IT blast trap is not being requested. Note that an IT blast not only delays the multiprocess trap but also resets any evidence of a trap request; the enable mulp trap and all six identification bit triggers are reset.

With all of the blocking conditions removed (except for IT blast), a mulp trap request trigger is turned on at I1 time. At this point the multiprocess trap must find its place in the area of priority. The traps of higher priority are:

1. IT Blast
2. Parity
3. MP Violation
4. Floating-Point

If any one of these traps is also requesting recognition, the mulp trap request trigger is reset at A4 time without being able to turn on the mulp trap trigger. In this case the multiprocess trap is not lost; instead the basic preliminary conditions must be again satisfied before the trap can proceed.

Privileged instructions can also delay the trap for one more instruction time (Figure 41). These instructions include:

1. Enable (ENB)
2. Restore Channel Traps (RCT)
3. Inhibit Channel Traps (ICT)
4. Read Select (RDS)
5. Write Select (WRS)
6. Execute (XEC)
7. Set Protect Mode (SPM)

When all of the previous blocking or delaying conditions have been removed, one more must be considered – MP mode. If the 7040 is in memory protect (MP) mode, a multiprocess trap is suspended in lieu of a pre-interrupt memory protect trap. The object of this pre-I/O trap is to allow the system program to leave MP mode so that the protected areas can be safely used by the multiprocess trap subroutine.

The conditions allowing the trap to begin are probably more involved than the actual trap itself. Up to this point the trap request has been analyzed. The accompanying description now explains the execution of the trap.

The objectives of the 7040 trap are:

1. To store the contents of the 7040 instruction counter (location of the next instruction to be executed in the computer program) into the address portion of location 00034_8.
2. To store identification bits in the decrement portion of location 00034_8.
3. To trap the computer program and cause execution of the instruction contained in location 00035_8.

The trap operation is accomplished by simulating (forcing) major portions of an STR operation. Actual STR decoding from the program register, however, is blocked. The storage bus is also blocked from being set into the program register during the initial I cycle to eliminate any conflict of logic with that being forced by the trap (STR) operation.

The first objective requires blocking a program counter advance at I4(D1) (Figure 41). In this manner, the address of the next instruction is retained in the instruction counter.

The address register is reset at I5(D1); the register is reset from the address generator circuitry with bits in AR(31, 32 and 33). This address of 00034_8 selects the memory location for the following store E cycle.

During the early portion of the E cycle, the instruction counter is routed to the storage register where it is, in turn, placed on the storage bus and set into the memory data register (MDR).

The second objective is accomplished during the store E cycle by gating the output of each of the six identification bit triggers (DOT OR'ing) onto the storage bus. The address and decrement portions of location 00034_8 are set with useful information; the prefix and tag portions are cleared.

The third objective is accomplished (Figure 41) by forcing a bit into AR(35) and allowing the computer to go into an I cycle. The value in the address register, 00035_8, selects that location in memory for the next instruction. Note that the normal circuitry for routing the instruction counter to the address register is blocked. This blocking prevents the instruction counter contents from destroying the generated value of 00035_8.

At I1(D1) the instruction counter is updated from the address register to contain 00035_8. At I2 time, the mulp trap trigger is turned off to complete the trap operation and the 7040 proceeds.

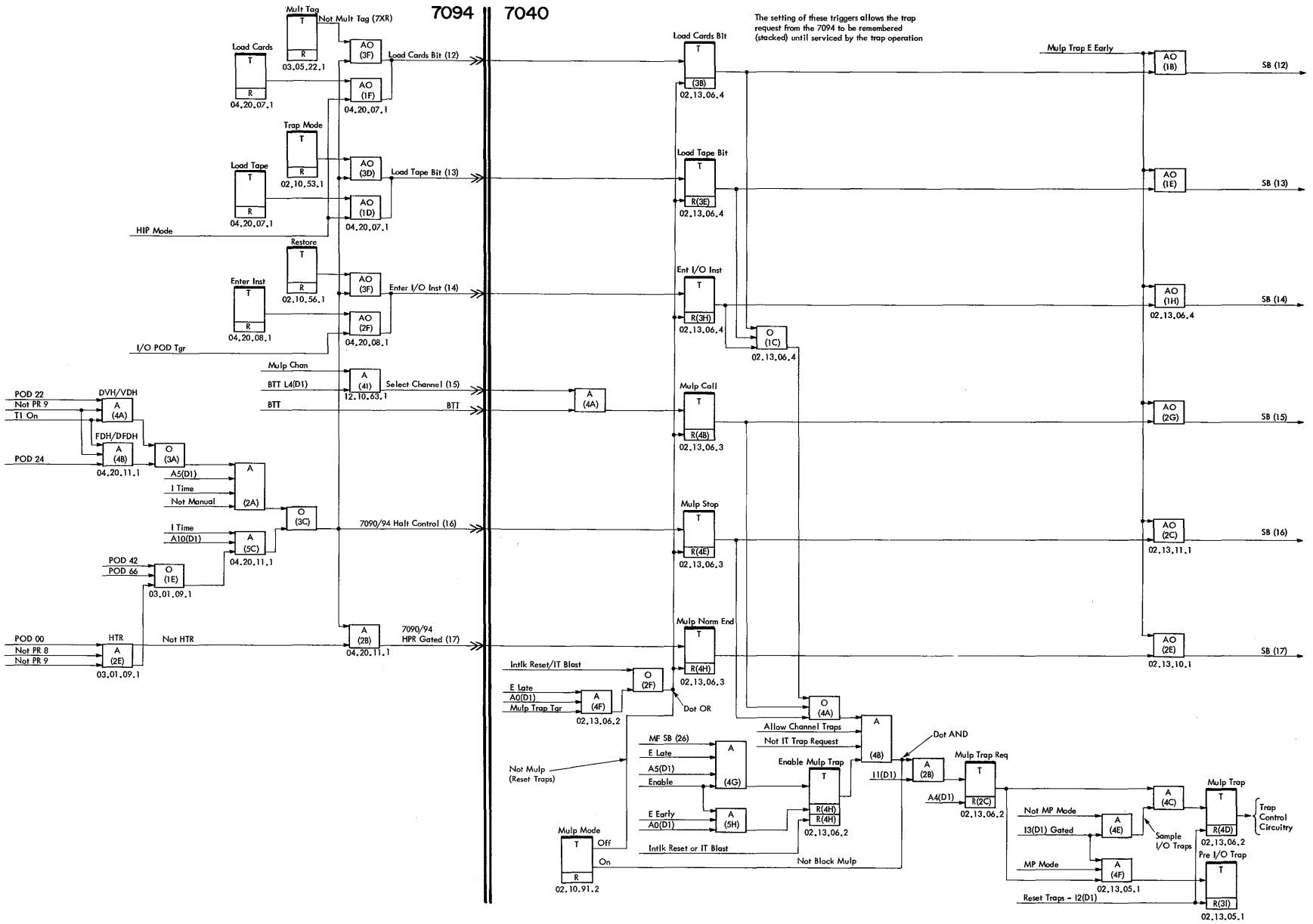


Figure 40. Trap to 7040 Condensed Logic

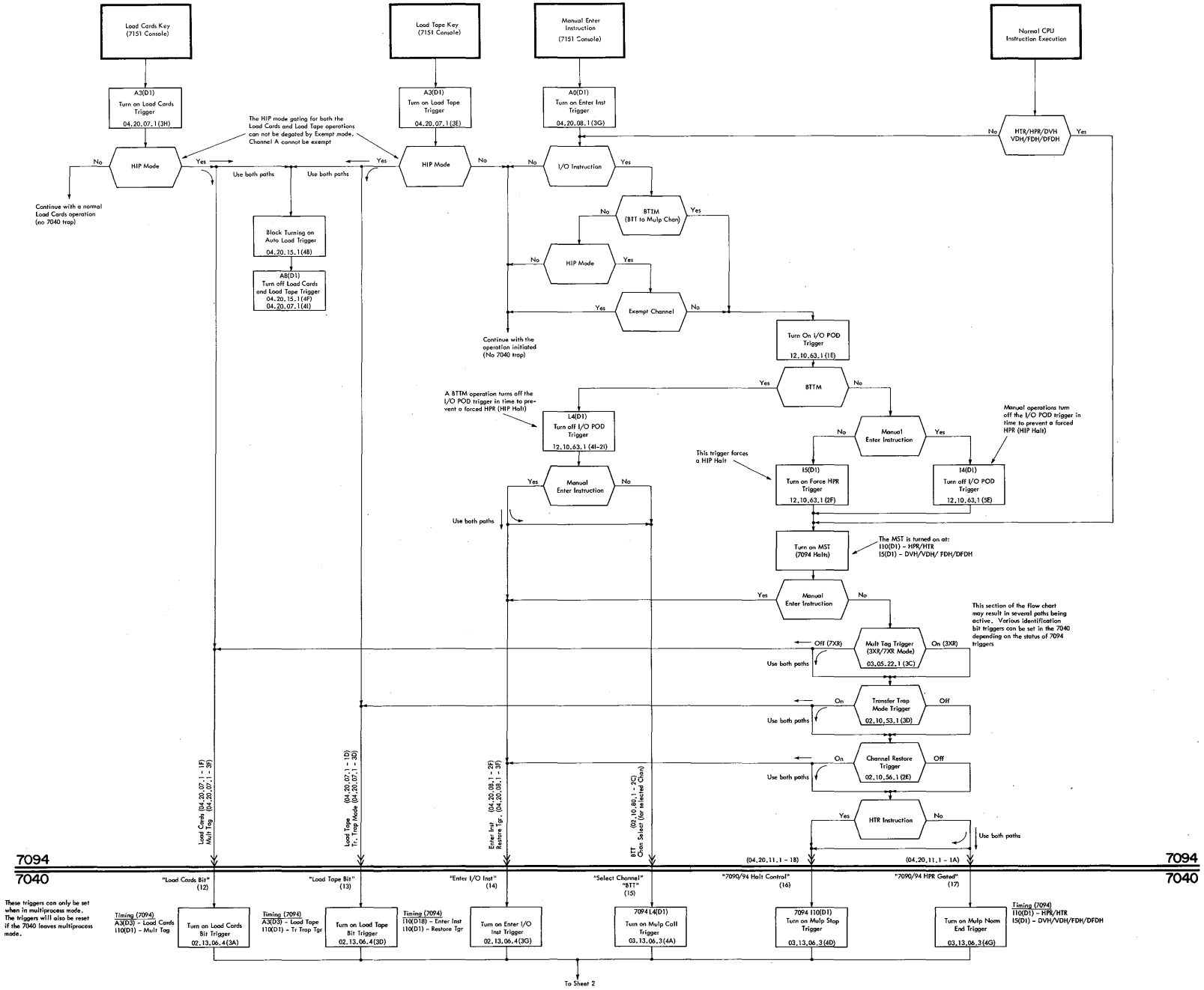
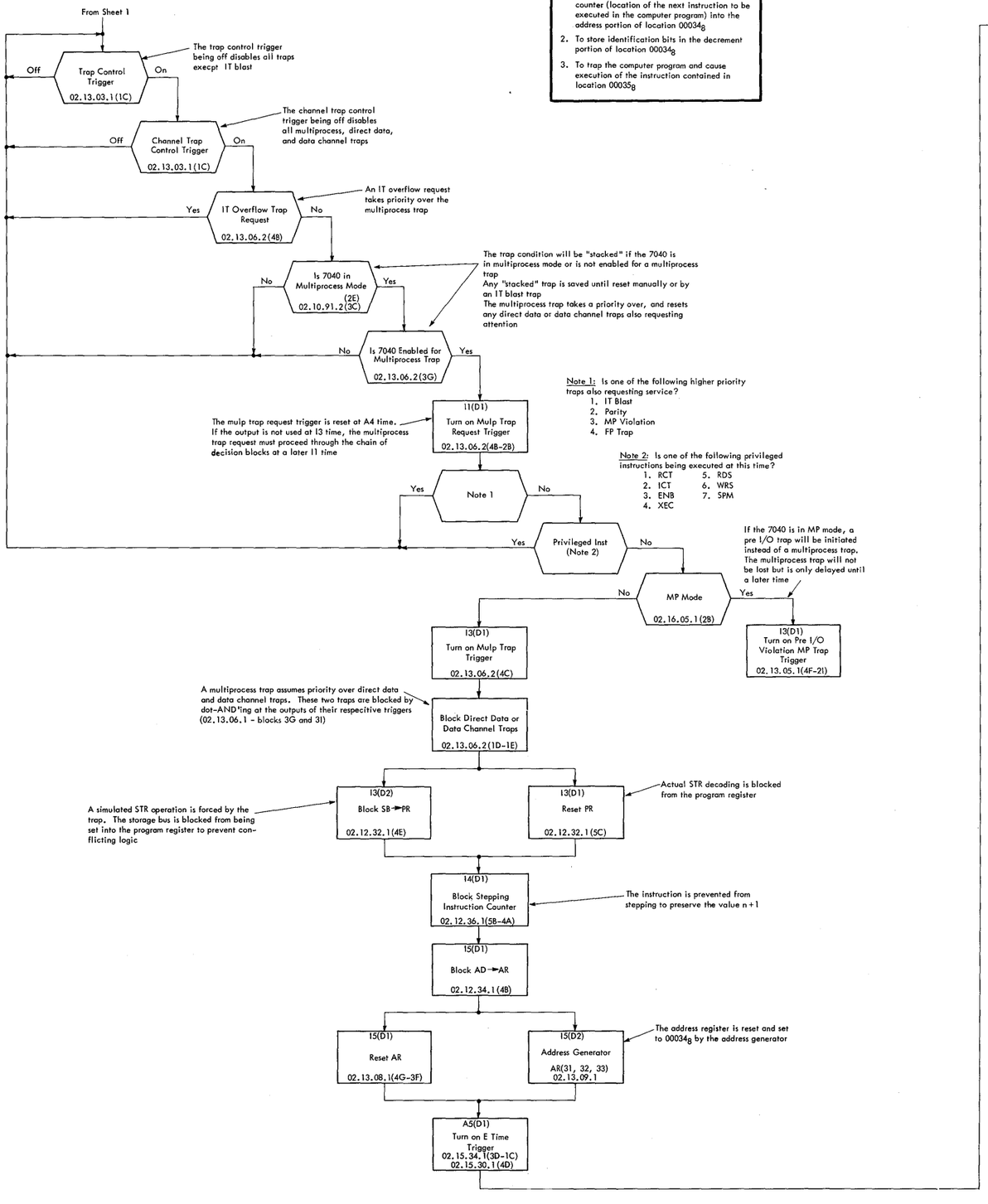


