AD-A169 870

Project Report
ATC-133

TCAS Experimental Unit (TEU) Hardware Description

D.A. Spencer
R.R. LaFrey
J. DiBartolo
W.H. Harman

6 June 1986

Lincoln Laboratory
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LEXINGTON, MASSACHUSETTS

Prepared for the Federal Aviation Administration

This document is available to the public through the National Technical Information Service, Springfield, Virginia 22161.

DTIC
ELECTE
JUL 1 8 1986
S D

Best
Available
Copy

TECHNICAL REPORT STANDARD TITLE PAGE

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
DOT/FAA/PM-85/2		

4. Title and Subtitle
TCAS Experimental Unit (TEU) Hardware Description

5. Report Date
6 June 1986

6. Performing Organization Code

7. Author(s)
David A. Spencer, Joseph DiBartolo, Raymond R. LaFrey, William H. Harman

8. Performing Organization Report No.
ATC-133

9. Performing Organization Name and Address
Lincoln Laboratory, MIT
P.O. Box 73
Lexington, MA 02173-0073

10. Work Unit No. (TRAIS)

11. Contract or Grant No.
DOT-FA77WAI-817

12. Sponsoring Agency Name and Address
Department of Transportation
Federal Aviation Administration
Program Engineering and Maintenance Service
Washington, DC 20591

13. Type of Report and Period Covered
Project Report

14. Sponsoring Agency Code

15. Supplementary Notes
The work reported in this document was performed at Lincoln Laboratory, a center for research operated by Massachusetts Institute of Technology, under Air Force Contract F19628-85-C-0002.

16. Abstract

This report describes the hardware design of the TCAS Experimental Units (TEU's) constructed by Lincoln Laboratory to support the design and validation of the Traffic Alert and Collision Avoidance System (TCAS) for the FAA. Section 1.0 presents an overview of the operation of the TEU's, in order to give some context for the hardware design. References are given to more extensive descriptions of the TCAS system operation and software design. Section 2.0 constitutes the bulk of the report, and is a detailed description of the TEU hardware design.

The purpose of this description is to document the design details of the equipment which was used to develop and validate the signal processing techniques and algorithms which appear in the TCAS II Minimum Operational Performance Standard, the TCAS National Standard and various technical reports listed in the references. A second purpose is to provide design guidance to potential TCAS II manufacturers, in the form of a detailed description of a feasible design with documented performance. Finally, this document is a manual for future use and maintenance of the TEU's.

17. Key Words		18. Distribution Statement
Traffic Alert and Collision Avoidance System (TCAS)	threat logic	Document is available to the public through the National Technical Information Service, Springfield, VA 22161.
collision avoidance	fruit	
Mode S	synchronous garble	
whisper-shout	dynamic minimum threshold	
degarble	squitter	
angle of arrival	TCAS Experimental Unit (TEU)	

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	130	

Form DOT F 1700.7 (8-69)

OVERVIEW

This report describes the hardware design of the TCAS Experimental Units (TEU's) constructed by Lincoln Laboratory to support the design and validation of the Traffic Alert and Collision Avoidance System (TCAS) for the FAA. Section 1.0 presents an overview of the operation of the TEU's, in order to give some context for the hardware design. References are given to more extensive descriptions of the TCAS system operation and software design. Section 2.0 constitutes the bulk of the report, and is a detailed description of the TEU hardware design.

The purpose of this description is to document the design details of the equipment which was used to develop and validate the signal processing techniques and algorithms which appear in the TCAS II Minimum Operational Performance Standard [Ref. 6], the TCAS National Standard [Ref. 8] and various technical reports listed in the references. A second purpose is to provide design guidance to potential TCAS II manufacturers, in the form of a detailed description of a feasible design with documented performance. Finally, this document is a manual for future use and maintenance of the TEU's.

Accession For
NTIS GRA&I
DTIC TAB
Unannounced
Justification
By
Distribution/
Availability Codes
Dist | Avail and/or Special
A-1

CONTENTS

	Overview	iii
	Illustrations	vi
	Tables	viii
1.0	INTRODUCTION	1
	1.1 Purpose	1
	1.2 Summary of Operation	1
2.0	TCAS EQUIPMENT	4
	2.1 Major System Elements	4
	2.2 Modulation Control Unit	8
	2.2.1 Function	8
	2.2.2 Programming	8
	2.2.3 Operation	9
	2.2.3.1 Summary	9
	2.2.3.2 MCU Controller	13
	2.2.3.3 Mode Decoder and Sequence Generator	13
	2.2.3.4 Mode S Data Buffer and DPSK Generator	17
	2.2.3.5 Range Counter	20
	2.2.3.6 Suppression Logic	22
	2.3 Transmitter/Receiver	24
	2.3.1 Microwave Signal Source (1030 MHz)	24
	2.3.2 Transmitter Chain	24
	2.3.2.1 PAM/DPSK Modulators	27
	2.3.2.2 Microwave Power Module	27
	2.3.2.3 Digital Attenuator	29
	2.3.2.4 Bandpass Filter	29
	2.3.3 Antenna Interface and Transmitter Monitor	29
	2.3.4 Receiver Chain	29
	2.4 deo Pulse Quantizer	32
	2.5 Video Digitizer and ATCRBS Reply Processor	39
	2.5.1 General Description	39
	2.5.2 Video Digitizer	45
	2.5.3 Pseudo Leading Edge Generation and Short Pulse Rejection	45
	2.5.4 Reply Detection	53
	2.5.5 Reply Decoder Processing	53
	2.5.6 Reply Assembler	58
	2.6 Mode S Reply Processor	62
	2.6.1 General Description	62
	2.6.2 State Controller	64
	2.6.3 Sliding Window Detector	68
	2.6.4 Preamble Detector	68
	2.6.5 Message Processor	68
	2.6.6 Computer Interface Controller	77
	2.6.6.1 Discrete or Squitter Reply Transfer	77
	2.6.6.2 Test Reply Transfer	77

2.7 Angle of Arrival (AOA) Subsystem 77
2.7.1 Antenna and Antenna Feed Network 83
2.7.2 AOA Receiver 83
2.7.3 AOA Digitizer 89
2.7.4 AOA Processor 89
2.7.4.1 AOA Control 93
2.7.4.1.1 AOA ATCRBS Control 93
2.7.4.1.2 AOA Mode S Controller 99
2.7.4.2 AOA Accumulator 99
2.7.4.3 AOA Reply Processor Interface 101

2.8 Mode S Transponder Interface 104
2.8.1 Purpose 104
2.8.2 TEU Implementation 106
2.8.3 General Principles 108

2.9 General Purpose I/O Interface 108
2.9.1 Function 108
2.9.2 Summary of I/O Interface Conventions 109
2.9.3 Programming the General Purpose I/O Interface 111
2.9.4 Operation 116

2.10 Real-Time Clock 118
2.10.1 General Description 118
2.10.2 Programming 119

2.11 Bootstrap Loading 119

REFERENCES 121

ILLUSTRATIONS

2.1-1	Major TEU components	5
2.1-2	Digital processing chassis components and interconnections	6
2.2-1	MCU control block format	10
2.2-2	MCU block diagram	12
2.2-3	MCU controller block diagram	14
2.2-4	MCU controller state diagram	15
2.2-5	MCU mode decoder and sequence generator	16
2.2-6	Sequence generator timing signals	18
2.2-7	Mode S data buffer and DPSK generator	19
2.2-8	Range counter	21
2.2-9	Suppression logic	23
2.3-1	TEU transmitter/receiver block diagram	25
2.3-2	Transmitter chain block diagram	26
2.3-3	Power module response to DPSK transition	28
2.3-4	Transmitter power supply interlock circuit	30
2.3-5	Receiver chain block diagram	31
2.3-6	Receiver log video response	34
2.4-1	VPQ simplified circuit diagram	36
2.4-2	VPQ waveforms	37
2.4-3	VPQ DMTL timing network	38
2.4-4	ATCRBS DMTL waveforms	40
2.4-5	Mode S DMTL waveforms	40
2.5-1	ATCRBS reply processor block diagram	41
2.5-2	Garbled reply and phantom reply examples	43
2.5-3	ATCRBS reply processor data format	44
2.5-4	Digitizer clock control	46
2.5-5	Digitizer logic	47
2.5-6	Trailing edge conditions	48
2.5-7	Digitizer timing	49
2.5-8	Pseudo and extra leading edge generator	50
2.5-9	Pseudo and extra leading edge pulse generation, no embedded ALE's	51
2.5-10	Extra leading edge generation with embedded ALE's	52
2.5-11	Reply detector	54
2.5-12	Reply decoder control	55
2.5-13	Reply decoder number 1	56
2.5-14	Reply decoder timing	57
2.5-15	Reply data output logic	59
2.5-16	Reply assembler state controller and associated circuitry	60
2.5-17	State diagram of reply assembler controller	61

2.6-1	Mode S reply processor block diagram	63
2.6-2	Mode S reply processor data formats	65
2.6-3	State controller block diagram	66
2.6-4	State controller state diagram	67
2.6-5	Sliding window detector	69
2.6-6	Mode S preamble detection timing	70
2.6-7	Preamble detector	71
2.6-8	Message processor and reply buffer block diagram	72
2.6-9	Message processor timing	74
2.6-10	Message processor timing	75
2.6-11	Address decoder	76
2.6-12	Mode S I/O controller	78
2.6-13	Mode S I/O controller state diagram	79
2.6-14	Mode S I/O controller timing diagram	80
2.7-1	TEU bearing augmentation - RF components	81
2.7-2	TEU bearing augmentation - digital components	82
2.7-3	AOA antenna orientation	84
2.7-4	AOA antenna feed network	85
2.7-5	Phase and amplitude response of the AOA antenna versus azimuth	86
2.7-6	Phase and amplitude response of the AOA antenna versus elevation angle	87
2.7-7	TEU receiver block diagram	88
2.7-8	TEU AOA digitizer	90
2.7-9	AOA digitizer waveforms	91
2.7-10	TEU AOA processor	92
2.7-11	TEU AOA Processor - AOA controller	94
2.7-12	TEU AOA ATCRBS timing diagram	96
2.7-13	TEU AOA ATCRBS state control diagram	97
2.7-14	AOA sampling of overlapped ATCRBS pulses	98
2.7-15	AOA accumulator	100
2.7-16	TEU AOA accumulator timing diagram	102
2.7-17	AOA reply processor interface	103
2.7-18	Modified Mode S I/O controller state diagram	105
2.7-19	Modified state diagram of ATCRBS reply assembler controller	106
2.7-20	AOA word format for ATCRBS and Mode S replies	107
2.9-1	General purpose I/O interface	117

TABLES

2.3-1	TEU RF Link Power Budget	33
2.9-1	GPIO Programmed I/O Summary	112
2.9-2	GPIO Data Channel Summary	114

1.0 INTRODUCTION

1.1 Purpose

This document describes the hardware design of the TCAS Experimental Units (TEU's) built by Lincoln Laboratory for the Federal Aviation Administration. These are experimental versions of TCAS (Traffic Alert and Collision Avoidance System), a system designed to alert pilots to nearby transponder-equipped aircraft and to assist them in avoiding collisions.

The purpose in constructing the TEU's was to demonstrate a complete TCAS system that integrated the solutions to the many technical problems in areas ranging from elimination of multipath interference on the air-air link, to the modeling of aircraft trajectories and selection of appropriate commands, to the design of an appropriate interface between TCAS and the pilot. During the development process new problems were discovered and solutions developed, leading to the evolution of a large body of experience which is reflected in the standards that have been developed for commercial TCAS equipment.

1.2 Summary of Operation

TCAS interrogates (at 1030 MHz) the radar beacons or transponders that are carried on all commercial and many general aviation aircraft. TCAS interrogates both the Air Traffic Control Radar Beacon System (ATCRBS) transponders that are the current standard transponders and also the Mode S transponders that will begin to be installed over the next decade. When interrogated these transponders reply (at 1090 MHz) after a fixed "turnaround" time. This allows the range to be determined. The reply contains data bits (12 for ATCRBS, 56 or 112 for Mode S). If the appropriate interrogation type is sent, the data in the reply contains a coded form of the aircraft's altitude. Some ATCRBS-equipped aircraft are not equipped with encoding altimeters, and their reply to an altitude interrogation is all zeros. Not all aircraft are equipped with transponders, and unequipped aircraft are not seen by the TCAS equipment.

