CR151238

MCDONNELL DOUGLAS TECHNICAL SERVICES CO.
HOUSTON ASTRONAUTICS DIVISION

SPACE SHUTTLE ENGINEERING AND OPERATION SUPPORT

1.3-DN-C0613-001

DATA PROCESSING SYSTEM AND INTERFACING ELEMENTS TIME BASE ANALYSIS

AVIONICS SYSTEMS ENGINEERING

21 MARCH 1977

This Design Note is Submitted to NASA Under Task Order No. C0613, in Fulfillment of Contract NAS 9-14960.

PREPARED BY: J. D. Blackburn
J. D. Blackburn
Staff Engineer
488-5660 Ext. 285

APPROVED BY: T. G. Politte
T. G. Politte
Task Order Manager
488-5660 Ext. 285

APPROVED BY: L. R. Blanke
L. R. Blanke
Technical Manager
488-5660 Ext. 260

APPROVED BY: R F Pannett
R. F. Pannett
Project Manager
488-5660 Ext. 258

1.0 SUMMARY

This note describes the processing of time in the Orbiter System Services software and the associated facilities provided to the user community. The descriptions are directed toward showing the functional intent of the design rather than the actual implementation. Simplified flow diagrams are included in Appendix A to enhance understanding of these time related functions.

Based upon detailed analysis of a preliminary review copy of the Approach and Landing Test (ALT) System Software Detailed Design Specification (Reference A) and the Program Listings for Version 17 Prime (Reference B), the processing of time has the potential for error free operation.

The processing of time is not expected to change between ALT and the Operational Flight Tests (OFT) other than differences in value of some constants for control and limit checking.

Due to the dynamic nature of onboard time processing and its criticality to the successful operation of the Orbiter, it is recommended that a comprehensive list of external variables, their locations, initial values, and a "where used" listing be produced, as a by-product of the link edit process, for all non-HAL coding. In addition, a careful review of the verification test procedures for the System Services time-related software is recommended.

2.0 INTRODUCTION

The Orbiter onboard processing of time and its use within the System Services software is not well understood by the engineering community outside of those individuals directly involved in the software development. This became evident when an issue arose regarding the Master Timing Unit (MTU) end of year rollover (reset of DAY counter to 1). Few of the engineers involved in that discussion understood that Greenwich Mean Time (GMT) was used as a basis for flight software interval time scheduling and that the rollover of time would produce a serious internal software time processing problem. This lack of understanding is largely due to the onboard processing of time being distributed over a number of subroutines and not being adequately described in consolidated documentation. This design note attempts to fill the need for a single document describing the functional intent of these time processes so that the engineering community may make more effective use of the time-related facilities provided by the System Services software.

3.0 DISCUSSION

Time is kept within each General Purpose Computer (GPC) to schedule events, to time tag events for downlist and/or logging, and to display "time" on the Cathode Ray Tube (CRT) displays. Internal time (MY TIME) is used for these purposes. MY TIME is periodically coordinated and corrected within the GPC common set (CS) by use of the MTU, or MY TIME of the Prime GPC (PRIME TIME).

Three formats are used for time. The Flight Computer Operating System (FCOS) format is used for computation throughout System Services, the MTU format is used when interfacing with the MTU, and the floating point format is used when interfa[illegible]g with applications programs.

Time [illegible]duling of events or processes is accomplished by setting an interval tim[illegible] the difference between the present time and the desired time of exe[illegible]n. When the interval timer runs out, the event or process is dispatched.

3.1 Time Sources

MY TIME is the time base for the software system. Time continuity is maintained by using a common time reference source to adjust MY TIME to agree with the common set selected time source. The preferred time reference source is the MTU. Next is PRIME TIME and finally MY TIME. The criteria for assessing acceptability of a time source is the absolute difference between the reference source and MY TIME being equal to or less than 580[1] microseconds.

The MTU is a line replaceable unit containing redundant master oscillators that step three accumulators for GMT and three for Mission Elapsed Time (MET). These accumulators keep ephemeris time and can be loaded and read by the GPC's.

---

(1) 450 microseconds for OFT (recommended by IBM-see Reference C)

Although a set of MET accumulators is provided in the MTU, these are not used to maintain MET for use by the flight software. Rather, an MET is synthesized in FCOS. However, the MTU MET accumulators do provide a direct MET cockpit time readout for OFT.