Figure 41. Trap to 7040 Flow Chart (Sheet 1 of 2)



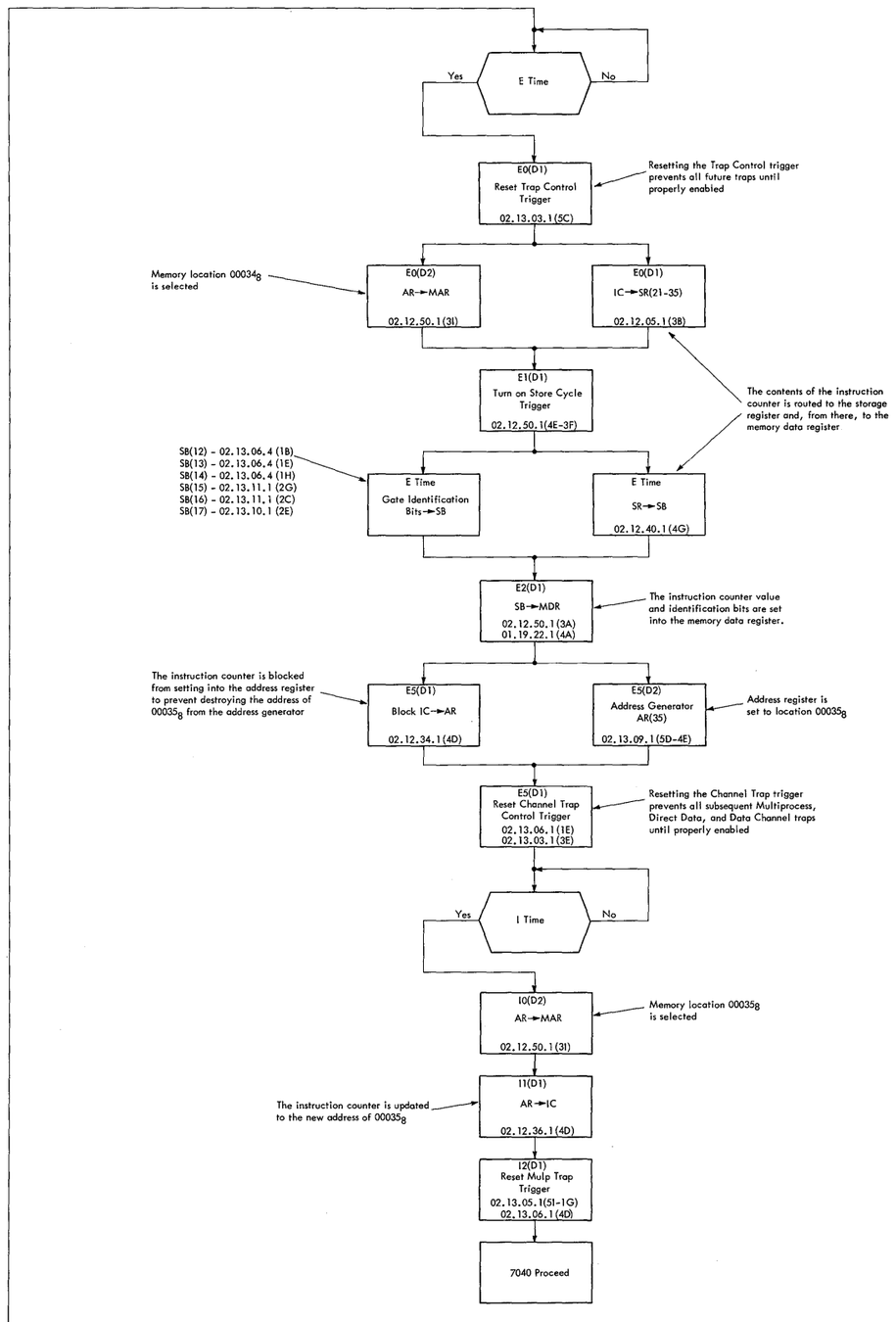


Figure 41. Trap to 7040 Flow Chart (Sheet 2 of 2)

In this section, only the differences between the 7094 and 7094 II are discussed. For the most part, the same basic concepts discussed in the previous sections also hold true for the 7094 II.

Certain timings will vary because of the difference in length of machine cycles between the two systems. The 7094 has a 2.0-microsecond, 12 cycle-point clock while the 7094 II has a 1.4-microsecond, 8 cycle-point clock. The 7094 II must also be concerned with both the 8 cycle-point CPU clock and the 12 cycle-point channel clock. To show these new timings a tape write select is taken as a typical operation and is discussed in detail, using both flow charts and timing charts.

The most pronounced area of 7094 II logic change is in setting the 7040 PEAR register. Here, both the routing paths as well as the concept are different. These areas will, therefore, be discussed in detail.

An additional signal line, mulp mode, is gated into the 7094 II from the 7040. This line is used as a conditional line when routing the program counter contents to the PEAR register. Other than this one small area, mulp mode plays no part in the internal 7094 II circuitry. As far as 7094 II operation is concerned, no distinction is made between a pure multiprocess mode, or a stand-alone condition. HIP and exempt modes, however, change the picture considerably.

A summary comparison of the two systems is given:

1. Similarities (7094 II vs 7094)
   a. Same basic logical concepts.
   b. Same basic back panel wiring for mulp, HIP, and exempt channel assignments.
2. Differences (7094 II vs 7094)
   a. CPU cycle timing and cycle length (1.4 microsecond vs 2.0 microsecond; 8 cycle-points vs 12 cycle-points).
   b. Both CPU and channel timing must be considered with the 7094 II.
   c. The setting of the PEAR register has changed in both overall concept and routing paths used.
   d. Mulp mode signal used in 7094 II logic.

7094 II Channel Operations

- **Normal channel operation occurs when in:**
  Stand-alone condition.
  Multiprocess mode.
  Exempt mode with an exempt channel.
- **A HIP halt occurs when:**
  Selecting a channel when in HIP mode but not in exempt mode.
  Selecting a non-exempt channel when in both HIP and exempt modes.
- **Physical wiring determines the multiprocess, HIP, and exempt channel assignments.**
- **Indirect addressing is performed on applicable channel instructions before a HIP halt.**
- **ENB and RCT instructions execute before a HIP halt.**

In discussing channel operations, a tape write select (WRS) will be taken as a typical example of machine timings. The WRS will be discussed as both a normal operation and one causing a HIP halt. Taking both cases will act as a refresher to recall how the normal operation is executed, and also to show what areas of the operation are affected by the HIP condition.

Channel Assignment

- **Physical wiring determines channel assignment.**
- **One channel is assigned as multiprocess channel.**
- **One or more channels can be assigned as exempt channels.**

The back panel wiring for channel assignments is the same as for the 7094. Figure 42 shows condensed logic for selecting channels A-E.

Multiprocess Channel Assignment

Using Figure 42 as an example, channel E is assumed to be the multiprocess channel. Because the majority of customer installations have a maximum of four channels, most installations have therefore chosen channel E because it represents a channel which has never been used before on their system. As a result, existing programs can be run without fear of instructions making reference to this channel.

A BTT is the only instruction which should be directed at the multiprocess channel. This instruction acts as a trap request from the 7094 II so that the 7040 system can service the needs of the 7094 II.

Multiprocess channel assignment wiring removes the normal select channel E signal from OR circuit (2G) on Systems 06.00.06.1. For a BTT(M) instruction, the I/O POD trigger is prevented from staying on and a select channel E signal is generated from a new AND circuit inserted at (2F). This L1(D2) pulse (Figure 42), when sent to the 7040, initiates a multiprocess trap request (Figure 40). Figure 30 is a summary of the BTT operation for various modes.

When in HIP mode, any instruction other than a BTT(M) directed to the multiprocess channel turns on the I/O POD trigger and causes a HIP halt. When not in HIP mode, any instruction other than a BTT(M) produces the same effect as if the channel were not attached to the system.

Exempt Channel Assignment

Again using Figure 42 as an example, channel B is assumed to be an exempt channel. Exempt channels are specified by the customer for his particular installation; back panel wiring is then added to correspond to these specified channels.

The purpose of exempt mode is to nullify the effect of HIP mode on the specified channels. Note that exempt mode affects CPU circuitry *only* when the system is also in HIP mode. The 7040 program can place the 7094 II in exempt mode and not in HIP mode, but this combination of modes has no logical significance.

Figure 42 shows the added wiring and circuitry needed for exempt channel operation. If a tape WRS instruction (+0766 . . . 2221) is taken, for example, channel B is decoded at the AND circuit (3F) on Systems 06.00.05.1. Select channel B is generated at OR circuit (2F) and is sent to the channel. The output of AND circuit (3F) has additional wiring which feeds to a DOT-OR circuit (5D/5F) on Systems 02.10.62.1. An output of this circuit AND'ed at (3G) with exempt mode causes HIP mode gated to drop. At this point, all knowledge of HIP mode is removed from the 7094 II system and the channel is allowed to complete the WRS operation.

HIP Channel Assignment

Individual channels do not require special wiring if they are to cause a HIP halt. As implied in the two previous sections, all channels are assumed to be HIP channels unless specifically wired as either multiprocess or exempt channels.

With reference to Figure 42, consider channel A as a HIP channel. A tape WRS(+0766 . . . 1221) instruction causes a momentary select channel A signal to be produced by the AND(3B-3C) and OR(2D) on Systems 06.00.05.1. However, at the same time, SOD decoding from the program register activates "any I/O select" which turns on the I/O POD trigger (12.10.61.1, 4D). The I/O POD trigger turning on, drops POD 76 (Systems 08.00.33.1, 2C) and, in turn, drops select channel A at AND circuit (3B-3C) on Systems 06.00.05.1. The partial select channel A signal which does get to the channel is of too short a duration to perform any actual selection.