From the measured range and encoded altitude in the replies TCAS derives range rate and altitude rate, and begins to track the other aircraft. If the track indicates that the other aircraft will pass too close to own aircraft within a short period of time (approximately 30 seconds, but this depends on the setting of the pilot's sensitivity level switch, own altitude, and sensitivity level commands sent from the ground) TCAS issues commands to the pilot. These may be either positive commands (climb, dive) or negative commands (don't climb, don't descend, don't climb faster than 1000 feet per minute, etc.). They are displayed as indicator lights on a modified Instantaneous Vertical Speed Indicator (IVSI). Positive commands light climb/dive arrows. Negative commands light yellow indicators around the periphery of the IVSI that act as speed limits. The pilot is advised to keep the IVSI needle out of the yellow regions.

TCAS units equipped to measure the bearing to other aircraft (also called the angle-of-arrival of the transponder reply, or AOA) can generate traffic advisory displays in a plan-view format. The aircraft displayed are normally limited to those within 5-10 nmi of own aircraft and in a band of altitudes within a few thousand feet of own altitude. This capability assists pilots in visually acquiring nearby aircraft. On the basis of visual information, they may then maneuver to avoid the need for collision avoidance commands, and may maneuver horizontally. The traffic advisory display also allows a pilot to better understand the local traffic situation, and therefore any commands that might be issued by the TCAS unit. It is able to display the bearing and range to non-altitude-reporting aircraft. TCAS cannot generate commands to avoid these aircraft due to the lack of altitude information, but can assist the pilot in visual acquisition if they happen to be nearly co-altitude.

The interrogation/reply link is complicated by several conditions. Both the interrogations and replies may be reflected from the ground. This is termed "multipath". It can cause a number of effects, including self-garbling of replies and false targets. Another problem with the link is that replies may overlap or garble. Garbling may be due to two or more aircraft being at the same range (synchronous garble) or due to replies which are responses to other interrogators, including ground interrogators. This is asynchronous garble, more usually called "fruit".

To divide up the ATCRBS aircraft population and reduce synchronous garble, especially when the TCAS unit uses an omni-directional antenna as do the TEU's, a technique called "whisper-shout" is used. Instead of a single interrogation, a series of interrogations are sent at increasing power levels. Each interrogation except the first is immediately preceeded by a suppression pulse pair spaced 2 μsec apart at a slightly lower (1-3 dB) power than the preceding interrogation. This suppression pulse pair causes ATCRBS transponders not to reply. The effect of this sequence is that the ATCRBS population is divided up based on link power margin. An ATCRBS transponder will typically reply to only two or three interrogations in the sequence. The lower powered interrogations will not be seen (will be below the transponder's threshold level). The suppression pulse pair preceeding the higher powered interrogations will suppress the transponder. At intermediate powers, the interrogation pulse pair will be above the transponder threshold but the suppression pair will be below threshold and the transponder will reply.

Multipath and fruit effects are eliminated by both hardware and software techniques. The primary hardware technique is a Dynamic Minimum Threshold Level (DMTL) which adjusts the receiver threshold such that the current reply is 6-9 dB above threshold. Multipath and fruit replies will usually be below this level and will not be seen by the receiver. Stronger fruit replies may capture the DMTL, however, and cause desired replies to be lost. This happens rarely, however, and the software tracking algorithms easily track through such misses.

Software techniques for dealing with multipath, fruit and other link problems are described in Ref. 1.

Mode S transponders are not affected by synchronous garble. Each transponder is interrogated individually by name (Mode S address), in contrast to the anonymous interrogations of ATCRBS. However, this introduces the problem of initially finding out what Mode S addresses are in the vicinity. This is done by passively listening to Mode S replies during times when neither ATCRBS nor Mode S surveillance is being performed. These replies may be replies to ground interrogators, replies to other TCAS aircraft or may be Mode S "squitter" replies which the transponder is required to emit once per second if it is not being interrogated.

For a more detailed TCAS summary see Ref. 2. References 3 and 4 give detailed technical discussions of the TCAS system design and validation. The ATCRBS surveillance algorithms used in the TEU correspond closely to those in Ref. 1, although that document contains some improved methods for tracking non-altitude-reporting aircraft and some other improvements that have not been incorporated in the flight software as of this date. The Mode S software demonstrated in the TEU corresponds, with the exception of recent refinements, to the software described in Ref. 5. Reference 6, Volume 2 describes the collision avoidance algorithms currently specified for TCAS. The algorithms in use in the TEU's as of this date are the earlier ones of Ref. 7 plus certain revisions made as flight experience was gained. References 6 and 8 are the standards developed for commercial TCAS installations.

2.0 TCAS EQUIPMENT

2.1 Major System Elements

The major components of a TCAS Experimental Unit (TEU) are shown in Fig. 2.1-1. These components are:

- A transmitter/receiver unit which generates interrogations at 1030 MHz and receives replies at 1090 MHz. The transmitter/receiver unit is capable, under computer control, of using either an antenna mounted on top of the aircraft or one mounted on the bottom. A reply log video signal is produced which is passed to the video pulse quantizer (VPQ) in the digital processing chassis. In the version of the TEU which provides bearing information (i.e., angle-of-arrival data) the receiver processes two signals from each antenna. The reply bearing information is contained in the relative phase of these two signals. In addition to log video, the AOA receiver produces two outputs called SINE and COSINE which are passed to the angle processor in the digital processing chassis.

- A digital processing chassis. As shown in Fig. 2.1-2, this contains:

  - A modulation control unit (MCU) to control the transmitter and reply processors.

  - A video pulse quantizer (VPQ) and video digitizer which process the log video returned from the receiver and provide several quantized signals to the reply processors.

  - An ATCRBS reply processor.

  - A Mode S reply processor.

  - An optional Mode S transponder interface, which provides for air-air and air-ground communication using the Mode S data link.

  - An optional angle-of-arrival (AOA) processor. This replaces the Mode S transponder interface (they are mutually exclusive due only to card slot limitations). In conjunction with an appropriate AOA receiver and antennas this provides the reply processors with information on the direction from which a reply was received.

  - A General Purpose I/O (GPIO) interface. This provides a data channel or direct memory access interface (i.e., one where data transfers take place independently of the software, once initiated) for the MCU, reply processors, and Mode S transponder