Each of the MTU accumulator pairs (GMT/MET) are connected to a flight critical Multiplexer Demultiplexer (MDM). FCOS reads the MTU GMT accumulators during System Software Interface Processing (SSIP) common set synchronization. Processing of this time is called by System Control (SC) and is performed by the Time Redundancy Management function in FCOS. Each GMT accumulator is inspected in turn to find one that is acceptable. If no acceptable accumulator is found, PRIME TIME is tested. A default to MY TIME occurs if PRIME TIME fails the 580[(1)] microsecond test. The difference between the time of the selected source and MY TIME is then used to update MY TIME to agree with the source.

3.2 GPC Time Formats

Time is represented in the GPC's in FCOS, MTU, and floating point formats. The FCOS format consists of a full word (32 bits) that counts microseconds up to one half hour, and a half word that counts the number of elapsed half hours. This format (shown in Figure 1a) has a capacity in excess of 1365 days and is the format used for all System Services time manipulation.

(1) 450 microseconds for OFT (recommended by IBM-see Reference C)

a. FCOS TIME FORMAT

	16 BITS
HALF HOUR MULTIPLE COUNTER	HALF HOURS

	32 BITS
HALF HOUR MICROSECOND COUNTER	MICROSECONDS

b. MTU TIME FORMAT

BIT POSITION 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

FIRST WORD	DAY $\times 10^2$	DAY $\times 10^1$	DAY $\times 10^0$	HR $\times 10^1$	HR $\times 10^0$	
SECOND WORD	MIN. $\times 10^1$	MIN. $\times 10^0$	SEC. $\times 10^1$	SEC. $\times 10^0$	UNUSED	
THIRD WORD	UNUSED	FRACTIONAL SECONDS IN BINARY WITH LEAST SIGNIFICANT BIT = 0.125 MILLISECONDS				

c. FLOATING POINT TIME FORMAT

S	EXP	MANTISA

1 BIT SIGN
7 BITS HEXADECIMAL EXPONENT (EXCESS 64)
56 BIT FRACTION - POSITIVE - HEXADECIMAL NORMALIZED

The MTU format, used to interchange time with the MTU, contains ephemeris time in four bit Binary Coded Decimal (BCD) characters as shown in Figure 1b. Note that the MTU steps in 125 microsecond increments.

The applications programs all use the floating point format. Time in this format is in seconds as a double precision floating point number per Figure 1c. The capacity far exceeds that of the other formats ($>10^5$ days while maintaining precision to one microsecond).

3.3 System Services Time Keeping

Two types of continuously incrementing time (MY TIME and interval time) are kept within the GPC as well as a reference time for MET = 0. MY TIME is kept by a combination of a running counter, Program Counter 1 (PC1), and the Software Clock (SWC). The PC1 is a combination of a 16 bit hardware register (least significant half) and a 16 bit core location (most significant half) forming a 32 bit decrementing register. The counter is loaded with a binary number representing 2,097,151 and counts down at a rate of once per microsecond. When PC1 is equal to zero, a PC1 interrupt is issued to System Services. The counter continues to decrement so that the time lag in recognizing and processing the interrupt is not critical. When the PC1 interrupt is processed, the binary value 2,097,151 is added to PC1 and also to the microsecond counter of the SWC. Since PC1 continues to decrement into the negative region after the interrupt, arithmetically adding to PC1 accounts for the time since the interrupt. The current value of MY TIME is calculated by adding the PC1 elapsed time (2,097,151 - PC1 value) to the SWC. Figure A-1 depicts these processes.

The interval timer, Program Counter 2 (PC2), operates similarly to PC1 causing a PC2 interrupt when the counter reaches zero. There is no equivalent to the SWC for the interval timer because PC2 possesses adequate capacity (greater than 71 minutes). PC2 is used as an interval timer to cause a process to be executed in a given number of microseconds from "now."

System Services also maintains the time reference denoting MET initialization (usually considered as the GMT of LIFT OFF). To initialize MET, MY TIME for the initializing event is computed and stored in "MET Reference Time" so that MET can be calculated at any future time by subtracting "MET Reference Time" from MY TIME.

3.4 Timer Queue Element (TQE) Times

The Timer Queue is a list of tasks to be performed at specific times. This list consists of entries called "elements" and is sorted into "time to execute" order. Three times are frequently encountered in examining the functions that reference the timer queue: "Last Expired TQE Time", "TQE Time of Expiration", and "Top TQE Time of Expiration".

The "Last Expired TQE Time" is the MY TIME when the last interval timer (PC2) interrupt occurred. This is the time that the last time-scheduled process was placed in "ready" status for execution. This time is important because it represents the time of initiation of the scheduled process and is used for all computations concerning the process.