Tape Write Select (Exempt Mode)

- **Operation is identical to stand-alone.**
- **HIP mode is degated.**
- **HIP halt is suppressed.**
- **The PC-1 is not gated to the 7040 PEAR register.**

Execution of a tape WRS (or any other channel instruction) to an exempt channel while in exempt mode is identical to a normal stand-alone operation. This first section explains the normal operation. The following section explains the HIP operation.

Assume for this operation that channel B is designated as an exempt channel and that the system is in multiprocess, HIP and exempt modes. Tape unit 1 is to be selected on channel B. The basic objectives of the WRS operation are:

1. Load the instruction into the program register.
2. Perform address modification, if specified, and set the shift counter.
3. Decode the instruction:
   a. Primary operation from PR(1-5).
   b. Secondary operation from PR(6-9).
   c. Channel selection from SR(23-26).
   d. Class address from SC(10-13).
   e. Unit address from SC(14-17).
4. Prevent a HIP halt from occurring.
5. Machine cycles:
   a. CPU goes into L time.
   b. Turn on channel L time trigger.
6. Set the class and unit address registers in the data channel.
7. End operation on signal from the channel.
8. Proceed to the next instruction.

Figure 43 is an overall flow chart of a 7094 II tape write select operation; Figure 44 is a timing chart showing a normal non-HIP operation (exempt channel or stand-alone).

At I4 time, the instruction is loaded into both the storage register and program register. Address modification is permitted on instructions of this type and programmers quite often take advantage of this feature to modify the address of the selected unit.

Address modification is accomplished at I5(D1) time by gating the specified index register to the index adders together with SR(21-35). At I5 CP set time, this modified address is routed to AR(3-17) and SC(10-17). Note that address modification only affects the class and unit address decoding from the shift counter. Channel selection is determined from the storage register and is, therefore, not subject to address modification by the index register.

At the end of I time, four signal lines are sent to the channels. These signals and their source of decoding are:

SIGNAL	SOURCE OF DECODING
Select Channel B	SB(23-26)
Secondary Operation WRS	PR(S, 1-9)
Tape Class Address	SC(10-13)
Unit Address	SC(14-17)

Of these four signals, the last three are unconditionally sent to all channels. The first signal, select channel B is decoded and sent exclusively to channel B.

While operating with exempt channel B, the 7094 II must not HIP halt. Additional circuit wiring (Figure 42) detects an exempt channel selection and causes HIP mode gated to be blocked (Systems 12.10.62.1). Without HIP mode gated, the 7094 II acts and operates like a normal stand-alone computer.

At the end of I time (Figure 44), the channel L time trigger is turned on to provide a 12 cycle-point clock output to the channel circuitry. The 7094 II goes into L time to wait until an I/O end-op signal is returned from the channel.

In the data channel, a test is made to determine if an operation is already in process. If the channel is busy, the CPU is held in both L time and channel L time until the present operation is finished. When finished, the WRS signals are accepted into the channel registers and an I/O end-op signal is returned to the 7094 II. This I/O end-op signal turns on the CPU end-op trigger and allows the program to continue.

During the WRS operation, the CPU can be held in a variable number of consecutive L times depending on both the status of the channel and the alignment of the CPU and channel clocks. In Figure 44, three CPU L cycles are required to synchronize the operation and properly select the data channel.

Tape Write Select (HIP Mode)

- **Channel is not selected.**
- **Channel L time is blocked.**
- **The PC-1 is gated to the 7040 PEAR register.**
- **An HPR is forced into the program register for a HIP halt.**
- **Trap identification bit signals are sent to the 7040.**

The previous section explained the exempt or stand-alone tape write select operation. In this section, assume that channel B is again designated as an exempt channel, but that the system is only in multiprocess and HIP modes. The basic objectives of this WRS are:

1. Load the instruction into the program register.
2. Perform address modification, if specified, and set the address register and shift counter.
3. Decode the instruction as for a normal operation.
4. Prevent selection of the specified channel.
5. Turn on the I/O POD trigger.
6. Decrement the program counter (−1) and set into SR(21-35).
7. Gate the storage register onto the storage bus and into the 7040 PEAR register.
8. Block channel L time.
9. Force a channel end-op to the CPU circuits.
10. Force an HPR into the program register.
11. Halt the 7094 II.
12. Send trap identification bits to the 7040.

Figure 43 is a tape write select flow chart for a HIP halt condition; Figure 45 is a timing chart of the HIP halt.

The initial loading of the storage register and program register, and address modification of SR(21-35) is the same as discussed in the previous section. The same four channel signal lines are also generated during I time. The last three signals are, again, unconditionally sent to all channels throughout the entire WRS operation. Actual channel selection is blocked, however, by blocking select channel B.

As shown in Figure 42, select channel B becomes active at I4 time. Note that HIP mode gated can not be blocked (Systems 12.10.62.1,2G) because exempt mode is not present (Systems 12.10.62.1,3G). With HIP mode gated active, the I/O POD trigger is turned on at L1(D1) time. The I/O POD trigger coming on degates "POD 76" (Systems 08.00.33.1,2C) and drops the select channel B signal to the channel. Even though this latter signal is active to the channel for a short period of time, it will not be sampled. The I/O POD trigger prevents the channel L time trigger from being turned on so that "L time" is not available at the channel circuits.

Turning on the I/O POD trigger at L1(D1) time changes the whole course of events from a normal channel select to a HIP halt. As mentioned in the previous paragraphs, both channel selection and channel L time are blocked. Because of the HIP halt, the 7040 PEAR register must be set with the location of the instruction causing the halt.

At the time that the WRS instruction is loaded into the program register, the program counter has already been stepped (+1). During L time, under control of the I/O POD trigger, the program counter is gated to the index adders together with all 1's to XAD(3-17) as a means of decrementing the program counter value (−1). The decremented value is then set into SR(21-35) as a routing path to the MDBI and eventually into the PEAR register (Figure 43). The storage register is gated to the 7040 and is set into the PEAR register at I0(D1) time of the upcoming forced HPR. This PEAR timing is asynchronous with the 7040 operation.

At I6(D1) time of the forced HPR, the master stop trigger is turned on to halt the 7094 II, and trap identification bit signals are sent to the 7040. These signals set

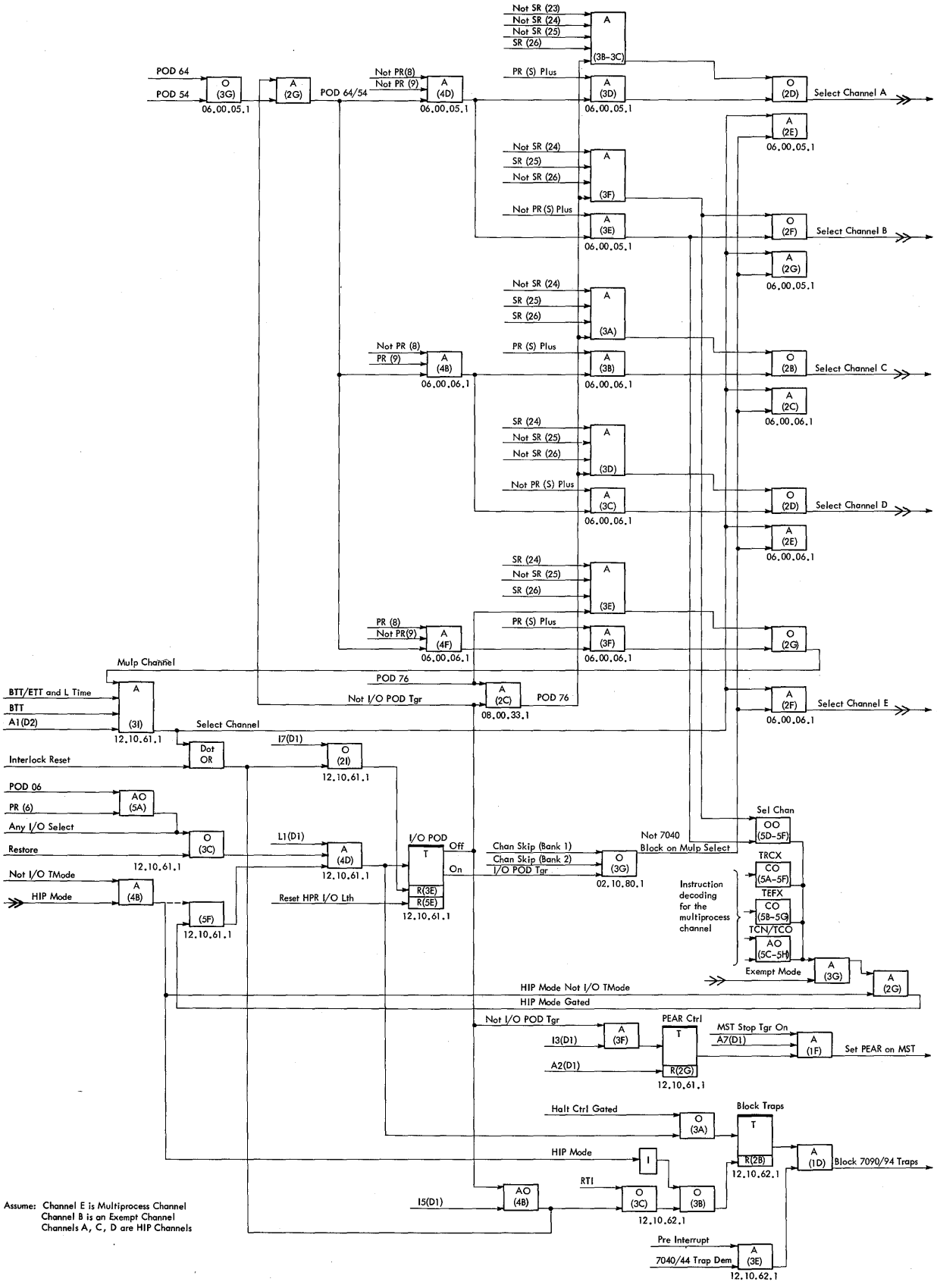
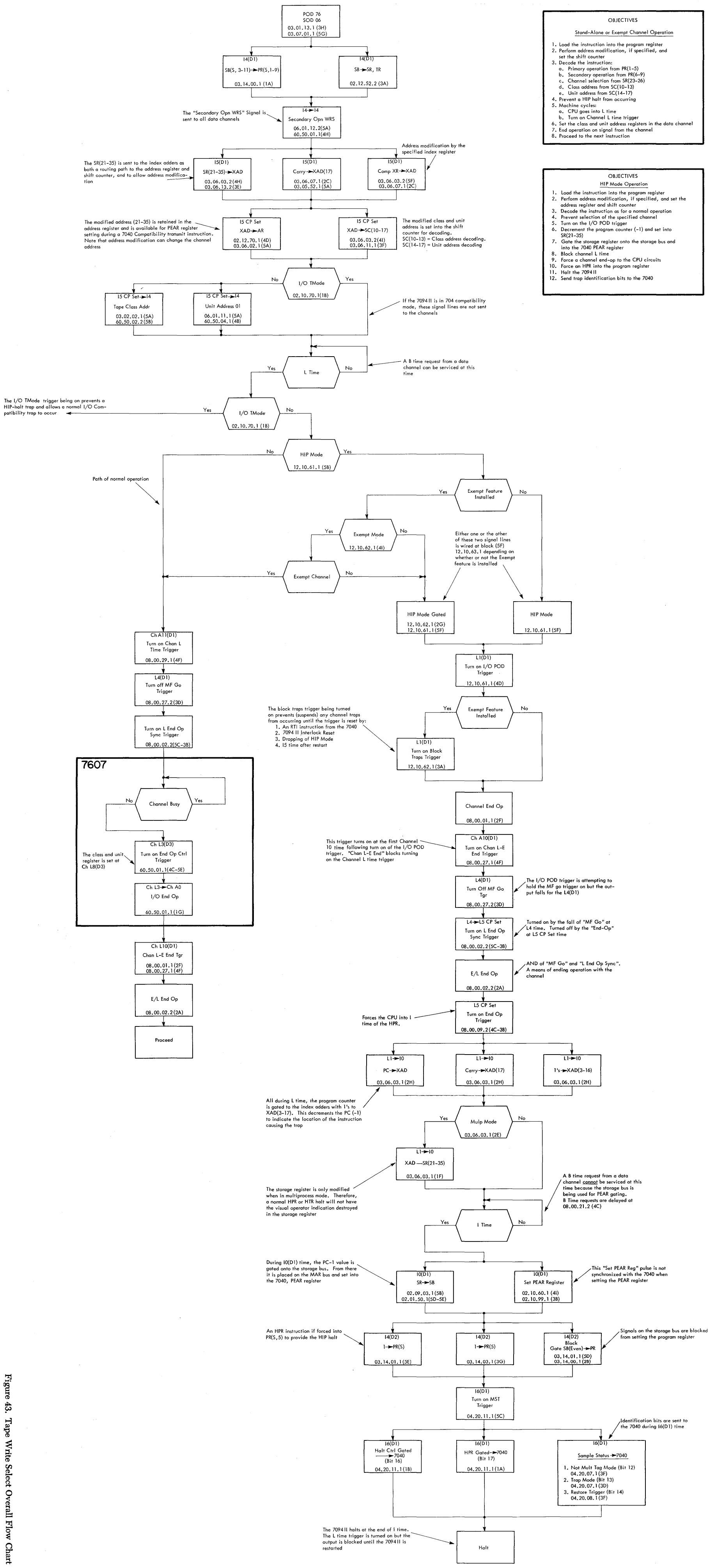