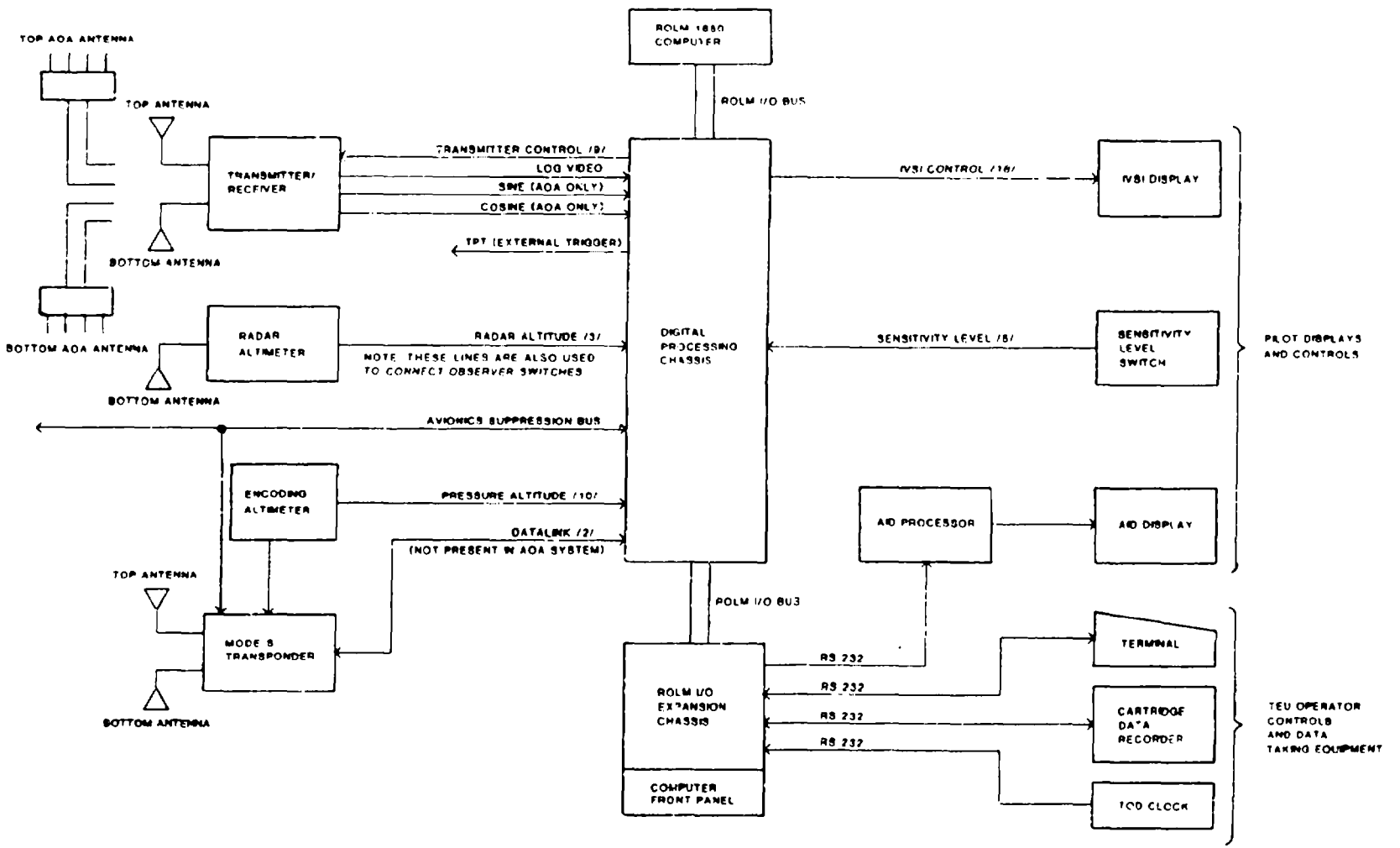


FIG. 2. 1 -1 MAJOR TEU COMPONENTS

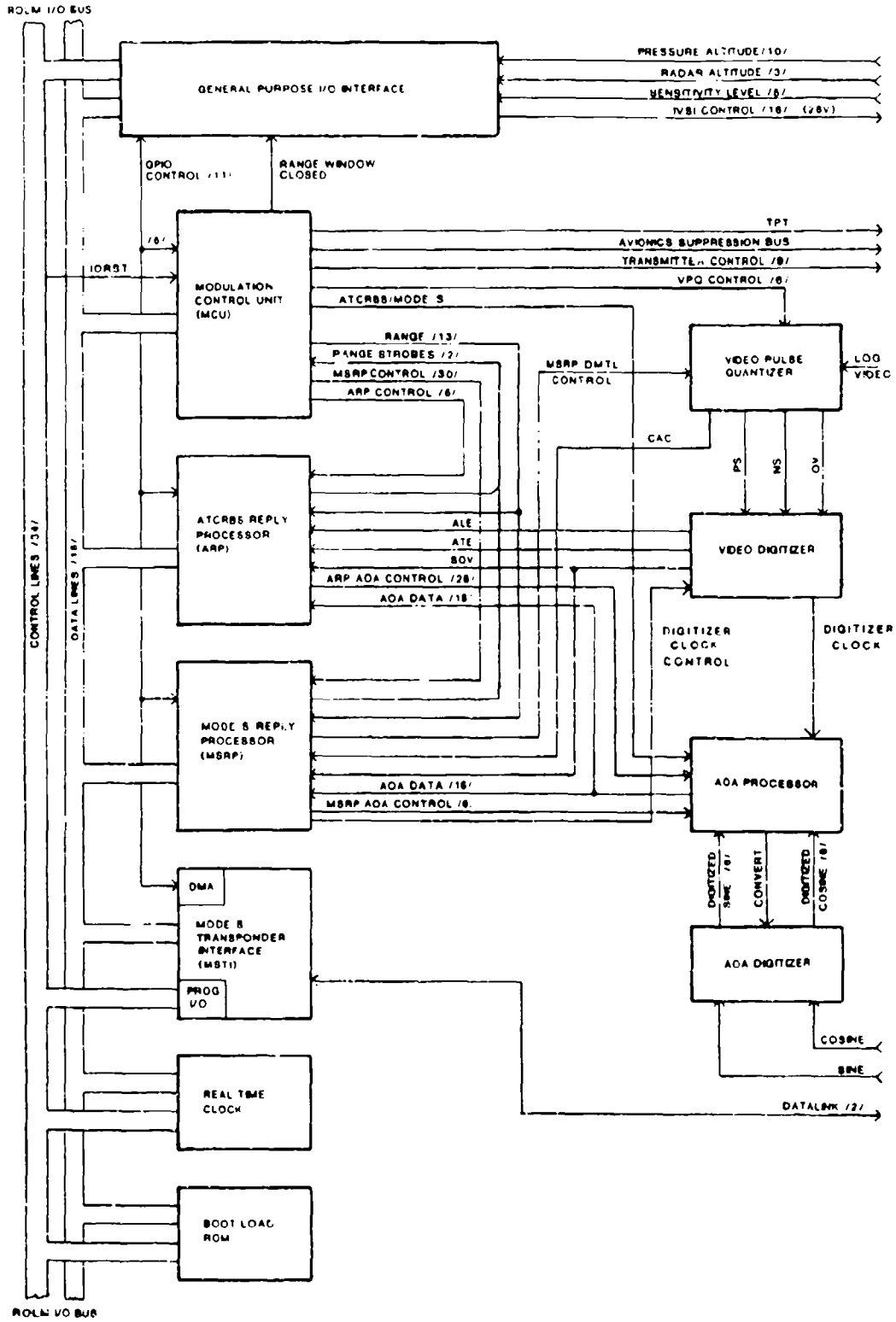


FIG 2.1-2 DIGITAL PROCESSING CHASSIS COMPONENTS AND INTERCONNECTIONS

interface. It also provides a programmed I/O interface (which requires explicit programming to transfer each word) for the encoding altimeter, the radar altimeter, the sensitivity level switch, and the IVSI display.

- A real-time clock

- A boot load ROM for initial system loading from a tape cartridge.

- An encoding altimeter, which provides altitude data both to the TEU processor and to the transponder.

- A Mode S transponder, which provides surveillance replies to both ground interrogators and other TCAS interrogators and also provides data link capabilities which are used for air-air and air-ground communication.

- An Instantaneous Vertical Speed Indicator (IVSI) modified to display collision avoidance commands.

- A sensitivity level control switch, which allows the pilot to control the "sensitivity" of the TEU by selecting among several settings of parameters in the collision avoidance logic.

- A microprocessor controlled CRT display, the Airborne Intelligent Display (AID), used to display traffic advisories to the pilot. Both tabular alphanumeric and plan view display formats have been used. The display is a modified color weather radar display, and remains able to perform weather radar functions. For more detail, see Ref. 12.

- Devices used for software debugging and data extraction:

  - A standard keyboard/printer.

  - A cartridge data recorder for program loading and data recording.

  - A time-of-day (TOD) clock for time tagging the extracted data.

- An optional radar altimeter.

- A ROLM 1650 computer with 32768 words of memory and a 1-μsec memory cycle time. The full TEU software system requires more than this amount of memory. Therefore, the Mode S surveillance software is normally not loaded.

- A ROLM 1650 I/O expansion chassis, with CPU front panel. This provides for the usual CPU manual controls used for debugging, as well as additional I/O interface slots for interfaces to the keyboard/printer, cartridge data recorder, time-of-day clock, and the AID.

- An auxiliary interface box (not shown in Fig. 2.1-1) which can be substituted for the I/O expansion chassis. It provides all the functions of the I/O expansion chassis except for the CPU front panel and the keyboard/printer interface. It is used with TEU's whose sole purpose is data taking and not software or hardware debugging.

The operation of the components of the transmitter/receiver and of the digital processing chassis will be discussed in the following sections. The other TEU components will be discussed only to the extent required to explain the operation of the transmitter/receiver and the digital processing chassis.

2.2 Modulation Control Unit

2.2.1 Function

The Modulation Control Unit (MCU) is the interface by which the CPU controls the transmission of interrogations and the processing of replies. The CPU transfers a block of control information to the MCU via the data channel to initiate a transmit/receive operation. The MCU then generates the appropriate control and timing signals for the transmitter, video pulse quantizer, angle-of-arrival processor, and ATCRBS and Mode S reply processors. The MCU also accepts range strobes from the ATCRBS and Mode S reply processors and uses these to latch the current range clock value for use by these processors. At the end of the interrogation listening window, it sends a range gate close pulse to the GPIO interface, triggering the range gate close interrupt. A final MCU function is to prevent the transmitter duty cycle from exceeding safe limits. It does this by ignoring any modulation requests that occur too soon after a previous transmission. A 2-msec interval is required after ATCRBS transmissions and a 16-msec interval following Mode S transmissions.

2.2.2 Programming

The MCU's operation is controlled by means of 10-word command blocks transferred from the CPU via the GPIO data channel interface. GPIO data channel programming is discussed in greater detail in Section 2.9. The software must first determine whether the MCU is ready to receive a command. This is done by reading the GPIO status via a DIB instruction. Note that the MCU does not have to be the selected GPIO device in order to read the MCU status. The status word returned contains two MCU status bits. If the MCU Ready (MR) bit (bit 1) is 1, the MCU is ready to receive a command block. However, if the command block specifies an interrogation, as opposed to a squitter listening function, the Modulation Enabled (ME) bit (bit 0) must also be 1 or else the MCU will not modulate the transmitter.

Before transferring the command block, any GPIO data channel transfer in progress must be terminated and the GPIO device address must be set to route data channel transfers to the MCU. Then the data channel buffer address and word count are initialized and a START pulse is issued to initiate the transfer. The software must then wait until the transfer is complete. This is signalled by the setting of DONE, clearing of BUSY, and the generation of an interrupt from device 7. The I/O skip instructions can be used to test

BUSY or DONE, or the interrupt can be used to trigger further processing. Given the short transfer time, and the fact that the GPIO channel must be reallocated immediately to one of the reply processors, it is usually best for the software to idle using an I/O skip instruction and with the device 7 interrupt disabled until the transfer is complete.

The format of the 10-word command block is shown in Fig. 2.2-1, along with an explanation of each of the fields.

2.2.3 Operation

2.2.3.1 Summary

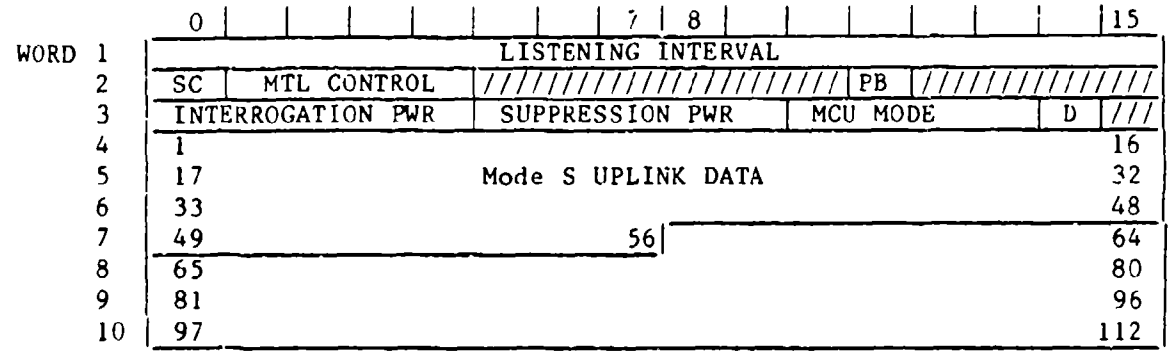
An overall MCU block diagram is provided by Fig. 2.2-2. When the GPIO data transfer is started by the software, this is signalled over the BUSY line to the MCU controller. The controller then goes through a series of states which transfer the 10 command words from the CPU memory and load them into appropriate MCU registers. The listening window (first word) is loaded into the range counter, the second and third words into the two control registers, and words 4 through 10 are loaded into the Mode S data buffer memory.

When the command block transfer is completed the MCU controller issues a START pulse. This initiates the MCU command processing as well as being sent to the ATCRBS and Mode S reply processors. The START pulse initializes the range counter, which begins incrementing once every 125 ns. The range window counter also starts incrementing at this rate starting from the initial listening window value. Finally, the mode decoder and sequence generator also is initialized and its interval timing counter begins incrementing at the same 8-MHz rate.

The control registers hold all the data regarding the nature of the interrogation and the type of reply processing required. Some of this data directly affects control lines to other devices. The mode information is decoded to create other external control signals and also to control the sequence generator, which provides time-varying signals. For Mode S interrogations, the sequence generator in turn controls the Mode S data buffer and DPSK generator which generates the DPSK data modulation signal for the transmitter and provides the MODE S ADDRESS data to the Mode S reply processor.

The suppression logic places the AVIONICS SUPPRESSION BUS in the suppressed state (+28 volts) at the START pulse of any interrogation. Squitter listening mode does not cause suppression. The bus is held in the suppressed state until either the interrogation is completed or the listening window closes. Selection between these two modes is done by software via the SC bit of the MCU control block (Fig. 2.2-1).

When the range window counter counts up to zero, the RANGE WINDOW closes and the RANGE WINDOW CLOSED pulse is sent to the GPIO interface, causing an interrupt of the CPU. This terminates reply processing. However, the range



LISTENING INTERVAL — Receiver listening interval in 2's complement format. LSB increment = 125 ns. The listening window must be set longer than the nominal listening range to account for various hardware delays. An additional 87 μsec should be allowed in ATCRBS modes, and an additional 230 μsec in Mode S.

SC — Suppression control.

- 0 = the external avionics suppression line will be held in the "suppress" (+28V) state for the duration of the listening window.
- 1 = the external avionics suppression line will be held in the "suppress" state from TPT to the end of the interrogation (approximately 30 μsec).

MTL CONTROL — Minimum threshold level control for the VPQ. LSB increment = 2 dB.

0 = -77 dBm referred to the TEU
15 = -47 dBm antenna input port

PB — Phantom rejection control for the ATCRBS reply processor. A phantom is a false reply created due to the accidental spacing of code and/or framing pulses of two real replies.

0 = Accept all replies
1 = Reject all phantoms

Fig. 2.2-1. MCU control block format.

DEFINITION OF FIELDS (Continued)

FIELD	SIGNIFICANCE
INTERROGATION PWR	Transmitter power attenuator control for the P1-P3-P4 interrogation pulses. LSB increment = 1 dB. 0 = 31-dB Attenuation 31 = 0-dB Attenuation. The nominal power at the transmitter output connector is +57 dBm in this case.
SUPPRESSION PWR	Transmitter power attenuator control for S1 S2 suppression pulse pair. LSB increment = 1 dB. 0 = 31-dB Attenuation 31 = 0-dB Attenuation
MCU MODE	MCU mode select. 0 0 0 0 — Mode A, P4, no suppression 0 0 0 1 — Mode C, P4, no suppression 0 0 1 0 — Mode A, P4, suppression 0 0 1 1 — Mode C, P4, suppression 0 1 0 0 — Mode A, No P4, no suppression 0 1 0 1 — Mode C, No P4, no suppression 0 1 1 0 — Mode A, No P4, suppression 0 1 1 1 — Mode C, No P4, suppression 1 0 0 0 — Long Mode S 1 0 0 1 — Short Mode S 1 0 1 0 — Squitter Listening 1 0 1 1 — Unused 1 1 0 0 — Unused 1 1 0 1 — Unused 1 1 1 0 — Unused 1 1 1 1 — Terminate squitter listening
D	Diversity Switch Control 1 = Top Antenna 0 = Bottom Antenna
MODE S UPLINK DATA	Data and address for the Mode S modes. The last 56 bits of this field are ignored for short Mode S transmissions. This entire field is ignored for ATCRBS and squitter listening modes but it must be supplied.

Fig. 2.2-1. MCU control block format (Cont'd).