"TQE Time of Expiration" is the time at which a TQE is to place a referenced process into the active run queue for execution. The "Top TQE Time of Expiration" is the "TQE Time of Expiration" corresponding to the top element in the active run queue, i.e. the element that is set to expire next. This will also be the element which PC2 is currently timing. The processing of TQE's is shown by Figure A-2.

3.5 Time Maintenance

The following subsections discuss the major elements of time initialization, MTU updating, and time redundancy management.

3.5.1 MY TIME Initialization

When the first GPC is brought online by Initial Program Load (IPL), it sees no other members of the common set and will initialize MY TIME. An attempt is made to read the MTU and, if successful, this time is used to set the SWC after accounting for the value in PC1 (i.e. PC1 is not reset but the value is used to bias the SWC to make MY TIME equal to the MTU time).

As succeeding GPC's are brought into the common set, they will utilize PRIME TIME (MY TIME of the Prime GPC) to initialize their internal GMT clocks (their MY TIME). Figures A-3a and b show the flow for both of these processes. Note that if the Prime GPC designation shifts to a GPC coming into the common set, MY TIME of the incoming GPC will be initialized prior to the switching of PRIME designation.

3.5.2 MTU Update

Updates can be initiated to bring the MTU into agreement with the ground processors and to synchronize the MTU-GMT and MY TIME. Provisions are available to reset and update the GMT and MET MTU accumulators. When an MTU update request is received, it is put into the MTU format and the Input/Output (I/O) operation is requested to perform an update. Updates to the MTU-MET accumulators also update the internal MET Reference Time, thus keeping these times in agreement.

If the request is to synchronize the accumulators, the MTU-GMT is read and a zero delta update is formulated. If no MTU time is available, the update is formulated from PRIME TIME. Execution of either of these updates will set all GMT accumulators alike. Following the Queue of the I/O to update the MTU, MY TIME is set to agree with the update time. Figure A-4 shows the flows for the MTU update functions.

3.5.3 Time Redundancy Management

Every eighteen[(1)] minor cycles the MTU's are read during the normal SSIP processing. After the Intercomputer Communication (ICC) message exchange and the MTU "reads" are complete, the Time Redundancy Management function is executed. The MTU times are compared, in turn, with MY TIME and are rejected as invalid if they differ by more than 580[(2)] microseconds. If no valid MTU is found, PRIME TIME is similarly tested and if that fails, internal time (MY TIME) is used. The selected source is flagged and MY TIME is adjusted to agree with the source, see Figure A-5.

3.6 Time/Date Tags for Applications Programs

There are three times (Runtime, Clocktime, and MET) and one date that can be requested by applications programs using Supervisor Call (SVC) 22. The time requests produce double precision floating point numbers scaled in seconds and the date request produces two 16 bit values, years and days, "packed" in Application General Register 5.

When "Runtime" is requested, the current MY TIME is calculated using the count in PC1 and the value of the SWC. These results are then converted to seconds in double precision floating point format and left in the Floating Point General Registers 0 and 1 as expected by the HAL/S compiled application programs.

---

(1) twenty four for OFT (recommended by IBM-see Reference C)

(2) 450 microseconds OFT (recommended by IBM-see Reference C)

A "Clocktime" request produces a time similar to "Runtime" except the converted time is that of the last event or process initiated by the interval timer, PC2.

A request for "Mission Elapsed Time" converts "Clocktime" minus MET Reference Time to floating point. MET Reference Time would normally be the time of lift off.

The "Date" request causes the program to calculate days into the mission then add the day and year of lift off. The year is placed in the upper (most significant) half of the Application General Register 5 and the day is placed in the lower half. Both of these numbers are fixed point binary integers. Figure A-6 shows the flow of these time and date functions supporting the FCOS/HAL interface.

An additional time tagging function is provided via SVC 32 which causes the "Last Expired TQE Time" to be converted to floating point and be stored in a specified location. The location in which to store the time is passed to the processing routine in the SVC parameter list. This function is shown in Figure A-3c.

Time tags are provided by System Services to display time in the upper right corner of the CRT displays. This time is MET as synthesized by the System Services software in FCOS (may change to GMT for OFT). Time tags are provided in the fault messages from error processors that are displayed on the fault message line. The Cyclic Display processor provides conversion of the FCOS time format to display format in hours: minutes: seconds as required by the CRT display. In some cases the fault message comes from applications programs in floating point format. In these cases, the FCOS routine for conversion from floating point to FCOS format is used as a preprocessor for the display conversion.