Figure 42. HIP and Exempt Channel Condensed Logic



OBJECTIVES

Stand-Alone or Exempt Channel Operation

1. Load the instruction into the program register
2. Perform address modification, if specified, and set the shift counter
3. Decode the instruction:
   a. Primary operation from PR(1-5)
   b. Secondary operation from PR(6-9)
   c. Channel selection from SR(23-26)
   d. Class address from SC(10-13)
   e. Unit address from SC(14-17)
4. Prevent a HIP halt from occurring
5. Machine cycles:
   a. CPU goes into L time
   b. Turn on Channel L time trigger
6. Set the class and unit address registers in the data channel
7. End operation on signal from the channel
8. Proceed to the next instruction

OBJECTIVES

HIP Mode Operation

1. Load the instruction into the program register
2. Perform address modification, if specified, and set the address register and shift counter
3. Decode the instruction as for a normal operation
4. Prevent selection of the specified channel
5. Turn on the I/O POD trigger
6. Decrement the program counter (-1) and set into SR(21-35)
7. Gate the storage register onto the storage bus and into the 7040 PEAR register
8. Block channel L time
9. Force a channel end-op to the CPU circuits
10. Force an HPR into the program register
11. Halt the 7094 II
12. Send trap identification bits to the 7040

Figure 43. Tape Write Select Overall Flow Chart

corresponding triggers in the 7040 and, if enabled for multiprocess traps, cause a trap to location 00035_8.

The 7094 II halts at the end of I time with the L time trigger on, and with SR(21-35) containing the value set into the PEAR register. The storage register retains this value until the 7094 II is restarted or until a compatibility transmit is executed by the 7040.

The block traps trigger is also turned on with the same set pulse which turned on the I/O POD trigger. Therefore, all 7094 II data channel or special feature traps are suspended until the 7040 issues a reset trap inhibit (RTI) or start remote computer (SRC) instruction.

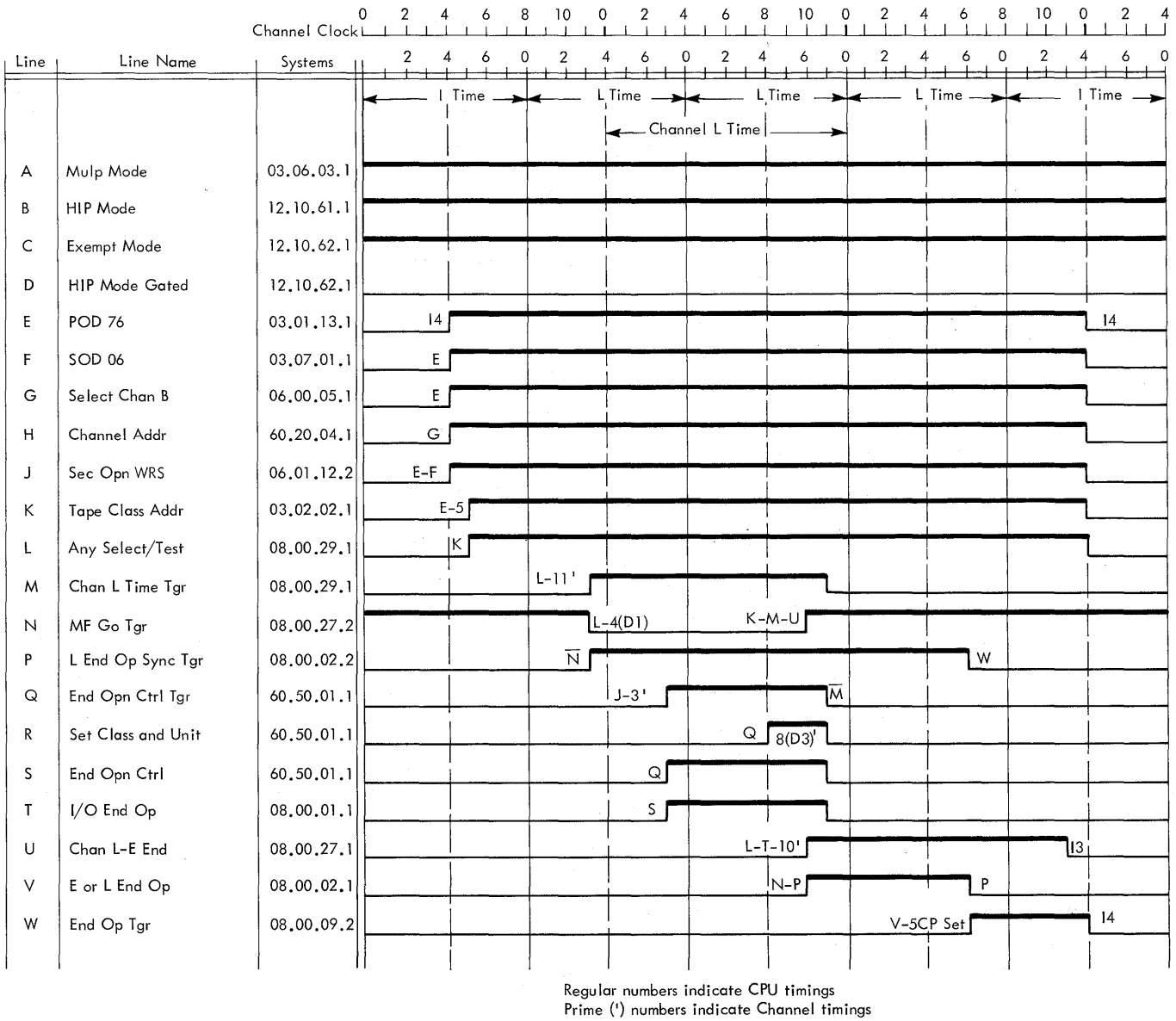


Figure 44. Tape Write Select Timing Chart (Exempt Channel)

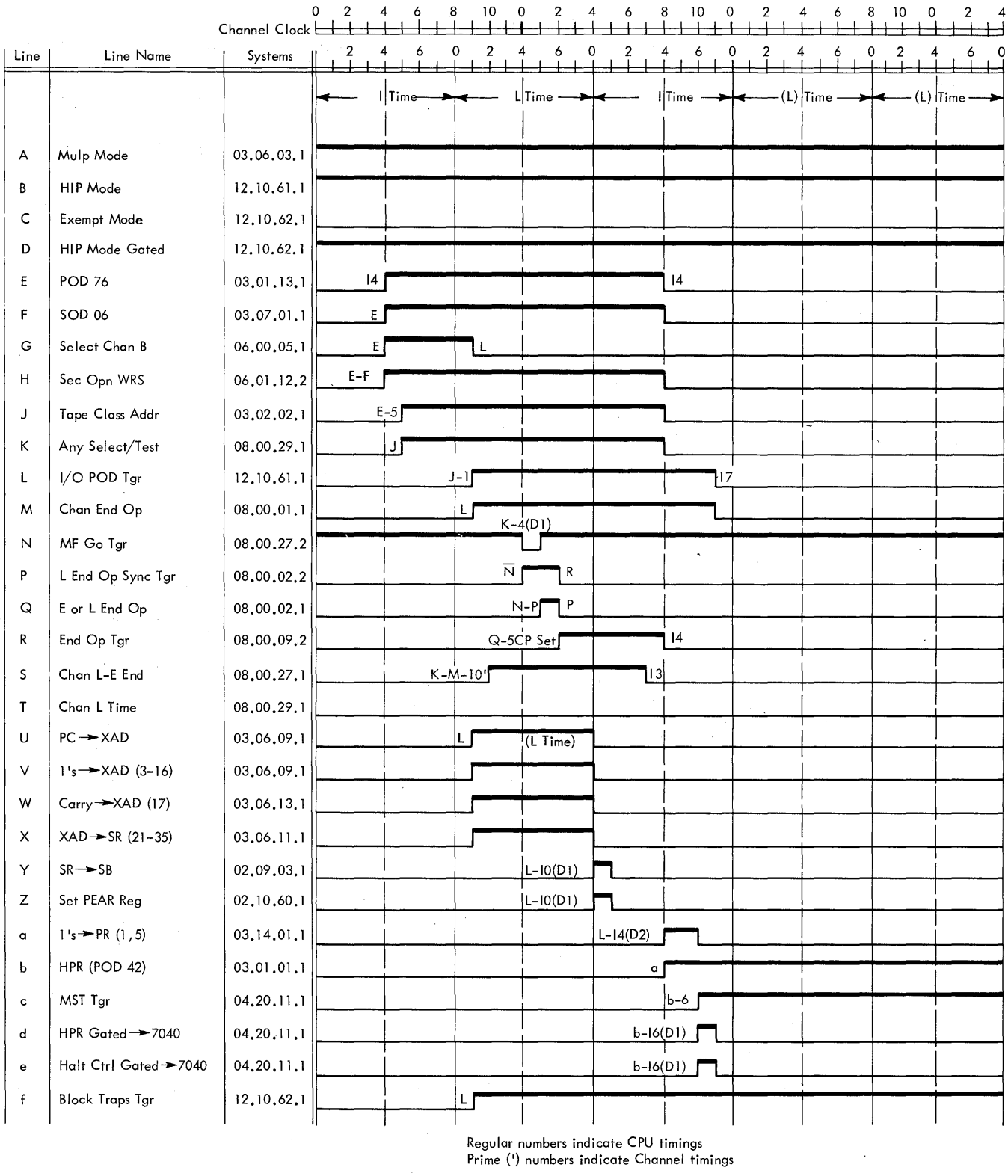


Figure 45. Tape Write Select Timing Chart (HIP Channel)

Setting the 7040 PEAR Register

- **The PEAR register is not set for each 7094 II instruction executed.**
- **PEAR register is only set for:**
  Each 7094 II I/O instruction causing a HIP halt.
  Each normal 7094 II halt.
  Each 7094 II execute (XEC) instruction.
  Each 7040 compatibility transmit (TMT) instruction.
- **The program counter is decremented (−1) when sent to the PEAR register.**
- **The address register is sent unmodified to the PEAR register.**

During a 7094 operation (as explained earlier in this manual) the PEAR register was set for each instruction executed and was not dependent on whether the instruction was actually a channel instruction (Figure 14). This method of setting was used because the correct instruction location value was always available at a certain time and could be gated conveniently onto the channel MAR bus.

Due to the nature of instruction overlapping on the 7094 II, however, the correct value is not always available for convenient gating to the 7040 PEAR register. Because of this, the PEAR register is only set when actually needed.

Figure 46 shows condensed logic of PEAR register controls. Note the similarity of the 7040 area with Figure 14.