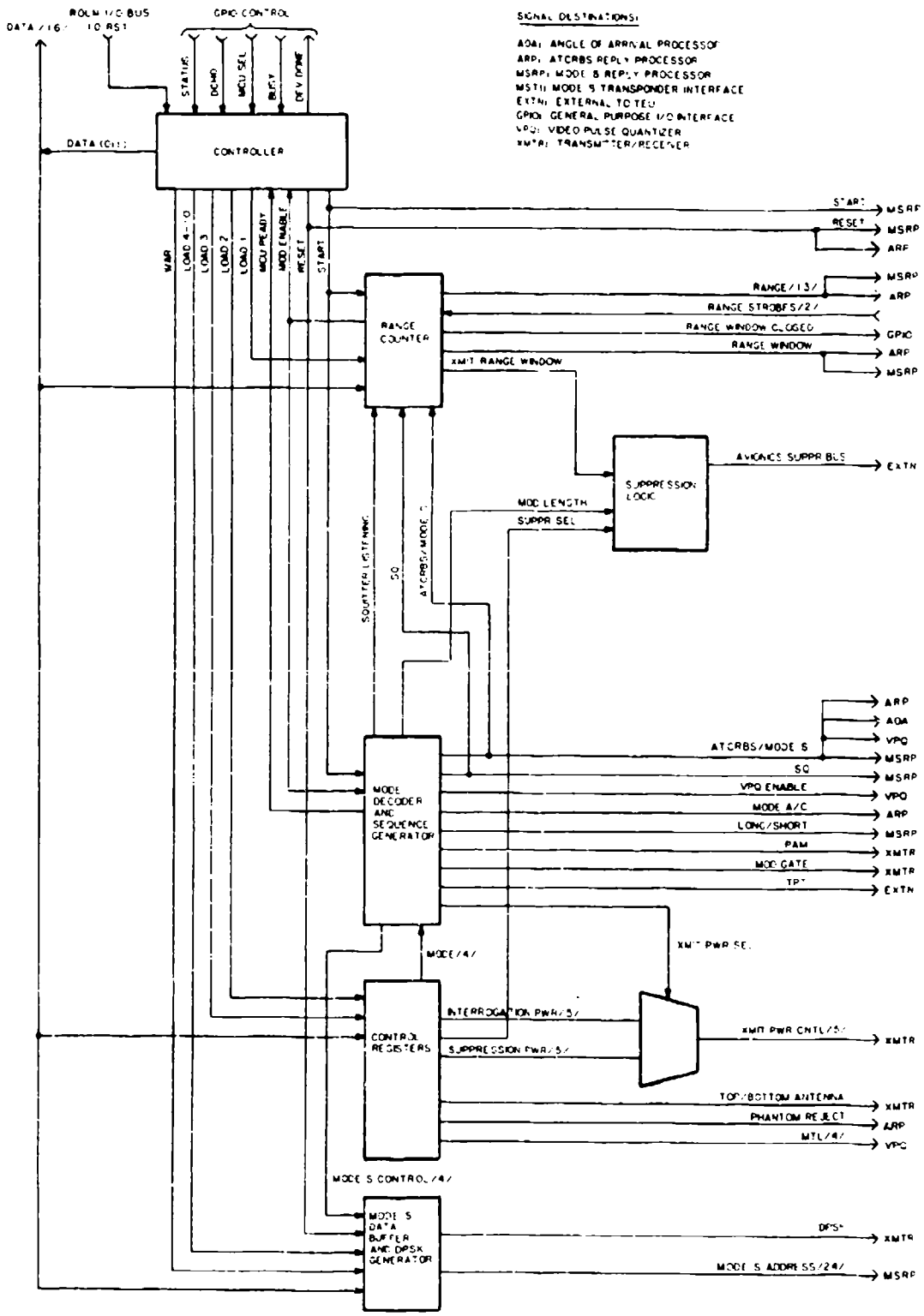


FIG. 2.2-2. MCU BLOCK DIAGRAM

counter continues to count, timing out either 2 msec (for an ATCRBS interrogation) or 16 msec (for a Mode S interrogation) at which point MODULATION ENABLED is asserted. At that point both MCU status bits will indicate that the MCU is ready for another interrogation command.

When the command block indicates squitter listening mode, a slightly different procedure is followed. The sequence generator is not activated, since no transmission is sent. Similarly, the range and range window counters are not started. The RANGE WINDOW is opened at the START pulse. The ATCRBS/Mode S line indicates Mode S, but the SQ line indicates to the Mode S reply processor that squitter reply processing is desired. This disables the Mode S address checking logic in the Mode S reply processor. The AVIONICS SUPPRESSION BUS is left in the unsuppressed state.

In squitter mode the RANGE WINDOW remains open, as the range window counter is not incrementing. Therefore, to stop squitter listening and close the range window a new MCU control block must be output by the software with the "stop squitter listening" mode. This must be done prior to sending any command block that generates an interrogation. This also allows the Mode S reply processor to terminate properly.

2.2.3.2 MCU Controller

The MCU controller is a standard state controller. The outputs and the next state are determined by the inputs and the current state. A PROM provides the translation from input/current state to output/next state. A register is used to latch the next state. A block diagram of the controller is shown in Fig. 2.2-3 and a state diagram for the controller is shown in Fig. 2.2-4. At power-on, the controller is initialized to state 0, from which it immediately passes to state 1 after issuing a RESET. It waits in state 1 until the conditions for a data channel transfer are satisfied. It then issues a RESET and begins the transfer. In state 3 it issues the DEVICE DONE pulse to request a word transfer from the GPIO interface. It waits in state 4 until the resulting DCHO signal is received, at which point it enters state 5 to read the data. It then returns to state 3 for the next word, or, if the last word has been transferred, it goes to state 6 which issues the START pulse to initiate the interrogation. An auxiliary word count register is used by the controller to route the data channel words to the appropriate registers. This is incremented in state 3. Note that the WRITE signal generated in state 5 does not control the loading of registers. Its only function is to increment the Mode S data buffer's write address register at the appropriate times. As can be seen in Fig. 2.2-3, WRITE is gated by LOAD4-10 to produce the WAR signal that increments the address counter.

2.2.3.3 Mode Decoder and Sequence Generator

Figure 2.2-5 is a block diagram of the mode decoder and sequence generator. The mode decoder is a PROM which generates control signals based on the current mode value as received from the MCU control registers. The sequence generator consists of a sequence counter, some PROMs which store the various timing signals, multiplexers for selecting the desired signals, and