4.0 CONCLUSION

The time processing in System Services is distributed among assembly programs and procedures. Each of these stands alone when assembled and they are interrelated via the link editor. This causes many symbols to be related via the EXTERNAL reference which provides no interrelationship in the assembly listing. In addition, the REPLACE pseudo operator, especially when combined with the NO LIST operator, further subverts efforts to understand the code. Design documentation and resulting code do not make it apparent that adequate interface control between System Services components and symbol name assignment control were established prior to the implementation phase of the design/development process. Coded adjustments to accommodate the resulting interfaces, variations in names used for the same parameters and the absence of control documentation are evidence that interface control was not systematically administered. These factors fragment time processing to a point where analysis is extremely difficult. This design note describes the functional intent of the design with the hope that it will be useful to those outside of the software development community. The research, performed in generating this paper, indicates a potential for trouble free time processing operation.

It is recommended, in as much as interface control and symbol name documentation is unavailable, that a comprehensive list of external symbols, their locations, initial values, and a "where used" listing be produced as a by-product of the link edit process to aid in anomaly analysis. These data are presently produced for HAL symbols in the HAL STAT and should likewise be generated for symbols of the NON-HAL programs of System Services. In addition, the verification test procedures should be reviewed for adequacy in testing the integrated performance of these time processing functions.

5.0 REFERENCES

A. Preliminary Review Copy of "Approach and Landing Test (ALT) System Software Design Specification, IBM Document No. 77-SS-3576, dated 28 February 1977.