HIP Halt

- **The 7040 PEAR register is set during I0(D1) of the forced HPR I cycle.**
- **The 7094 II program counter is decremented (−1) when set into the PEAR register.**
- **The 7040 PEAR register indicates location of instruction causing trap.**
- **The 7094 II program counter indicates next sequential instruction.**
- **A 7040 multiprocess trap is initiated by the 7094 II HIP halt.**

The HIP halt operation was discussed under channel operations and illustrated with a flow chart and timing chart in Figures 43, 45. As a result of HIP mode, the channel instruction is not executed. Instead, the 7094 II forces an HPR into the program register and halts.

Two important facts should be remembered about the values of the PEAR register and program counter after the HIP halt:

1. The 7040 PEAR register contains the location of the 7094 II I/O instruction which caused the HIP halt. This assures the 7040 program the capability of obtaining the I/O instruction during a subsequent compatibility transmit.

2. The 7094 II program counter contains the location of the next sequential instruction. This location is by definition for an HPR instruction.

Both of these conditions are identical for either the 7094 or 7094 II system. Timings differ between the two systems but the end result is the same. This fact is necessary to maintain programming compatibility.

Normal HPR Halt

- **The 7040 PEAR register is set during I7(D1) time of the HPR instruction.**
- **The 7094 II program counter is decremented (−1) when set into the PEAR register.**
- **The 7040 PEAR register indicates location of HPR instruction.**
- **The 7094 II program counter indicates next sequential instruction.**
- **A 7040 multiprocess trap is initiated by a 7094 II HPR halt.**

The PEAR register must be set on a normal HPR halt because of the corresponding multiprocess trap which is initiated in the 7040. Remember that this is not a HIP halt; there is no I/O instruction involved.

Because the I/O POD trigger is not turned on in this case, other means must be used to gate the proper program counter value to the PEAR register. A PEAR control trigger is turned on every I3 time when the I/O POD trigger is not on (Figure 42). This PEAR control trigger actually represents a period of I time not resulting from a HIP halt. When the master stop trigger (MST) turns on at I6 time, the PEAR control trigger allows an I7(D1) pulse to send a set PEAR register signal to the 7040. Figure 47 is a timing chart showing a normal HPR operation.

The value actually gated to the PEAR register is formed by gating the program counter to the index adder with all 1's to XAD(3-17) as a decrementing (−1) function. The index adders are then gated to SR(21-35) and eventually to the channel MAR bus through multiplexor circuitry (Figure 46). Two items should be remembered during this operation:

1. The program counter is not affected while decrementing the contents to the storage register and PEAR register. The program counter, therefore, retains the location of the next sequential instruction.

2. The storage register is modified only if the system is in multiprocess mode. If this were not done, the visual indication in the storage register would not be a correct one for operator usage in a stand-alone situation.

Trap identification bits are sent to the 7040 as a result of the HPR halt, and a multiprocess trap is initiated. Refer to Figures 38, 39, 40, 41.

Even though there is no HIP halt involved in this operation, the block traps trigger is turned on to suspend any subsequent 7094 II data channel or special feature traps (Figure 42). This blocking is necessary to preserve the 7094 II program counter setting until the 7040 has had time to analyze the reason for the multiprocess trap.

Normal HTR Halt

- **The 7040 PEAR register is set during I7(D1) time of the HTR instruction.**
- **The 7094 II program counter is decremented (−1) when set into the PEAR register.**
- **The 7040 PEAR register indicates location of HTR instruction.**
- **The 7094 II program counter is decremented (−1) to indicate location of HTR instruction.**
- **A 7040 multiprocess trap is initiated by a 7094 II HTR halt.**

As explained in the previous section, the PEAR register must be set on an HTR halt because of the multiprocess trap initiated to the 7040. Again, there is no HIP halt involved in this operation.

The PEAR control trigger is turned on at I3 time of the HTR instruction (Figure 42). When the master stop trigger (MST) turns on at I6 time, the PEAR control trigger allows an I7(D1) pulse to send a set PEAR register signal to the 7040. Figure 48 is a timing chart for a normal HTR operation.

The value gated to the PEAR register is formed by gating the program counter to the index adder with all 1's to XAD(3-17) as a decrementing (−1) function. Remember that the program counter had already been unconditionally stepped (+1) before the HTR was decoded from the program register. The decremented value sent to the PEAR register, therefore, indicates the location of the HTR instruction.

Note that as a result of this HTR instruction, the decremented value from the index adders is also returned to the program counter. Thus, the 7094 II halts with both the program counter and SR(21-35) indicating the same value.

The storage register is modified only if the system is in multiprocess mode. This restriction is imposed for the same reason as explained previously for a normal HPR; the visual indication in SR(21-35) must be retained for operator usage in a stand-alone situation.

Trap identification bits are sent to the 7040 as a result of the HTR halt and a multiprocess trap is initiated. Refer to Figures 38, 39, 40, 41.

Even though there is no HIP halt involved in this operation, the block traps trigger is turned on to suspend any subsequent 7094 II data channel or special feature traps (Figure 42). The blocking, as explained in the previous section for an HPR, is necessary to preserve the 7094 II program counter setting until the 7040 has had time to analyze the reason for the multiprocess trap.

Execute (XEC) to an HTR/HPR/I/O Instruction

- **The 7040 PEAR register is set from the address register during I1(D2) of the XEC instruction.**
- **The PEAR register indicates the location of instruction referenced by the XEC instruction.**
- **The PEAR register is not set from the program counter due to an executed (XEC) halt or forced HIP halt.**
- **A 7040 multiprocess trap is initiated by the 7094 II halt.**

The execute instruction has not been altered in its basic operation. Changes during the execute operation include additional circuitry to route the unmodified address register to the 7040 PEAR register. Note that this PEAR register setting is unconditional and is not dependent on whether the instruction to be executed is an I/O POD or halt instruction.

Setting the PEAR register because of an XEC to a CLA instruction, for example, causes no problem in the 7040; the 7040 program does not use the PEAR value until trapped by the 7094 II. At the time of the trap, the PEAR register will already have been reloaded with a new value.

As an example, take an XEC to a tape write select as discussed earlier. Figures 43 and 45 show the HIP halt due to direct execution; Figure 49 is a timing chart of the executed WRS operation.

During I time of the execute instruction, the location of the instruction to be executed is formed in the address register. This final value is subject to both indirect addressing and tagging before being gated from the address register to MAR. At I0(D3) time (actually the second I time of the XEC operation) the address register is gated to the 7040 and is set into the PEAR register as an indication of the WRS instruction. This gating is accomplished by forcing a proceed to E line used for POD 54 instructions (Figure 46). The 7094 II does not go into an E cycle; the "proceed to E" is only utilized to perform the required functions.

Remember that the program counter has no meaning as far as the WRS instruction is concerned. For this reason the normal PC −1 setting of the PEAR register must be blocked at the time of the HIP halt. This blocking is accomplished by means of an XEC trigger added to Systems 02.10.60.2. This trigger is turned on during the initial I time of the XEC instruction and remains on

until I4 time of the forced HPR. The trigger output is then used to degate the normal IO(D1) set PEAR register signal (Figure 45). Note that the actual gating of the program counter to the input of the PEAR register is not blocked – only the PEAR set pulse is blocked.

At the end of the execute HIP halt the following conditions exist:

1. The program counter contains the location of the XEC +1.
2. The address register contains the modified address portion of the I/O instruction.
3. The PEAR register contains the address of the I/O instruction being executed.
4. The storage register (21-35) contains the effective address (21-35) of the XEC instruction.

An execute instruction can also refer to a normal HPR or HTR. In either case, the PEAR setting is as explained above; i.e., the PEAR register is set during the XEC and is blocked during the I time of the HPR or HTR.

7040 Compatibility Transmit

- **The 7040 compatibility transmit instruction should be executed only when the 7094 II is halted.**
- **The 7094 II address register is gated to the 7040 PEAR register during the "dummy" 7094 II channel B cycle.**

The overall objectives of the 7040-7094 II compatibility transmit are identical to those of the 7040-7094 compatibility transmit discussed previously in this manual.

Figure 50 shows a complete compatibility transmit operation between the 7044 and 7094 II systems. All other timing charts in this manual have been drawn with reference to a 7040 operation; Figure 50 now shows an operation based on the 2.0 microsecond 7044 machine cycle.

The need for the 7040 compatibility transmit instruction and how it is used were discussed previously in this manual. In this section we are only concerned with setting the PEAR register from the 7094 II address register during the second half of the compatibility transmit operation.

The second half of the operation is indicated by the 7040 shift counter stepping from 2 to 1. When SC(17) equals 1, a "4X 19 bit" signal is sent to the 7094 II.

Also, during the second half of the operation, a "dummy" 7094 II B cycle is requested. This B cycle is not needed for data transfer between the two systems but occurs to perform logical functions in the 7040. The output of the 7090 B cycle trigger in the 7040 (labeled mulp channel in use) is AND'ed in the 7094 II with "4X 19 bit" to provide a 2.10 microsecond gate for setting the 7040 PEAR register (Figure 46).

The 7040 PEAR register is set from the 7094 II address register. To accomplish this gating, an AR TO MDBI signal is produced at OR circuit (2H-2I), Figure 46. This signal in turn activates existing 7094 II proceed to E circuitry which:

1. Gates the address register to the index adders (Systems 03.06.06.2,5C).
2. Gates the index adders to the storage register (Systems 03.06.11.1,3D).
3. Gates the storage register to the MDBI (Systems 02.09.03.1,4I).

Remember that the 7094 II does not go into an E cycle; the proceed to E circuitry which exists for POD 54 instructions is used only as a convenient means of performing the required logic.

During the same period of time that the address register is being gated to the PEAR register, a set PEAR register signal is sent to the 7040 (Figure 46).

After setting the PEAR register, the 7040 proceeds to an E cycle to store the PEAR register contents, and ends operation (Figure 50).