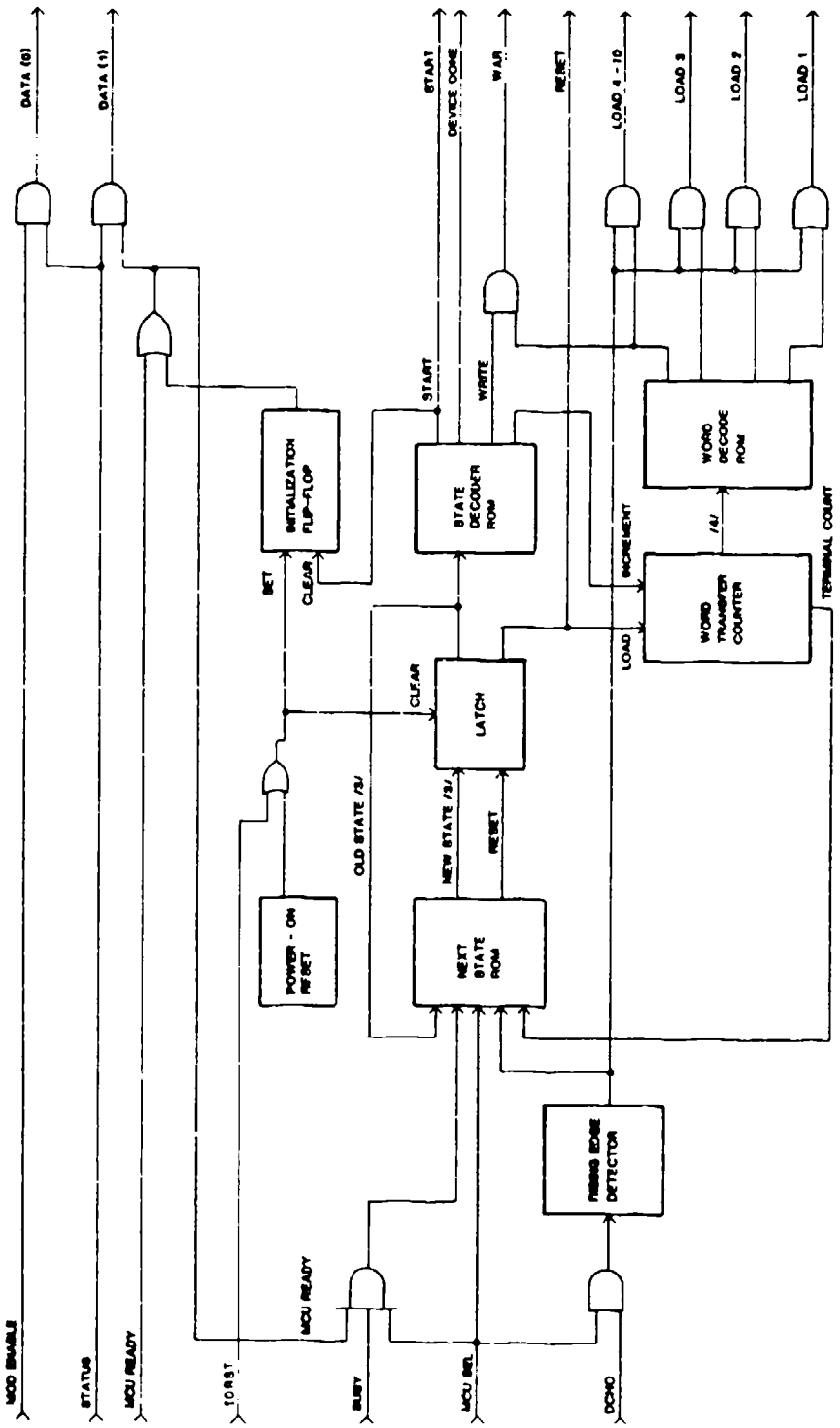


FIG. 2.2-4. MDU CONTROLLER BLOCK DIAGRAM

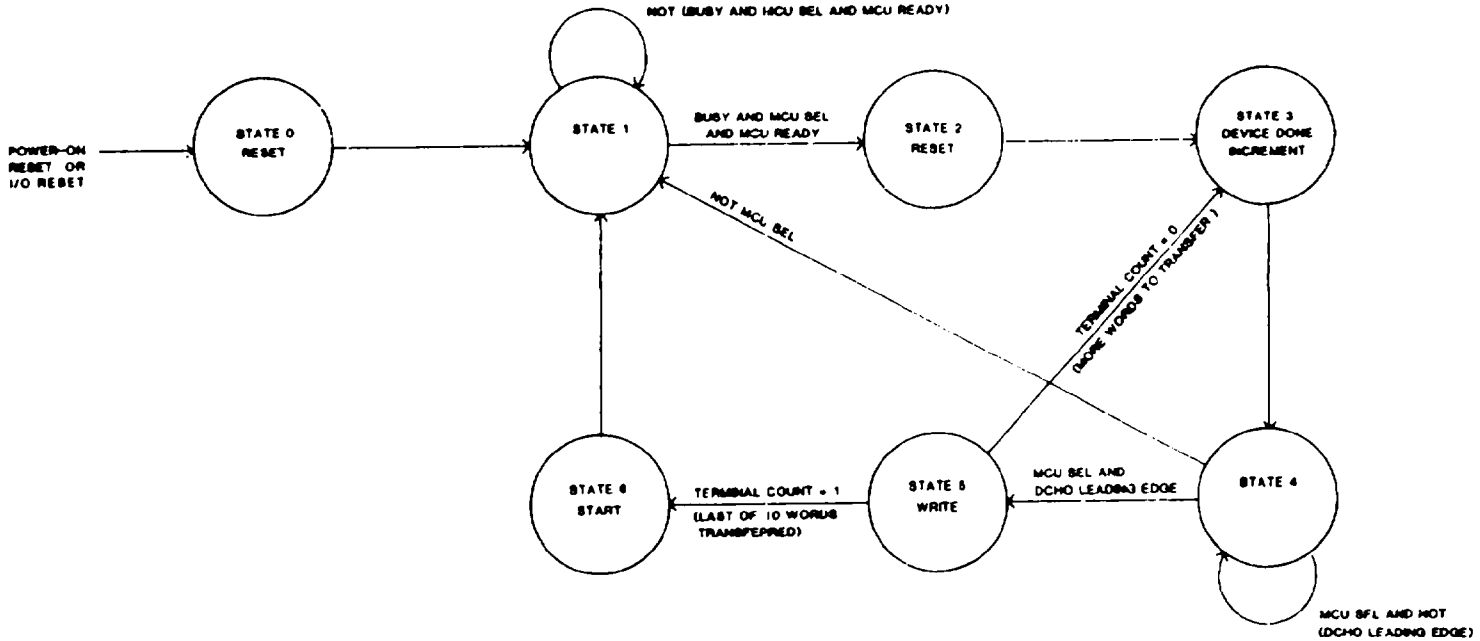


FIG. 2. 2 -4. MCU CONTROLLER STATE DIAGRAM

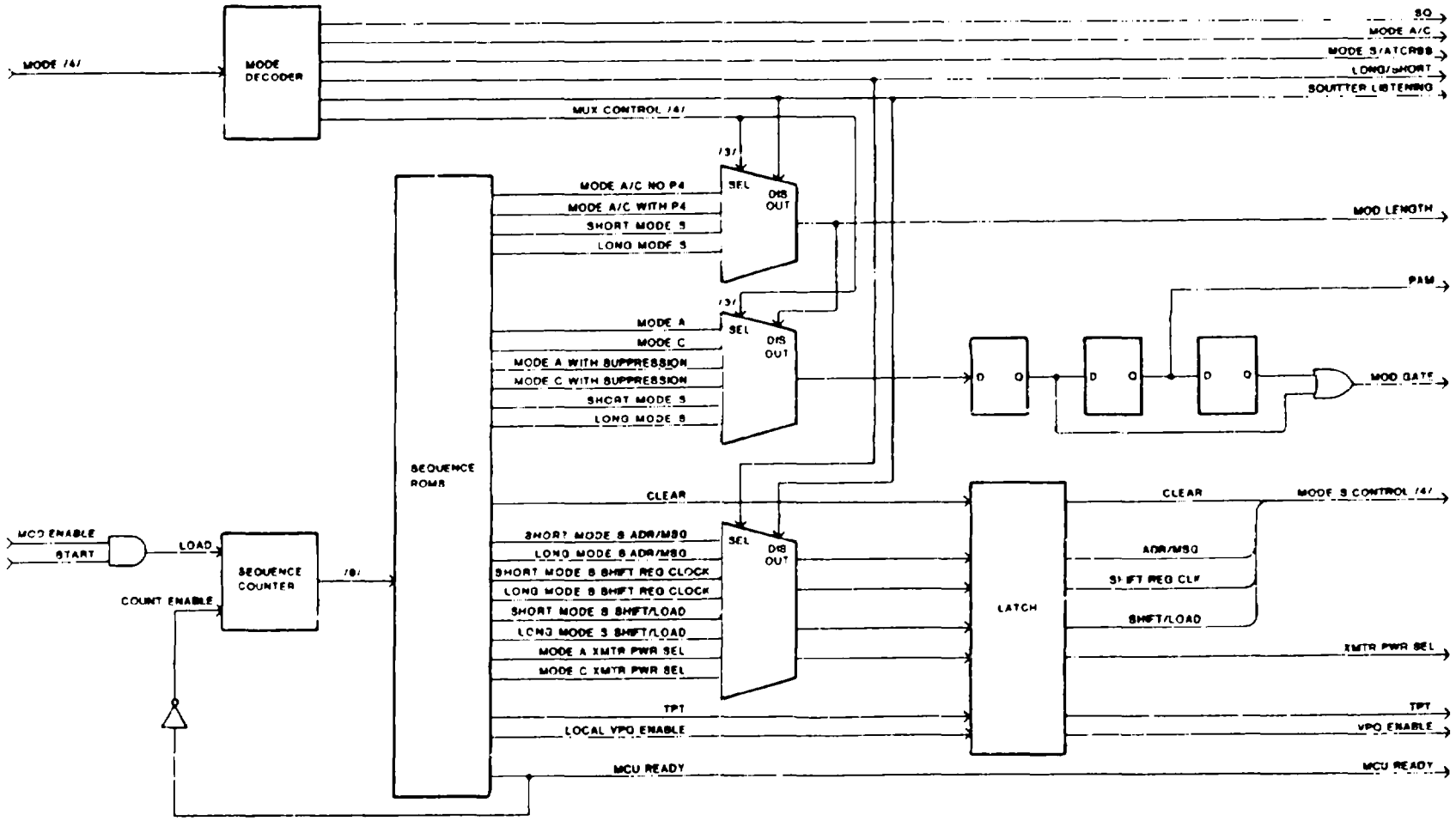


FIG. 2.2-6. MCU MODE DECODER AND SEQUENCE GENERATOR

latches for synchronizing and holding the outputs. The sequence counter is loaded at the START pulse, if modulation of the transmitter is allowed. It counts at an 8-MHz rate through a 64-μsec sequence. At the end of this time, the MCU READY signal becomes true, and disables the counter. The counter output is used to address the PROMs, which generate the signals shown. Particular signals are selected by the multiplexers based on control signals from the mode decoder. The MOD LENGTH signal determines the length of the PAM modulation, and therefore whether or not the P4 pulse, present in all the ATCRBS PAM outputs from the PROMs, will be output or inhibited at the multiplexer. The MOD GATE signal is required by the transmitter before modulation is initiated. The circuit shown assures that MOD GATE goes high one clock before PAM, and stays high until one clock after PAM goes low.

Figure 2.2-6 shows the timing of the various PROM outputs. The timing shown is at the output of the sequence PROMs. The relative timing of some signals changes before their point of use due to differing numbers of delaying flip-flops along their paths.

2.2.3.4 Mode S Data Buffer and DPSK Generator

Figure 2.2-7 is a block diagram of this portion of the MCU. When the MCU controller initiates the transfer of a 10-word command block over the data channel, it pulses the RESET line, setting the buffer address register to its initial state. As words 4 through 10 of the command block are transferred, LOAD4-10 is pulsed by the MCU controller as each word is available on the DATA lines. After each LOAD4-10 pulse, a WAR pulse is sent which allows the buffer address register to increment by one. In this way, all the Mode S data bits are loaded into the buffer memory. These words are loaded into the buffer memory whether or not the control block's mode field indicates a Mode S interrogation.

Next, unless the mode field of the MCU control block selected squitter listening, the sequence generator will supply the various control signals for the DPSK modulation. See Fig. 2.2-6 for the timing of these signals. The sequence begins with the arrival of simultaneous CLEAR and SHIFT REG CLK pulses. CLEAR puts the output of the Adr/Msg multiplexer in a high impedance state, allowing the pull-up resistor on the output to drive the input to the toggle flip-flop high. SHIFT REG CLK is never used directly as a clocking signal, but rather always enables either the system's 8-MHz CLOCK signal, or its complement $\overline{\text{CLOCK}}$. In this case, it allows the toggle flip-flop to clock, toggling its output and causing a phase shift. This is the sync phase reversal of the Mode S DPSK modulation. The analog timing adjustment delay is used to precisely align the DPSK phase shifts relative to the PAM modulation.

The CLEAR pulse also resets the buffer address register to its initial value and, after a one clock delay, clears the tapped shift register used for the Mode S parity encoding function.