B. Program Listing ALT Version 17 Prime System Services, released 10 December 1976.

C. GPC Timekeeping (an update to TR 77-SS-3605), dated 3 February 1977.

APPENDIX A

ALT SYSTEM SERVICES SOFTWARE FUNCTIONAL TIMING FLOW DIAGRAMS

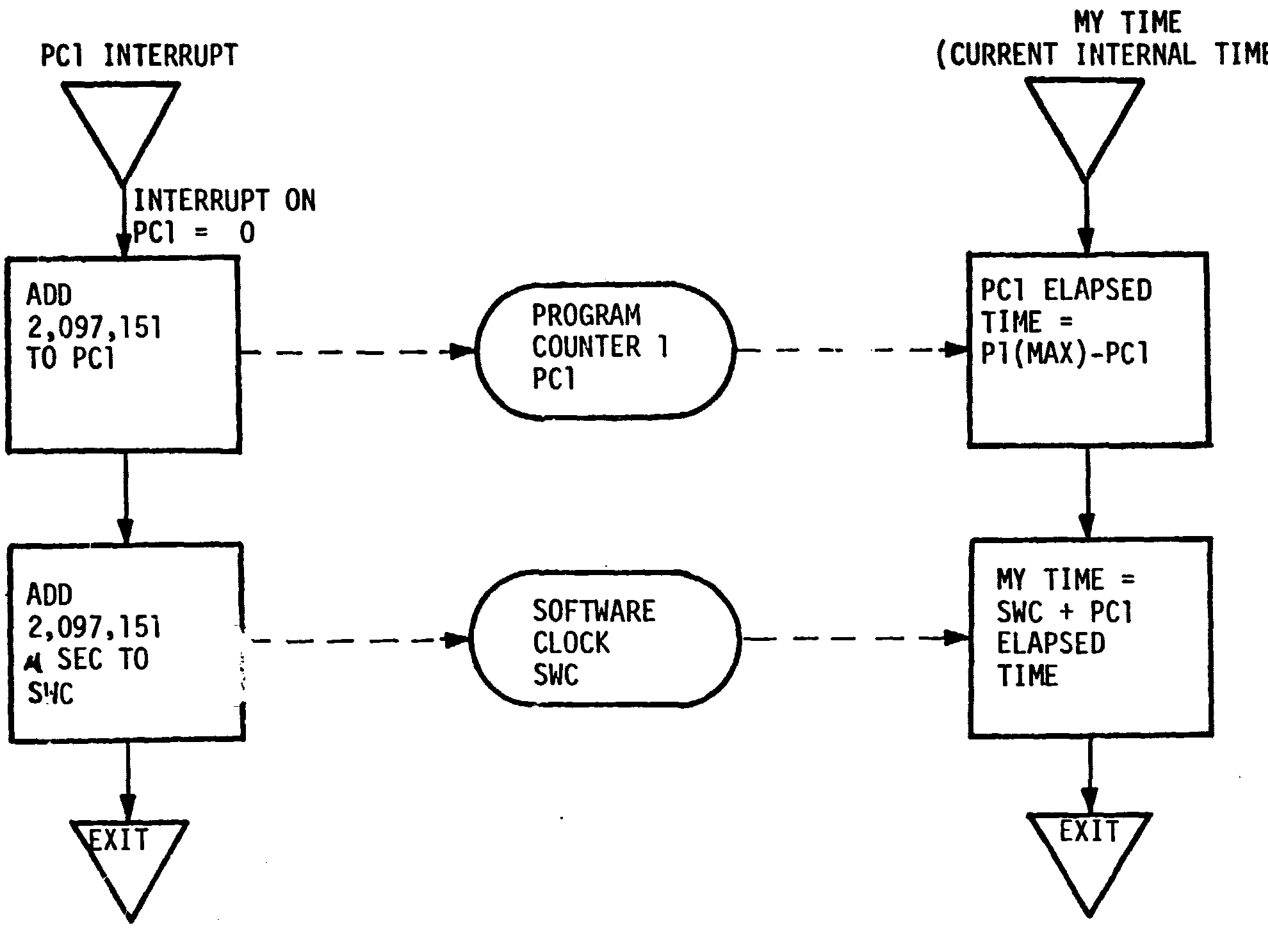

FIGURE A-1 MY TIME (INTERNAL GPC TIME)