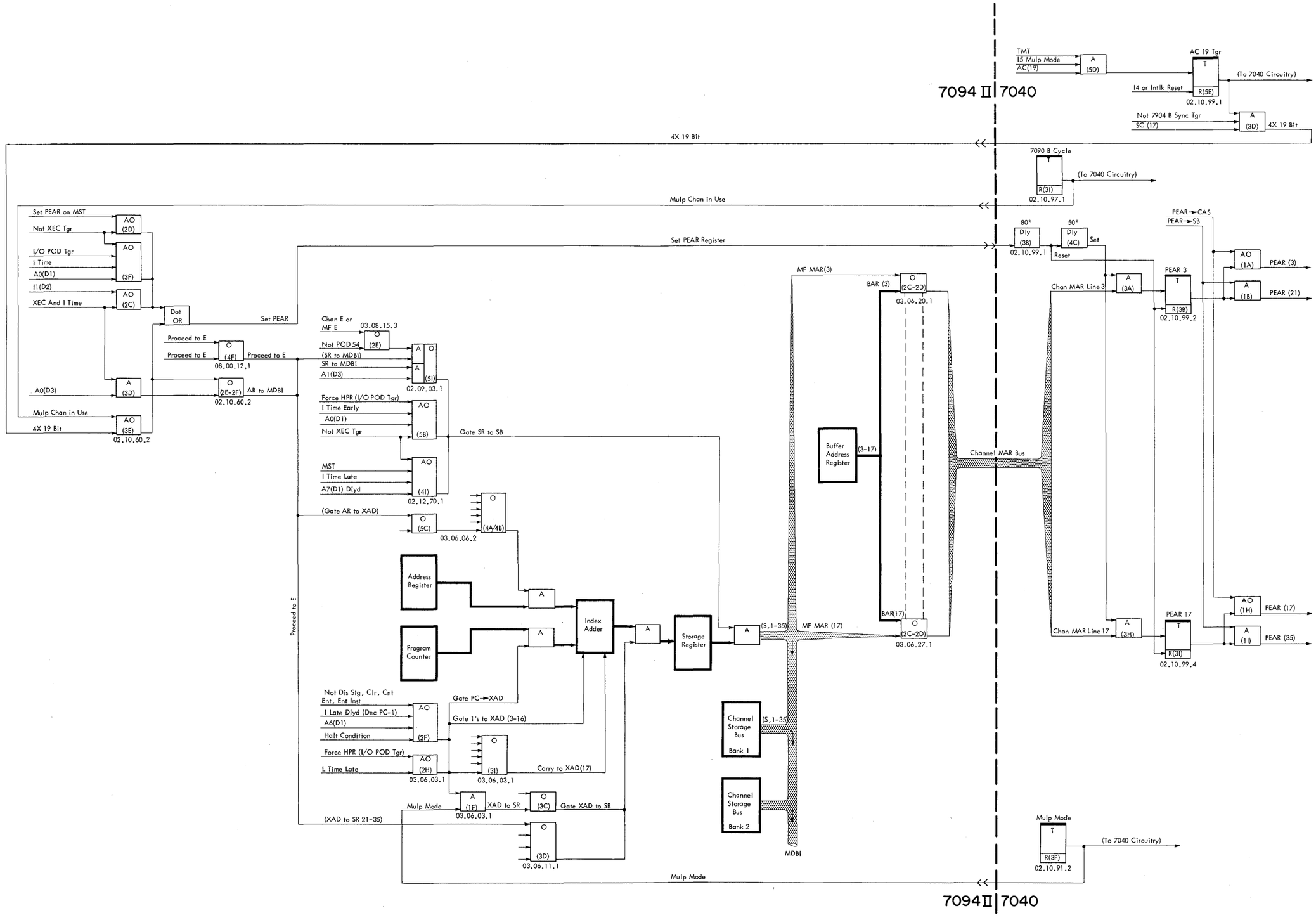


Figure 46. PEAR Register Gating Condensed Logic

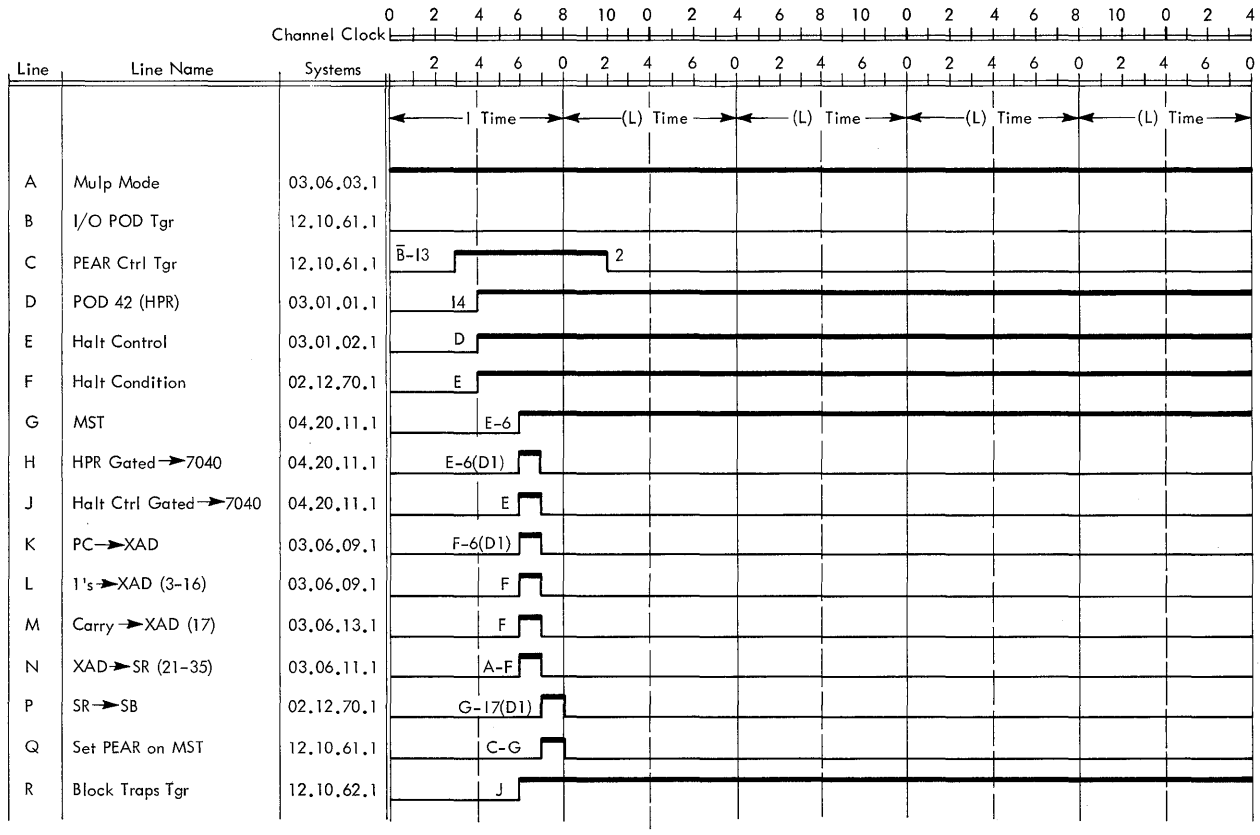


Figure 47. Normal HPR Timing Chart

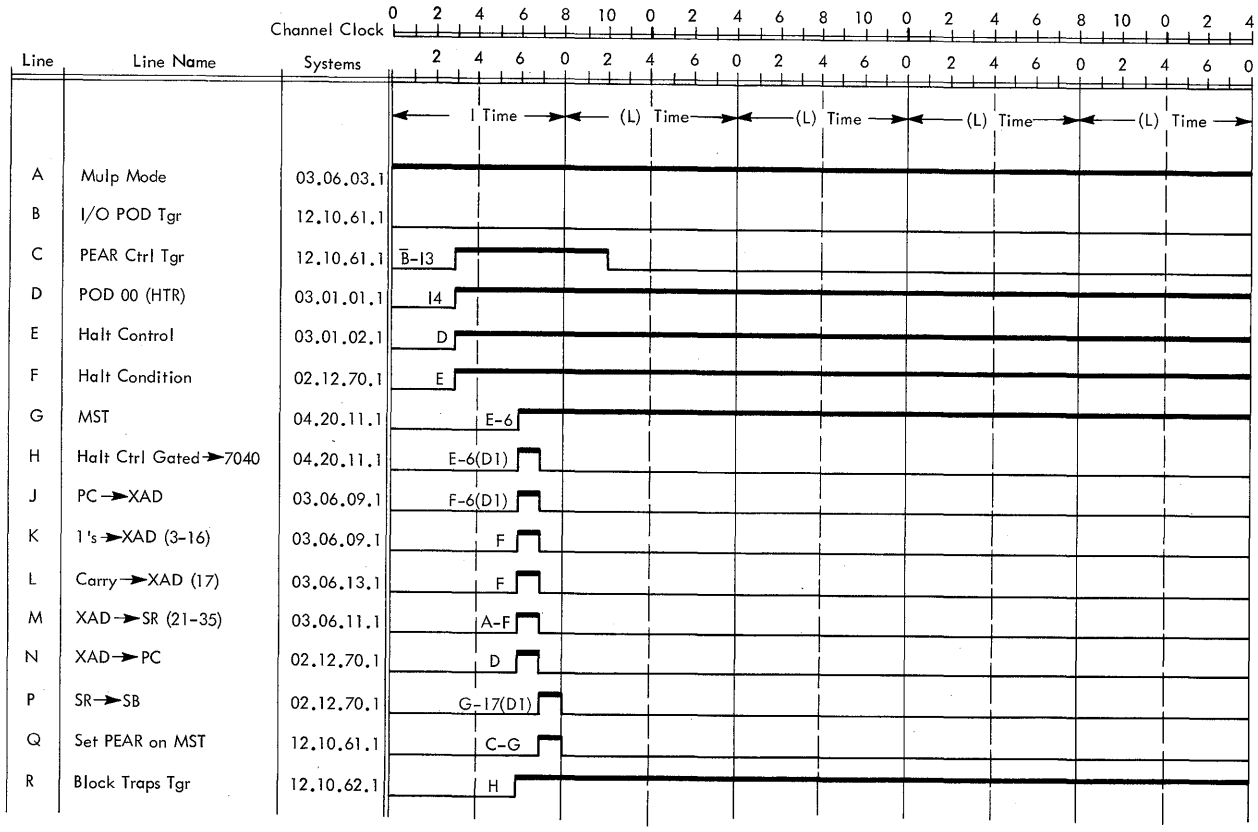


Figure 48. Normal HTR Timing Chart

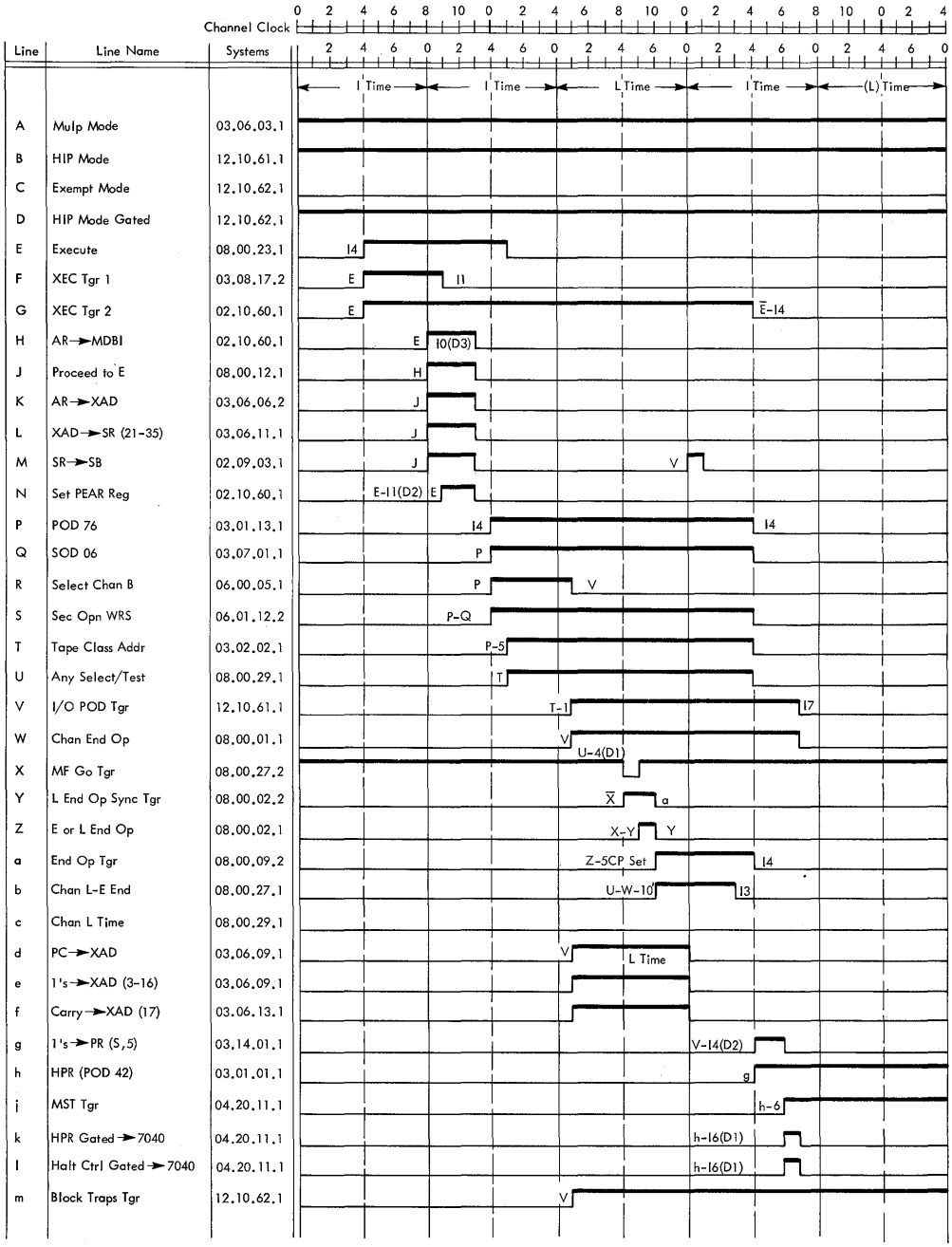


Figure 49. Execute (XEC) of a Tape WRS Timing Chart (HIP Channel)