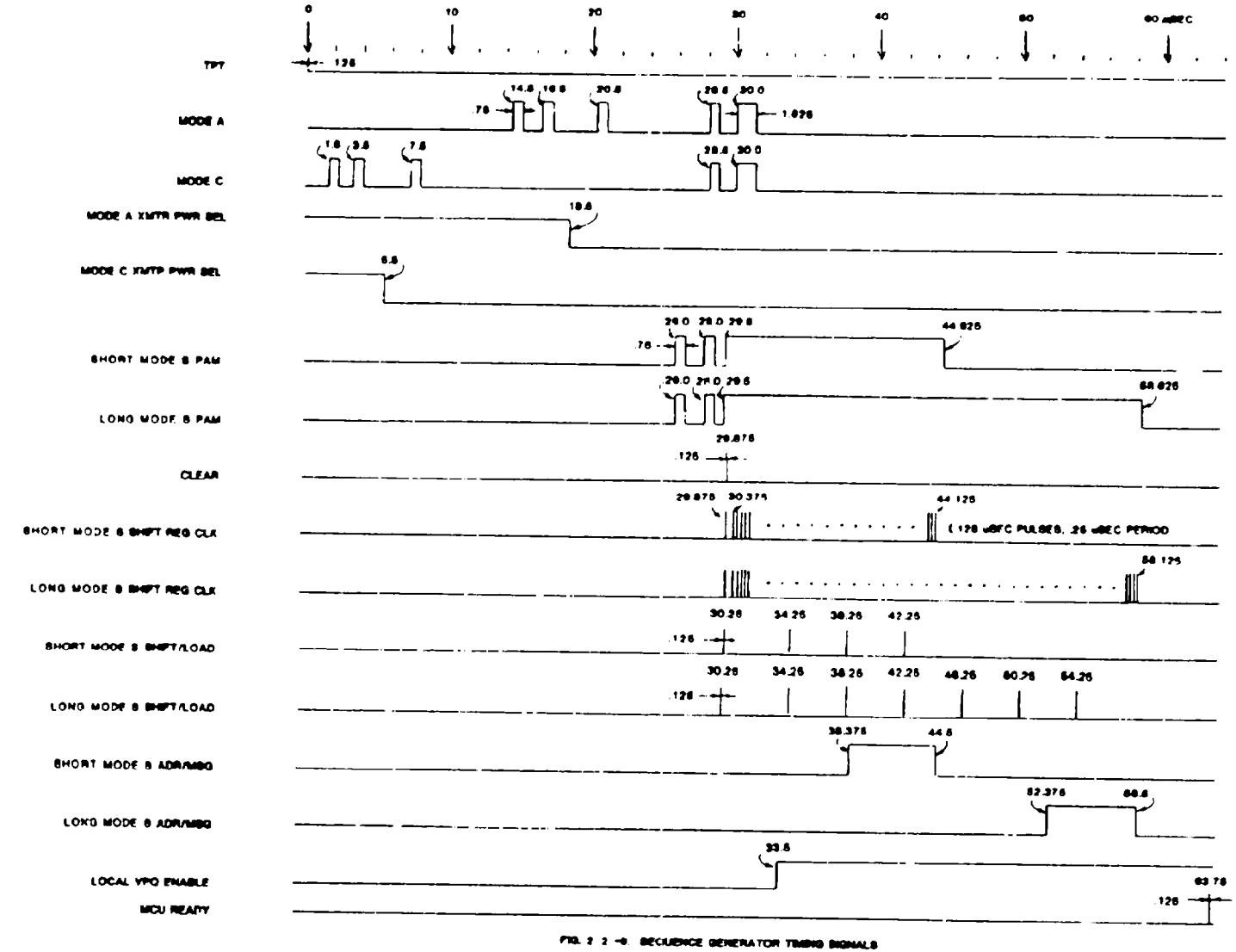


FIG. 2 2 -9. SEQUENCE GENERATOR TIMING SIGNALS

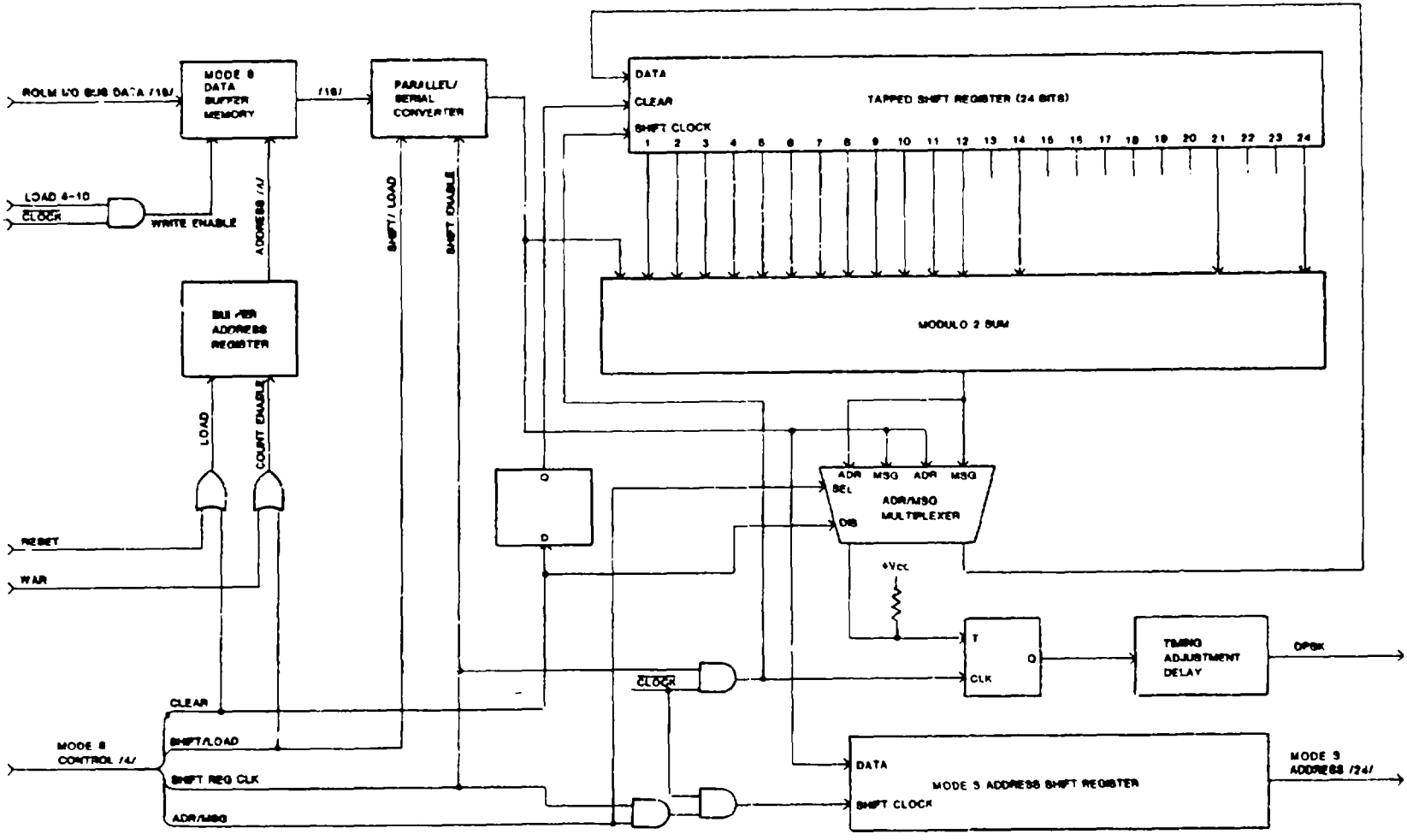


FIG. 2.2-7. MODE 5 DATA BUFFER AND DPSK GENERATOR

The SHIFT/LOAD and SHIFT REG CLK signals then control the operation of the parallel/serial converter, the parity generation circuitry, the Mode S address shift register and the DPSK toggle flip-flop in a straightforward manner. During the message part of the Mode S transmission (the first 32 bits for a short message, or 88 bits for a long Mode S interrogation) the output of the parallel-to-serial converter directly drives the DPSK toggle flip-flop. It also feeds into the parity encoding process as one of the inputs of the modulo-2 summing circuit, the output of which feeds into the tapped shift register.

During the address portion of the transmission, the DPSK toggle flip-flop is driven by the output of the modulo-2 summing circuit, which provides the encoded Mode S address overlaid (modulo-2 summed with) the parity code developed during the message portion of the transmission. The output of the parallel-to-serial converter is directly routed to the input of the tapped shift register.

For detailed information on the design of the Mode S uplink message encoder and on the properties of the code, please refer to Ref. 10.

During the address portion of the message, the output of the parallel-to-serial converter is also shifted directly into the Mode S address register. It is held there until the next MCU transmission. The Mode S ADDRESS output is used by the Mode S reply processor to determine if a reply is from the interrogated transponder.

Note that the DPSK modulation signal is generated for any MCU control block except for those specifying squitter listening. For ATCRBS interrogations, however, the PAM signal to the transmitter is not present (except possibly for the P4 pulse) during the DPSK modulation, and so no waveform is transmitted. DPSK modulation during the P4 pulse does not affect any of the uses of the P4 pulse.

2.2.3.5 Range Counter

The range counter circuitry (Fig. 2.2-8) actually involves two counters. The first provides the system's 8-MHz range clock, and the second counts out the range window specified by the software in the first word of the MCU control block.

The range window counter is loaded from the data bus at the LOAD-1 signal generated by the MCU controller. The value loaded is the two's complement of the desired range window interval, and has an LSB of 125 nsec. On receipt of a START pulse while the MCU mode is other than SQUITTER LISTENING, the two J-K flip-flops will be cleared. In addition, if MOD ENABLE is true, the range clock is initialized. The initial value loaded depends on whether an ATCRBS

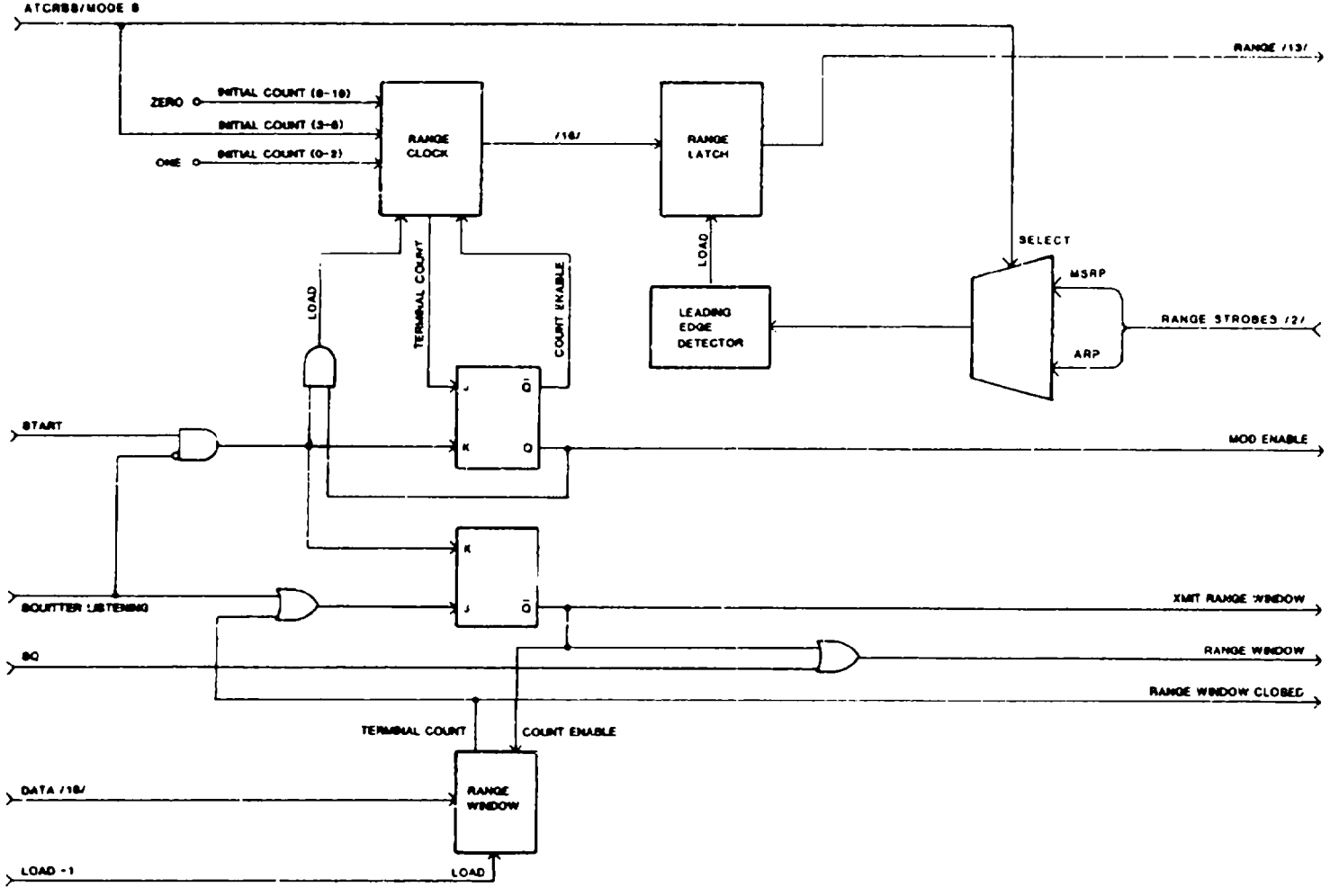


FIG. 2.2-8. RANGE COUNTER

or a Mode S transmission is being sent. The values loaded are such that the range clock overflows 2 msec after the START pulse for an ATCRBS interrogation and 16 msec after the START pulse for a Mode S interrogation.

The clearing of the flip-flops opens the RANGE WINDOW. This also enables the range window counter. When this counter overflows, it sets its associated flip-flop, thereby closing the RANGE WINDOW. It also generates the RANGE WINDOW CLOSED pulse which goes to the GPIO circuitry and triggers an interrupt of the computer.

In the case of squitter listening, the flip-flops do not clear at the START pulse. Instead, the SQ signal opens the RANGE WINDOW. This signal must be removed, by issuing a "Stop Squitter Listening" command block to the MCU, in order to close the RANGE WINDOW.

The least significant bits of the range clock are sampled into the range latch when an appropriate RANGE STROBE arrives. Those bits are all 0 at the START pulse, and count time in 125-nsec increments from that pulse.

2.2.3.6 Suppression Logic

Figure 2.2-9 shows the avionics suppression circuit. The SUPPRESSION SELECT line, which is controlled via the SC bit in the MCU control block, determines whether the MOD LENGTH or XMIT RANGE WINDOW signal drives the suppression bus. If MOD LENGTH is selected, suppression ends at the end of the interrogation. If XMIT RANGE WINDOW is selected, suppression lasts for the entire listening window.

While these were the two options supplied in the TEU for experimental purposes, it has since been determined that an intermediate suppression length is best. Suppressing the other on-board avionics serves two purposes:

a) It prevents the Mode S transponder from replying to its own ship's TCAS interrogations by suppressing the transponder during the interrogation.

b) It prevents the transponder from replying to interrogations reflected from the ground after transmission from own TCAS. This is accomplished by extending the suppression into the listening window. Suppressing for the entire listening window is an extreme form of this.

Suppressing for the entire listening window has the advantage that it prevents all replies from own ship's transponder during the listening window, and so eliminates this powerful interference source. However, in an operational TCAS it would unacceptably reduce the transponder's round reliability as seen by ground interrogators. Therefore a shorter suppression interval has been chosen for operational TCAS systems that prevents responses as in (a) and (b) above, but does allow for the possibility that the transponder might reply to a ground interrogator such that it interferes with a TCAS listening interval.

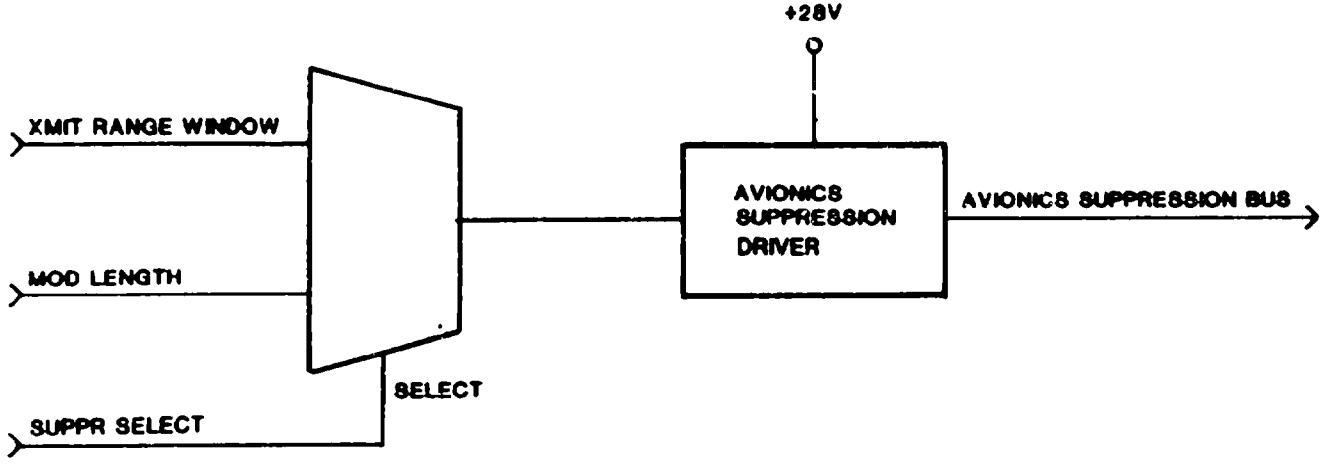


FIG. 2. 2 -9. SUPPRESSION LOGIC

2.3 Transmitter/Receiver

In this section, the TEU transmitter/receiver circuitry for a non-AOA TEU is described. The additional circuitry required for AOA measurements is discussed in Section 2.7.