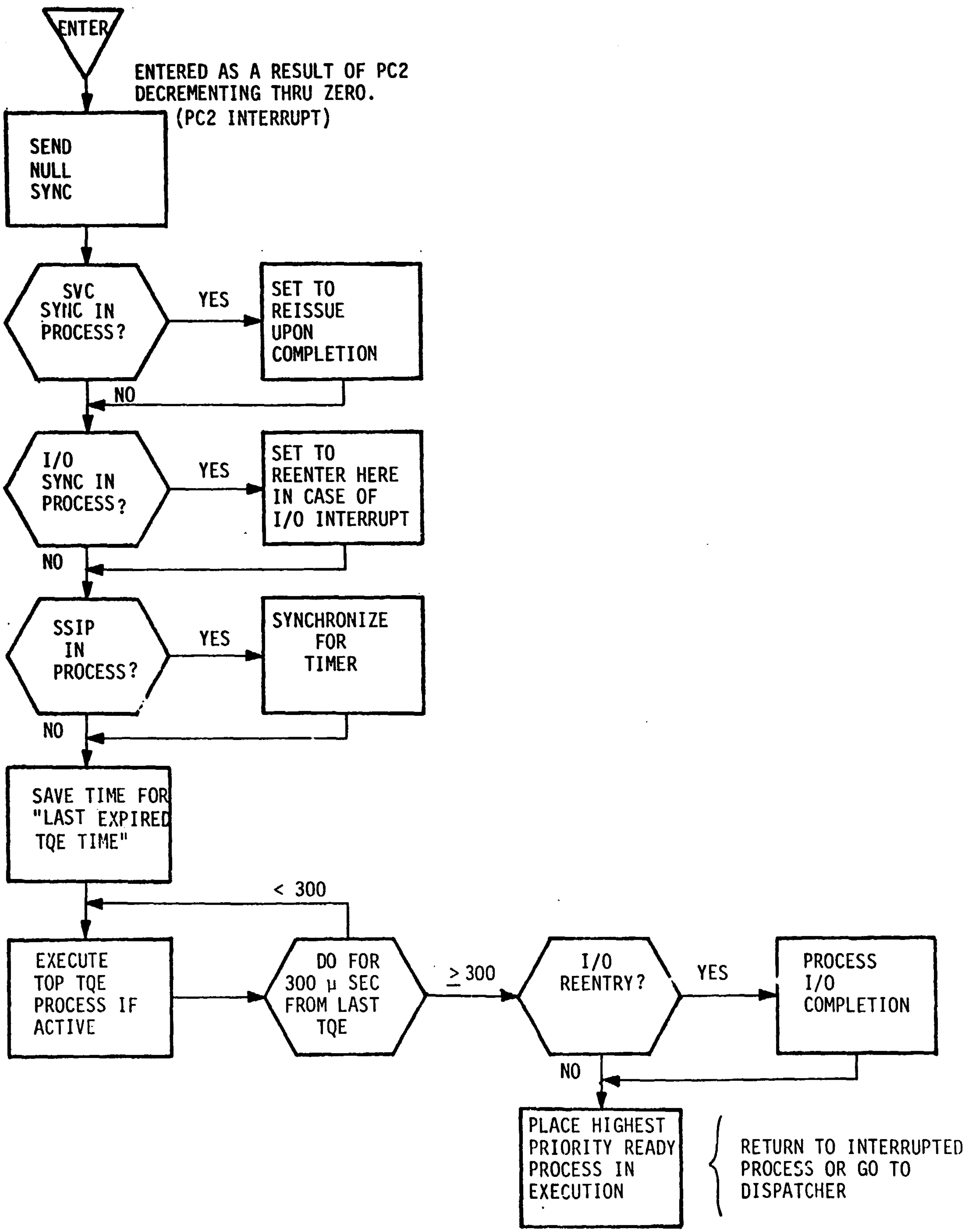

FIGURE A-2 TIMER QUEUE ELEMENT EXPIRATION PROCESSING

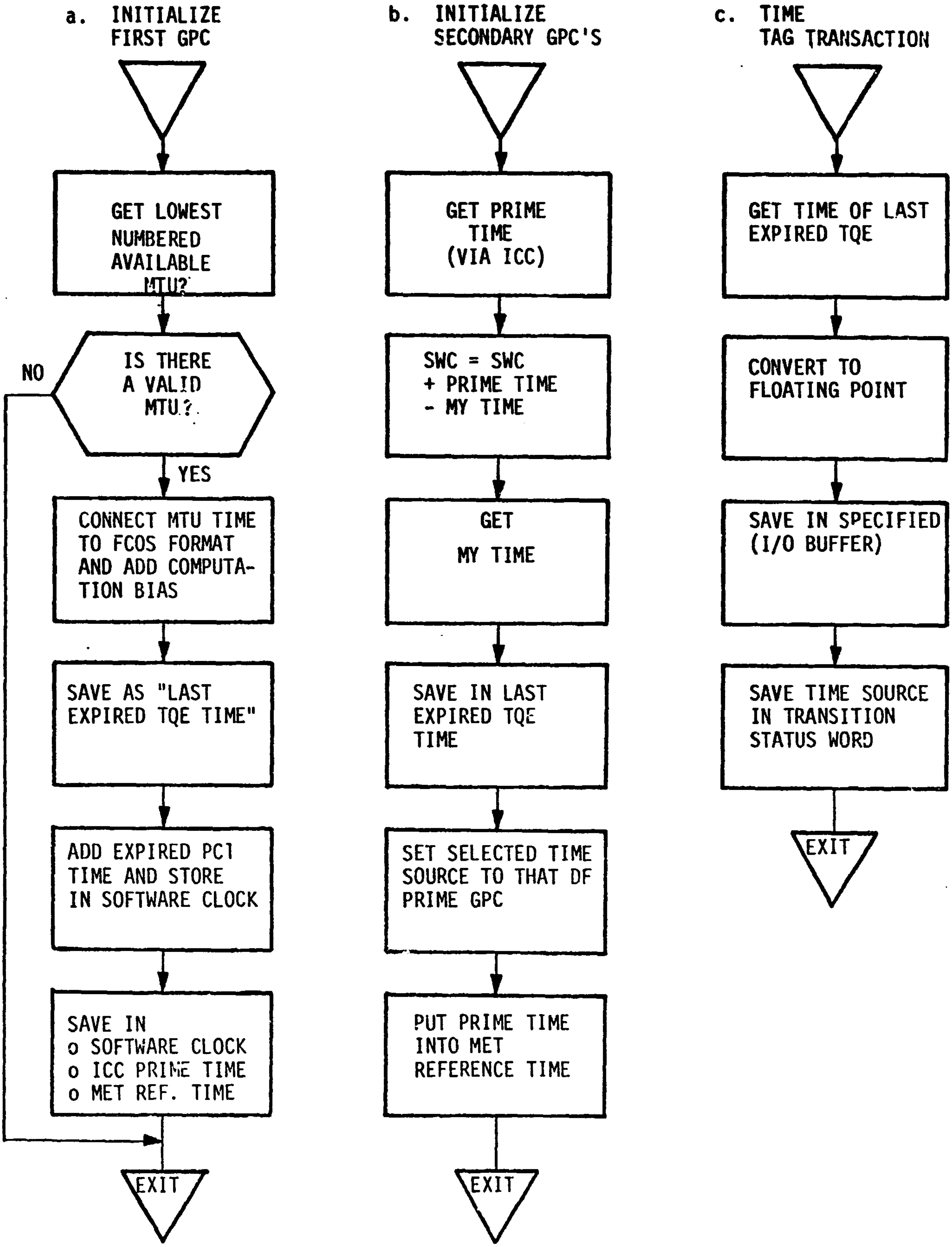

FIGURE A-3 GPC TIME (MTU REDUNDANCY MANAGEMENT)