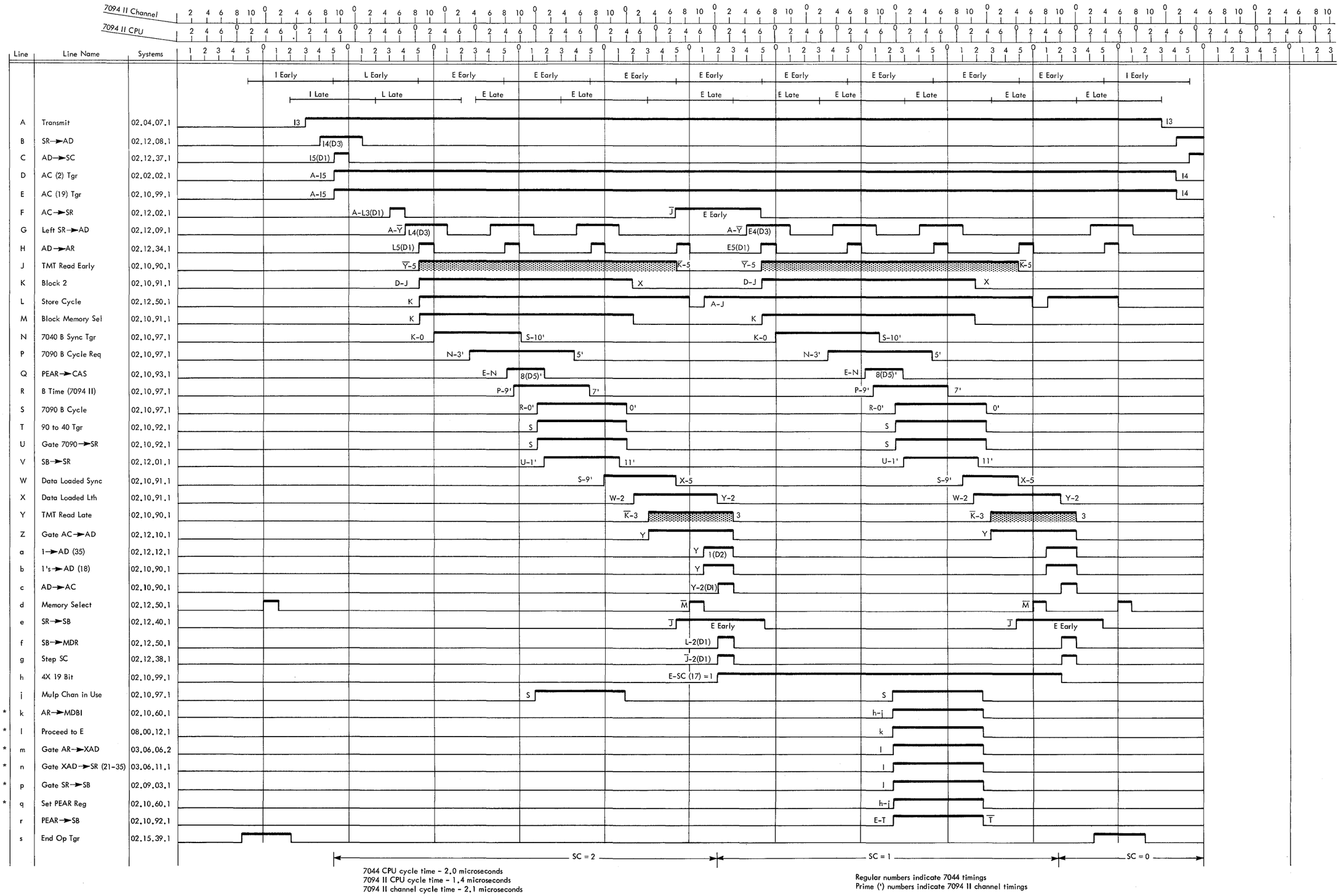
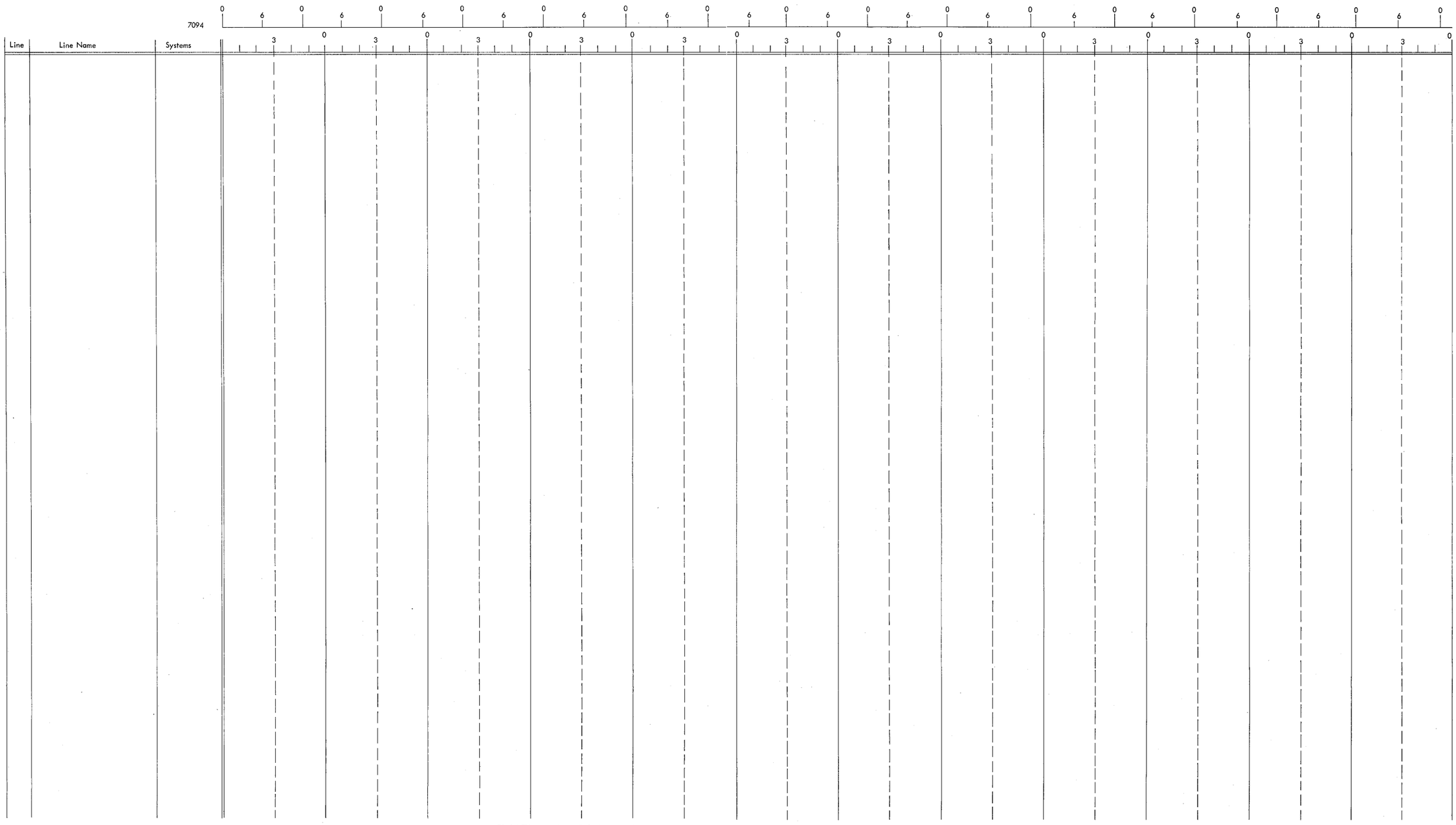


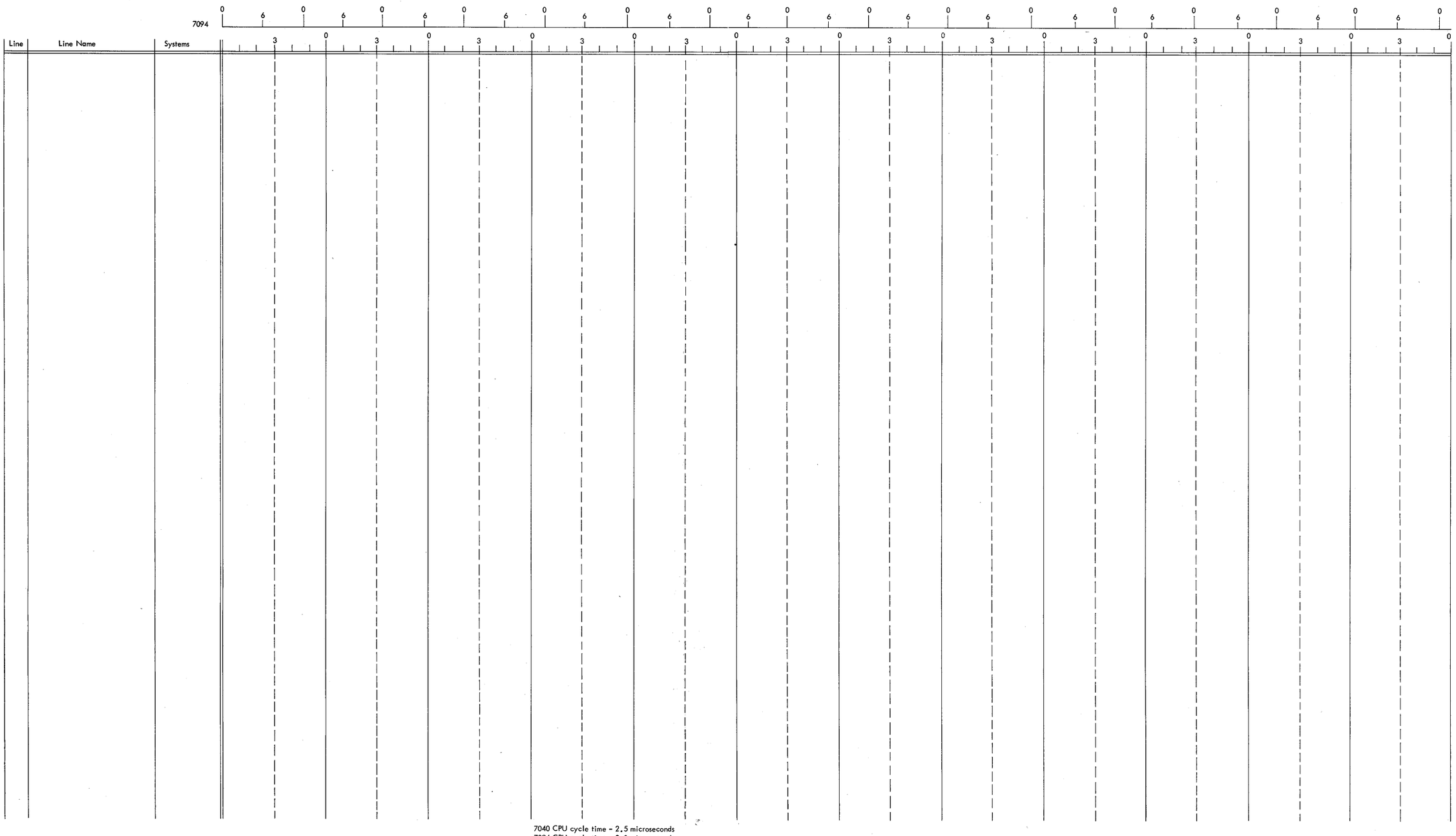
Figure 50. 7044 to 7094-II Compatibility Transmit Timing Chart