The TEU transmitter/receiver, Fig. 2.3-1, contains the following functional subunits:

a) Microwave signal source
b) Transmitter chain
c) Antenna interface and transmitter monitor
d) Receiver chain

All of the power supplies necessary to power the RF unit are housed in the TEU Digital Processor assembly.

2.3.1 Microwave Signal Source (1030 MHz)

A phase-locked, temperature controlled crystal oscillator, stabilized to within ± 0.0005% of the nominal frequency, and a solid state multiplier chain generate the L-band transmitted carrier frequency. The output power level of the oscillator multiplier chain is high enough to permit the insertion of an L-band circulator and a 10-dB pad between the generator and the output ports to the transmitter and receiver. The isolation, and the use of an absorptive rather than a reflective pulse amplitude modulation (PAM) switch, provide the carrier frequency stability required by the TCAS National Standard.

The microwave source is enclosed in a shielded enclosure to minimize leakage. This is important because of the possibility of interference with the operation of the co-located airborne transponder whose receiver is tuned to the same frequency (1030 MHz).

The same 1030-MHz frequency source is used to drive the transmitter and to provide the receiver's local oscillator, so one-half of the microwave source output power is directed to the transmitter chain, and the other half is directed to the "L" port of the receiver's down-converter. The latter, mixed with the transponder's reply frequency (1090 MHz), generates the 60-MHz intermediate frequency (IF).

2.3.2 Transmitter Chain

The transmitter chain (Fig. 2.3-2) consists of

a. DPSK and PAM modulators
b. Microwave power module
c. Digital attenuator
d. Bandpass filter

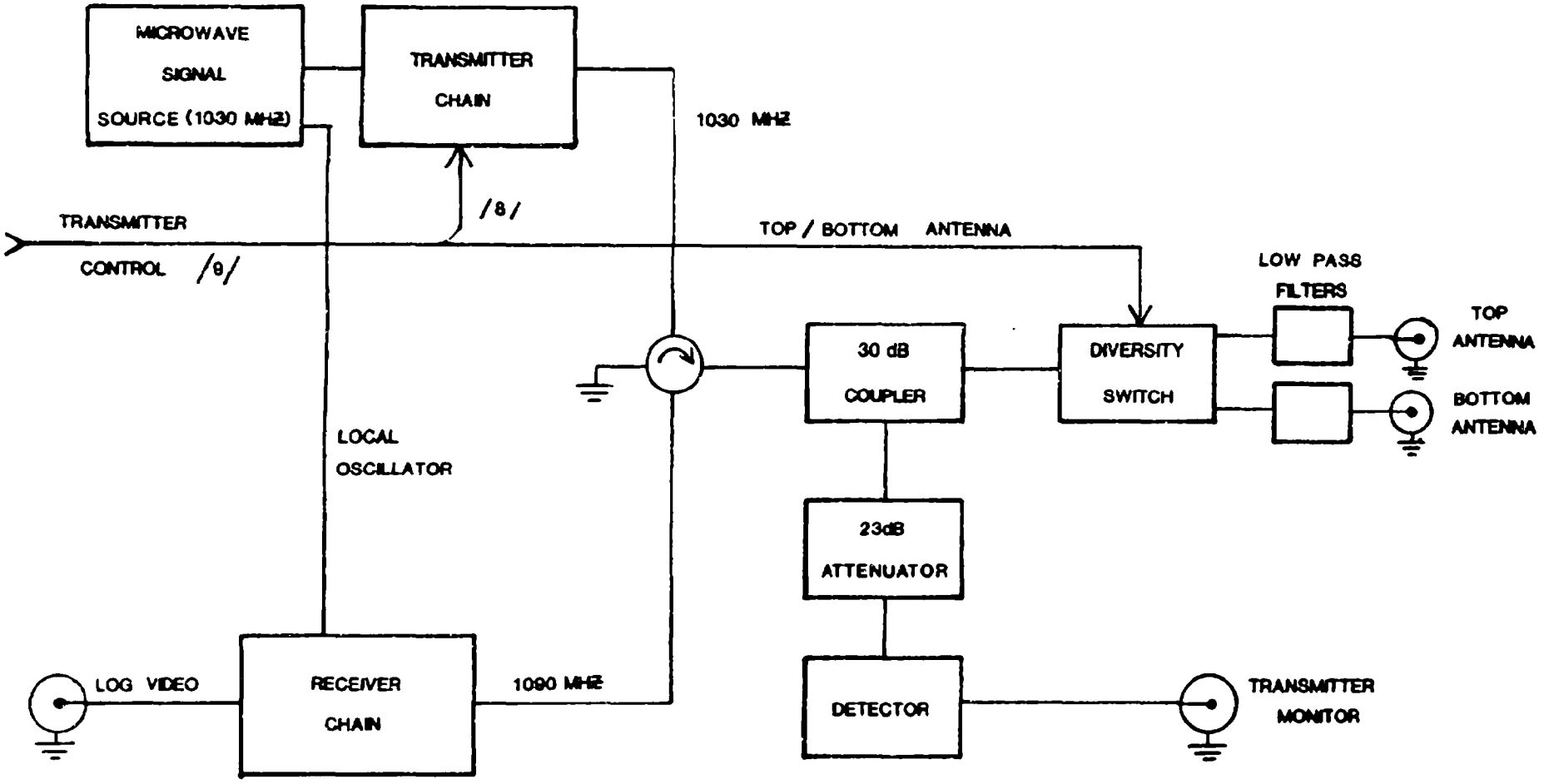


Fig. 2.3-1 TEU transmitter/receiver block diagram.

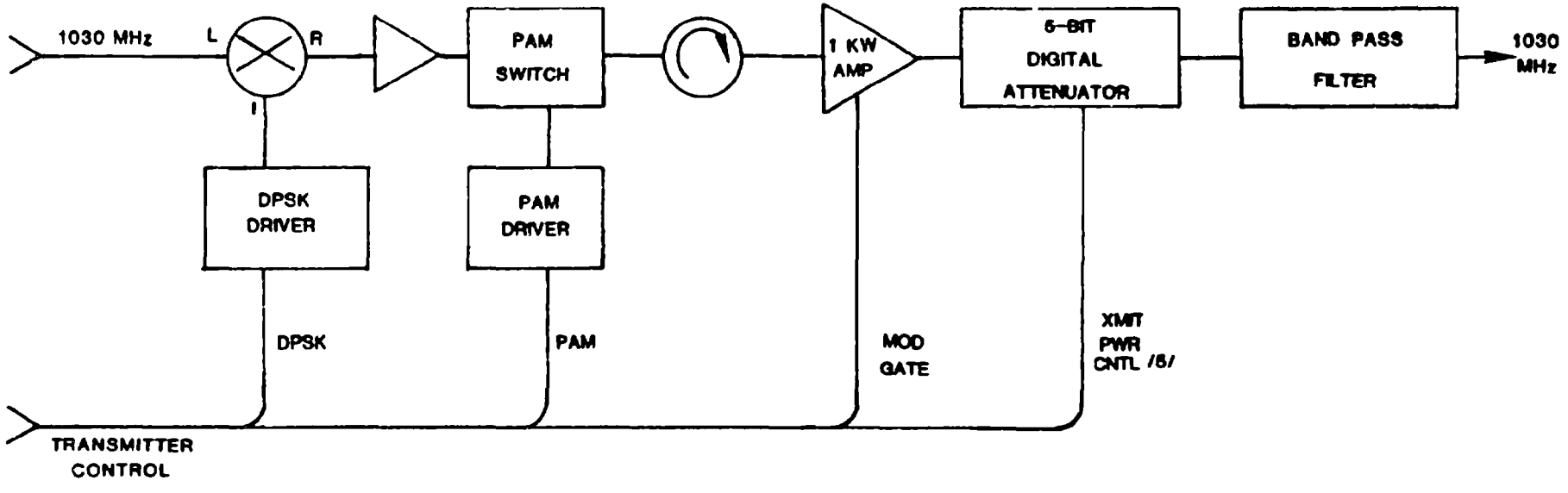


Fig. 2.3-2 Transmitter chain block diagram.

2.3.2.1 PAM/DPSK Modulators

Two types of modulators are required: a pulse amplitude modulator (PAM), and a differential phase-shift keyed (DPSK) modulator. The PAM modulator determines the duration of the transmitted microwave pulse. It consists of an absorptive type switch with an off/on ratio of 110 dB.

The DPSK modulator consists of a double balanced mixer and a video driver. The microwave signal is fed to the "L" port and a bipolar signal is fed to the "I" port. As long as the video signal magnitude is larger than zero the microwave signal is transmitted from the "L" to the "R" port. However, to minimize insertion loss, the "I" port is driven at approximately 20 mw. Every time the polarity of the video signal at the "I" port is changed the phase of the microwave signal at the "R" port is reversed.

Since the bandwidth of the double-balanced mixer is large relative to the system's bandwidth, the DPSK transition is determined by the rise-time of the driver's video pulse and by the bandwidth of the system. The TCAS National Standard requires that these transitions be completed within 80 nanoseconds, 10 degrees to 170 degrees.

2.3.2.2 Microwave Power Module

The microwave module consists of five electron tubes providing a 47-dB power gain at 1030 MHz. The first three stages are gated while the last two are RF-driven. The output power of the L-band amplifier chain can be adjusted from 1- to 2-KW peak by setting the magnitude of the high voltage, and by setting filament voltages for optimum electron emission consistent with specified tube life. The maximum microwave power required at the input of the power module is +16 dBm (40 mw), resulting in an available power gain of 47 dB.

The power module is energized by the aircraft's 28-volt source; a DC-to-DC inverter is used to provide both heater and high voltages. Because vacuum tubes have a tendency to arc, stored energy ($E = 1/2\ CV^2$) is limited by limiting the value of the capacitor to that needed to satisfy microwave pulse rise-time requirements. A pulse-modulated, current-limited, power supply provides the required energy and assures that the maximum pulse droop is within 1-dB. In the event that an arc occurs, or the pulse width is extended beyond 30-μsec, or the PRF is accidentally increased to a rate unsafe for the tubes to handle, the power supply automatically shuts off, resuming operation when safe conditions are re-established.

Typical performance of the transmitter is shown in Fig. 2.3-3 which illustrates a transition of the DPSK modulation. The upper waveform indicates the input to the 1-kW amplifier, and the lower waveform the output of the amplifier. The horizontal scale is 50 nanosecond/cm. It can be seen that the bandwidth of the transmitter is adequate to achieve the time limit allowed by the National Standard for 180° DPSK phase reversals.