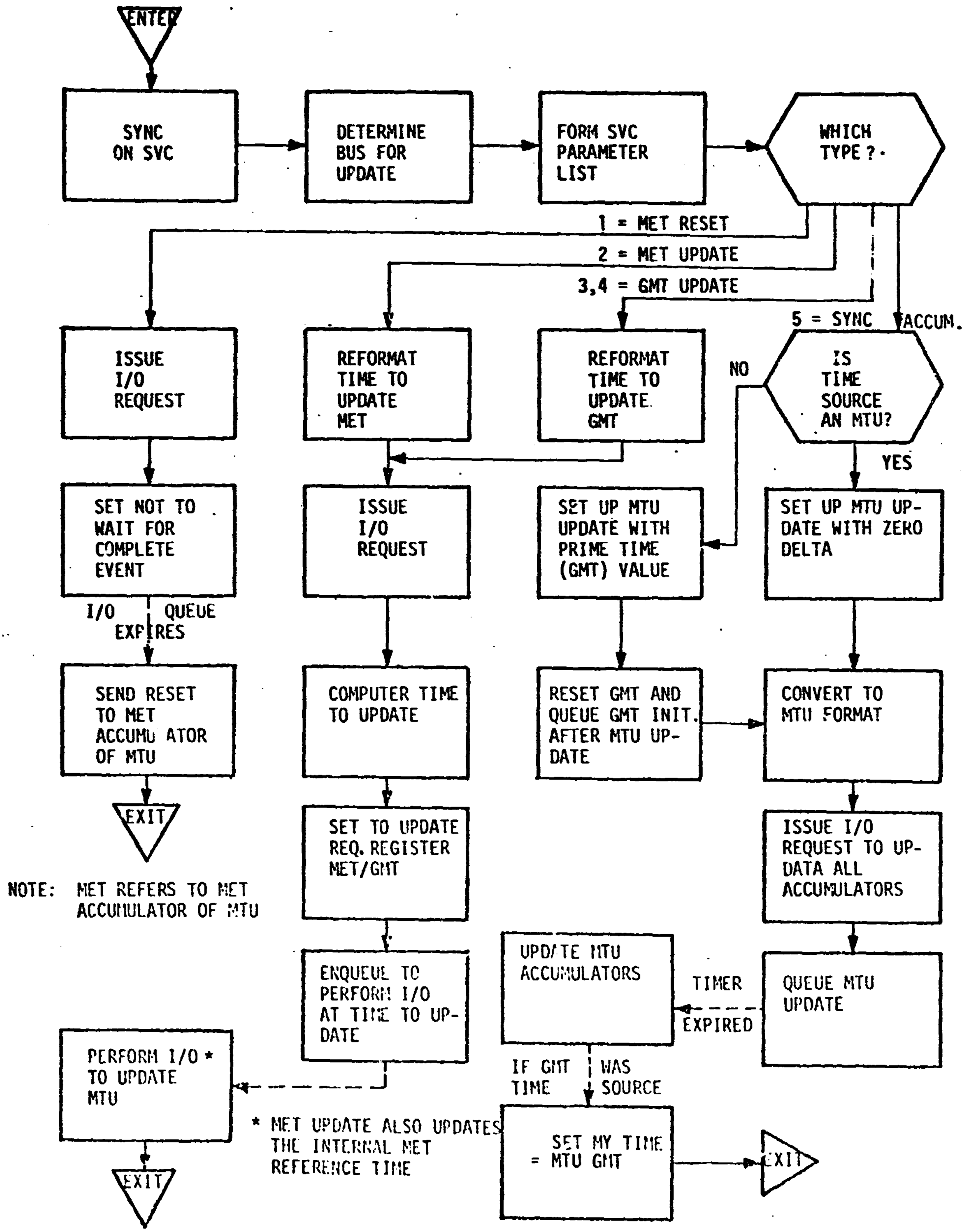

FIGURE A-4 MTU UPDATE

O DONE EVERY 18 MINOR CYCLES[1] AFTER SSIP (COMMON SET SYNC)

(1) 18 FOR ALT (IBM RECOMMENDS 24 FOR OFT)

(2) MTU TIME IS ACCEPTABLE IF NO I/O ERROR OCCURRED AND MTU TIME IS WITHIN 580 MICROSECONDS (450 FOR OFT) OF TIME OF READ (IN MY ICC BUFFER).