Appendix A: 7040 to 7094 Timing Chart Work Sheets



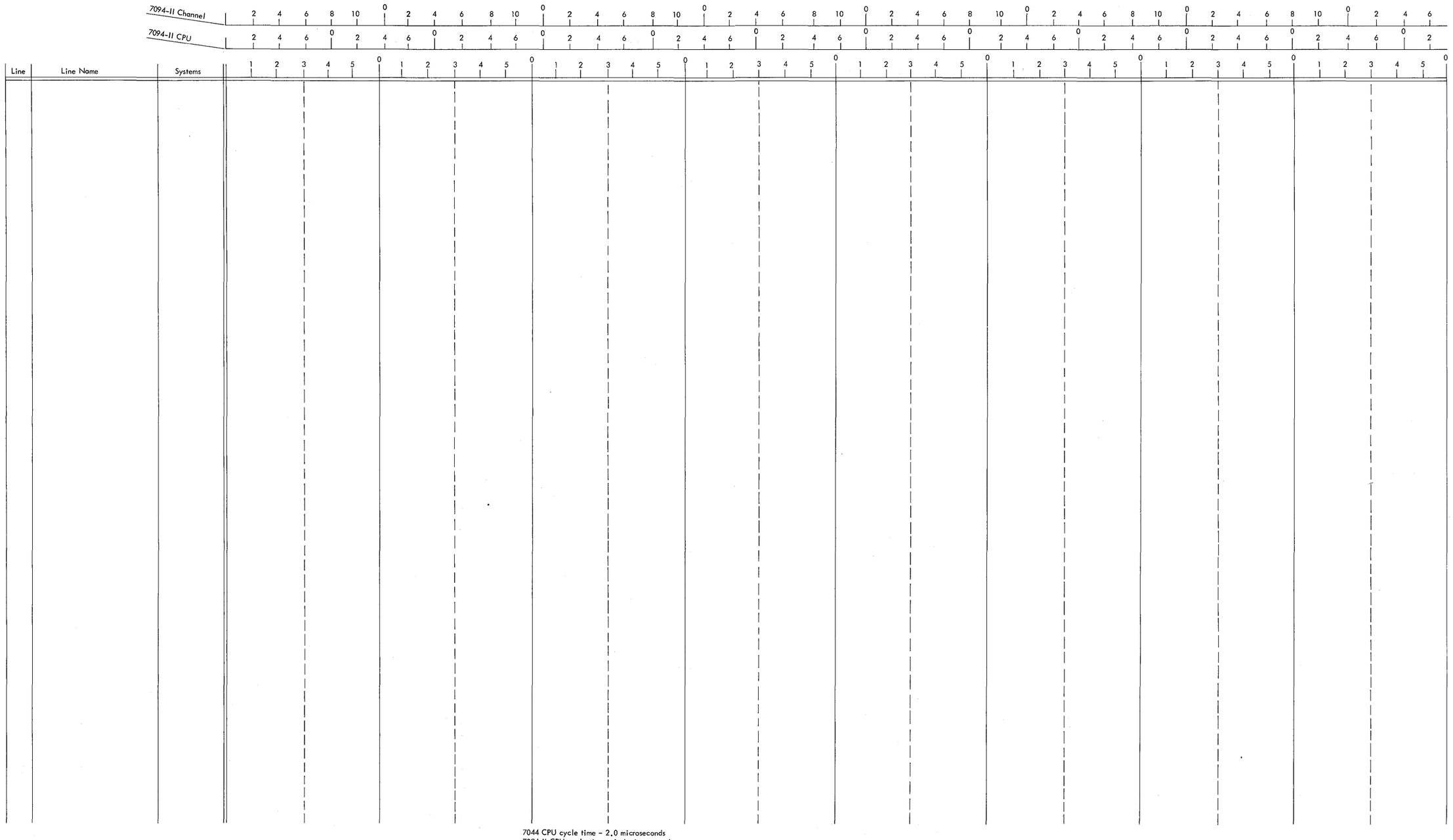
7040 to 7094 Timing Chart (Work Sheet 1)

Appendix A: 7040 to 7094 Timing Chart Work Sheets



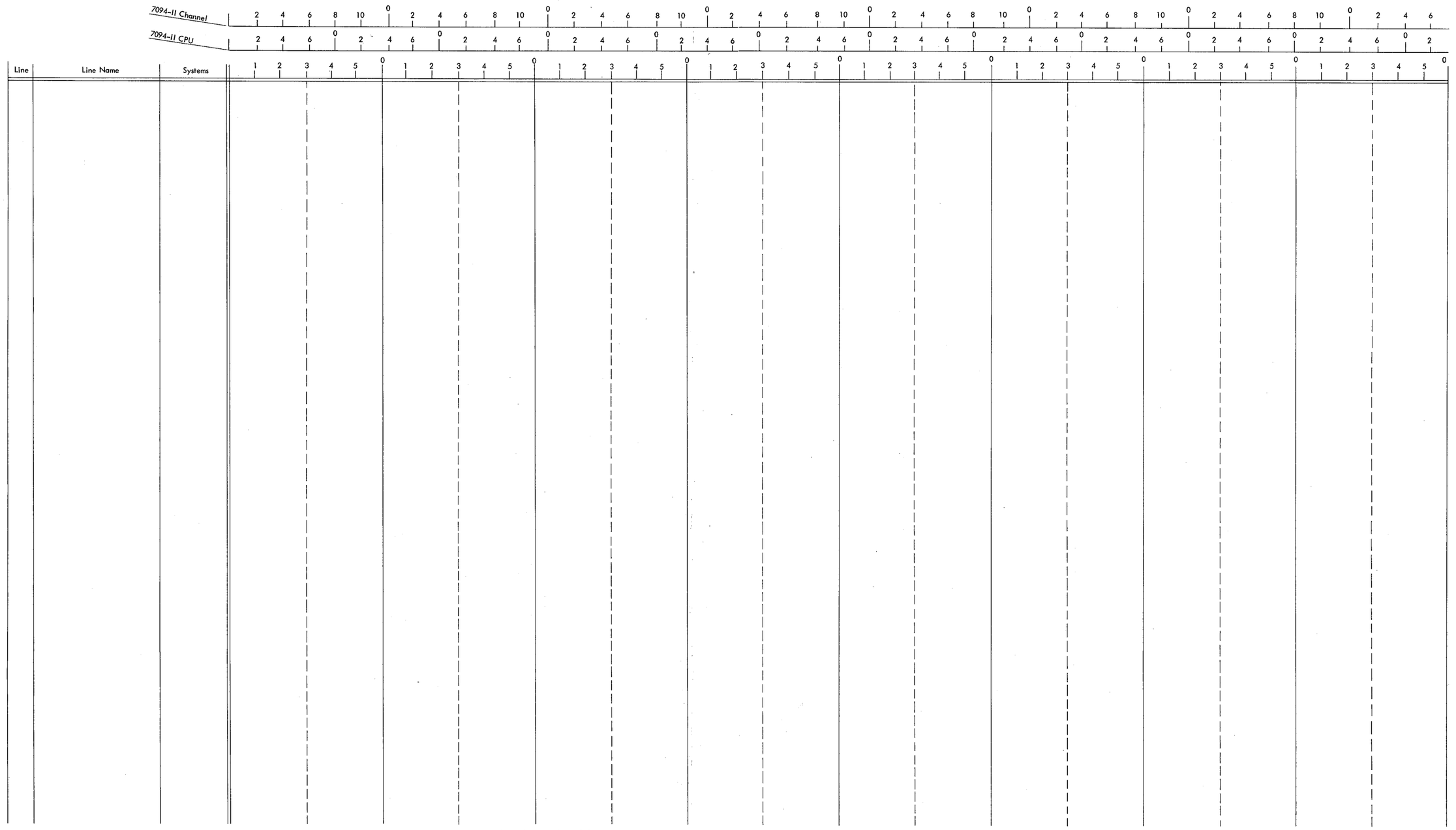
7040 to 7094 Timing Chart (Work Sheet 2)

Appendix B: 7044 to 7094 II Timing Chart Work Sheets



7044 to 7094 II Timing Chart (Work Sheet 1)

Appendix B: 7044 to 7094 II Timing Chart Work Sheets



7044 CPU cycle time - 2.0 microseconds
7094 II CPU cycle time - 1.4 microseconds
7094 II channel cycle time - 2.1 microseconds

7044 to 7094 II Timing Chart (Work Sheet 2)

Appendix C: Manual References

The following manuals are for reference use:

IBM 7090/7094/7094 II Systems

FORM	TITLE
223-6895	*7090 Data Processing System, Customer Engineering Instruction-Maintenance Manual.*
223-2639	*7094 Data Processing System, Customer Engineering Instruction-Maintenance Manual, Volume 1 (Introduction, System and Functional Components, and Timing).*
223-2550	*7094 Data Processing System, Customer Engineering Instruction-Maintenance Manual, Volume 2 (Arithmetic Operations).*
223-2707	*7094 Data Processing System Customer Engineering Instruction-Maintenance Manual, Volume 3 (Non-Arithmetic Operations, Trapping, Compatibility, Instruction Overlap, and 7151-2 Console).*
223-2721	*7094 II Data Processing System, Customer Engineering Instruction-Maintenance Manual, Volume 1 (Introduction, Component Circuits, System and Functional Components, and Timing).*
223-2722	*7094 II Data Processing System, Customer Engineering Instruction-Maintenance Manual, Volume 2 (Arithmetic Operations).*
223-2723	*7094 II Data Processing System, Customer Engineering Instruction-Maintenance Manual, Volume 3 (Non-Arithmetic Operations, Instruction Overlap, Trapping, Compatibility, and 7151-2 Console).*
223-6910	*7607 Data Channel, Customer Engineering Instruction-Maintenance Manual.*

IBM 7040/7044 Systems

FORM	TITLE
223-2651	*7040/7044 Data Processing System, Customer Engineering Manual of Instruction, Central Processing Unit.*
223-2659	*7040/7044 Data Processing System, Customer Engineering Maintenance Manual, Central Processing Unit.*
223-2755	*7040/7044 Data Processing System, Customer Engineering Manual of Instruction, Channels B, C, D, E.*
223-2712	*7040/7044 Data Processing System, Customer Engineering Maintenance Manual, Channels B, C, D, E.*
223-2753	*7040/7044 Data Processing System, Customer Engineering Instruction-Maintenance Manual, Trapping.*
A22-6803	*Direct Coupled Processing Units — IBM 7040 to 7090/7094/7094 II; 7044 to 7094/7094 II.*

Index

CUT ALONG LINE

COMMENT SHEET

IBM 7040 TO 7090/7094; 7044 TO 7094/7094 II

DIRECT COUPLED MULTIPROCESSING

FIELD ENGINEERING INSTRUCTION-MAINTENANCE, FORM 223-2806-0

FROM

NAME _____

OFFICE NO. _____

FOLD FOLD

CHECK ONE OF THE COMMENTS AND EXPLAIN IN THE SPACE PROVIDED

☐ SUGGESTED ADDITION (PAGE , TIMING CHART, DRAWING, PROCEDURE, ETC.)

☐ SUGGESTED DELETION (PAGE)

☐ ERROR (PAGE)

EXPLANATION

FOLD FOLD

NO POSTAGE NECESSARY IF MAILED IN U.S.A.

FOLD ON TWO LINES, STAPLE, AND MAIL

STAPLE STAPLE

FOLD FOLD

FIRST CLASS
PERMIT NO. 81
POUGHKEEPSIE, N. Y.

BUSINESS REPLY MAIL
NO POSTAGE STAMP NECESSARY IF MAILED IN U. S. A.

POSTAGE WILL BE PAID BY
IBM CORPORATION
P. O. BOX 390
POUGHKEEPSIE, N. Y.

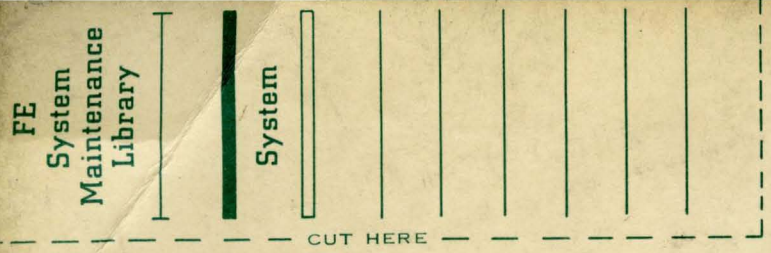
ATTN: FE MANUALS, DEPARTMENT B96

CUT ALONG LINE

FOLD FOLD

2/65:1.5M-EP-120

STAPLE STAPLE



223-2806-0

Printed in U. S. A. 223-2806-0

IBM

International Business Machines Corporation
Field Engineering Division
112 East Post Road, White Plains, N.Y. 10601