(3) PRIME TIME IS ACCEPTABLE IF IT IS WITHIN 580 MICROSECONDS OF TIME OF READ (450 FOR OFT)

(4) K IS A LOADABLE CONSTANT

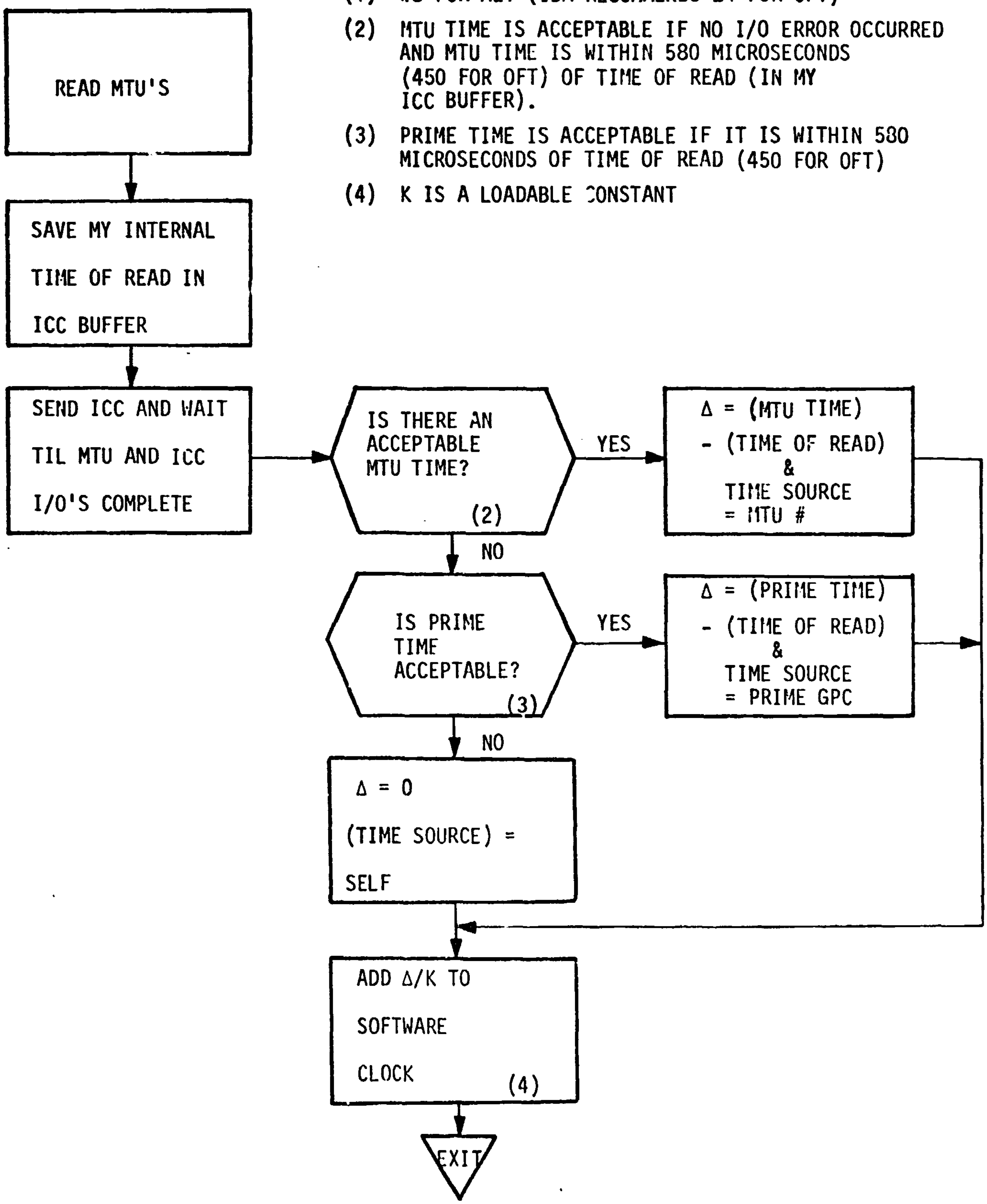

FIGURE A-5 TIME REDUNDANCY MANAGEMENT FUNCTION