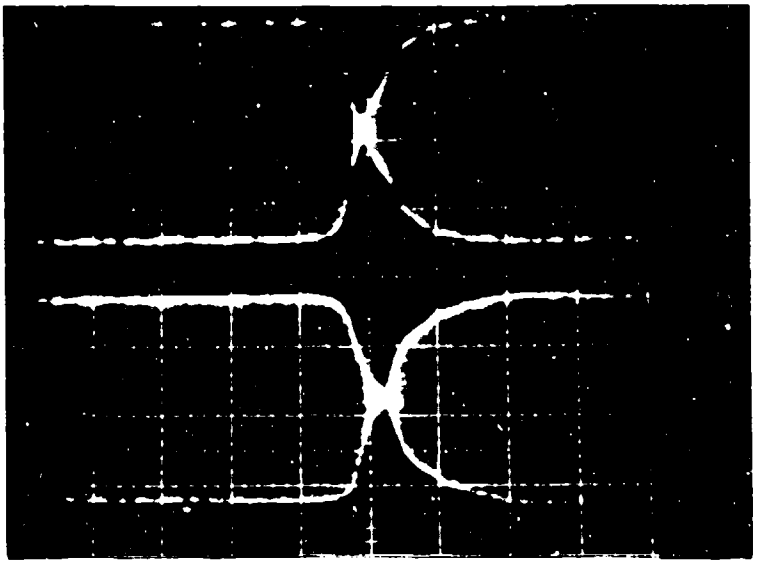


Fig. 2.3-3. Power module response to DPSK transition.

2.3.2.3 Digital Attenuator

The full output of the microwave power module is passed through a 5-bit digital attenuator, providing a least significant bit attenuation change of 1 dB and most significant bit attenuation change of 16 dB. This attenuator passes 1-Kw peak power, 2-watts average, and provides 32 values of attenuation in 1-dB steps. Maximum insertion loss is 1.5 dB; switching time is 1.0 microsecond. A 5-bit digital switch driver is provided with the attenuator.

Since the attenuator is a cold switching unit, it must not be switched while microwave energy is passing through it. The digital control system assures that this is so. In addition, an interlock is included to ensure that critical supply voltages (plus 5 volts and minus 100 volts) are present before high microwave power is applied to the attenuator. This interlock is shown in Fig. 2.3-4.

2.3.2.4 Bandpass Filter

This filter is centered at 1030 MHz, has a bandwidth of 20 MHz, an insertion loss of 0.5 dB, and provides 60-dB (with respect to band-center peak power) attenuation at the receive frequency (1090 MHz.)

2.3.3 Antenna Interface and Transmitter Monitor

The antenna interface and transmitter monitor consists of the following components:

a) A 4-port circulator. This device acts as diplexer, connecting transmitter, receiver and antennas. It provides 40-dB isolation between transmitter and receiver.

b) A transmitter monitor coupler and detector. A 30-dB RF coupler terminated with a diode detector permits monitoring the transmitter on an oscilloscope.

c) A diversity switch. This solid-state SPDT switch connects either the top-or bottom-mounted antenna, as commanded by the computer software. It will handle 1-Kw peak power, 2-watts average, and provides 20-dB inter-channel isolation. Switching time is 10 microseconds.

d) Two antenna low pass filters. These low pass filters prevent the radiation of harmonics generated by the diversity switch.

The TEU antennas are standard quarter-wave monopole stubs, with an approximately omni-directional radiation pattern in the horizontal plane.

2.3.4 Receiver Chain

The receiver chain (Fig. 2.3-5) consists of:

a) A receiver input limiter. This is a reflective-type device used for receiver input protection. It can withstand 1-kW peak pulses

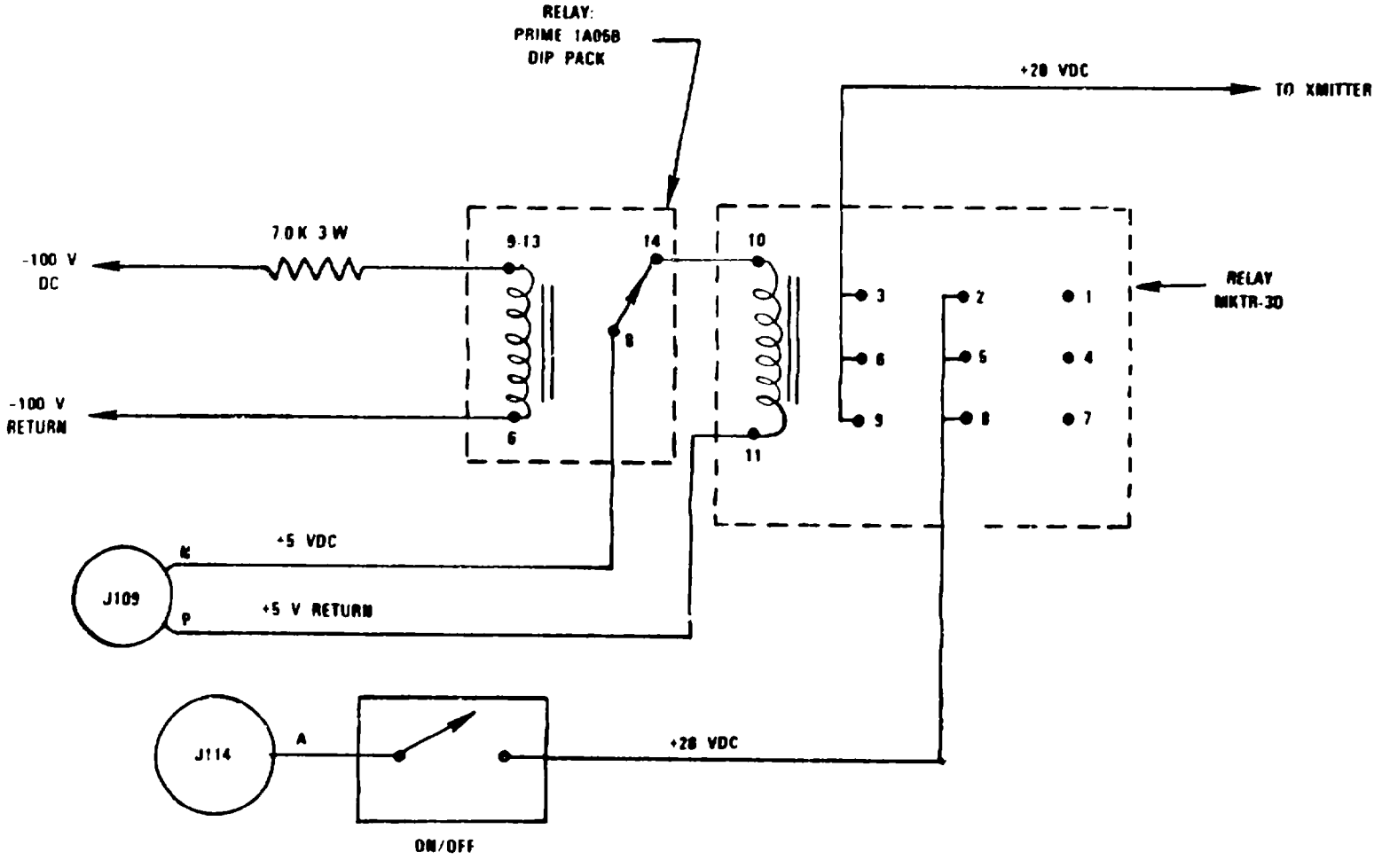


Fig. 2.3-4 Transmitter power supply interlock circuit.

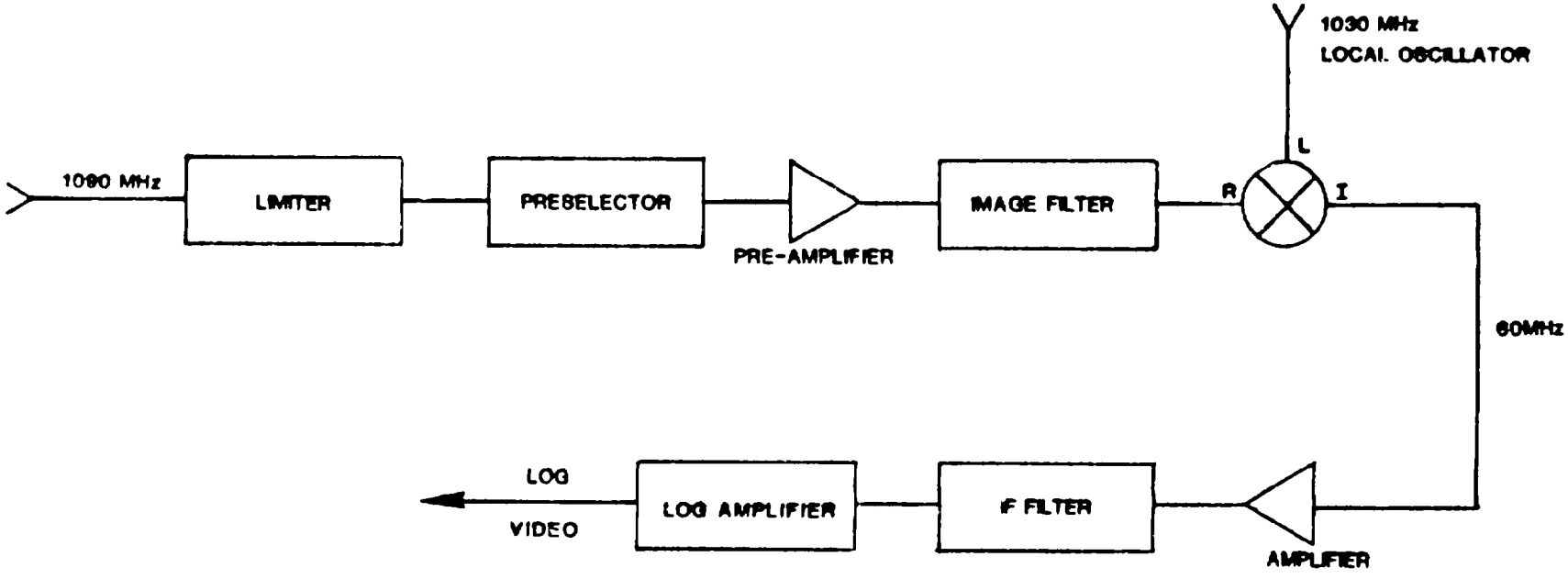


Fig. 2.3-5 Receiver chain block diagram

at a duty cycle of 0.2%, and will limit with a maximum cw leakage of 30 mw and maximum spike leakage of 100 mw.

b) A preselector filter. This filter is centered at 1090 MHz, has a bandwidth of 20 MHz, and provides 40-dB rejection to unwanted signals at the transmitter frequency (1030 MHz.)

c) A preamplifier. This 1090-MHz solid state amplifier has a noise figure of 2.5 dB, a power gain of 27 dB, and dynamic range of approximately 86 dB. Its 1-dB compression point occurs at an input level of + 12 dBm.

d) An image filter. This filter has a bandwidth of 120 MHz and an insertion loss of 0.6 dB. Its primary purpose is to reject image frequencies.

e) A down-converter. The insertion loss of the down-converter is 6 dB when + 7-dBm local oscillator power is injected to the L-port. The unit can withstand up to 400 mw at room temperature (50 mw at 100°C). The 1-dB compression point occurs when the signal power level at the L-port is 1 mw.

f) An IF amplifier. This amplifier provides 15-dB of gain. Its noise figure is 5 dB, which contributes an insignificant amount of noise to the system's front-end noise power level.

g) A 60-MHz IF filter. This filter has a signal bandwidth of 10 MHz and establishes the receiver bandwidth. It is a 5-pole Bessel filter adjusted to maintain phase for all frequencies within its 3-dB bandwidth. Its insertion loss is 2 dB. A 6-pole filter was used prior to the interference tests [Ref. 9] and may still be in some units.

h) A log amplifier. The 60-MHz signal is further amplified and video-detected using a miniature log amplifier. Its center frequency is 60 MHz and bandwidth is 20 MHz; rise-time is 50 nsec. Over an input range of -70 to 0 dBm its output rises from 0.2 to 2 volts into 93 ohms. This represents a transfer characteristic of 25.7 mv/dB. The detected output video is fed to the receiver video monitor and the video pulse quantizer (VPQ).

Design of the receiver is based on the power budget and assumptions given in Table 2.3-1. (For more information on TCAS air-to-air power budgets see Ref. 11). Receiver video performance is demonstrated in Fig. 2.3-6a,b,c. This figure indicates a typical receiver log video response for the first two pulses of a Mode S reply preamble at -46 dBm.

2.4 Video Pulse Quantizer

The Video Pulse Quantizer (VPQ) processes receiver log video pulses to produce quantized slope and signal strength, and a Mode S chip amplitude comparison signal. The VPQ is designed to produce, in conjunction with the pulse digitizer described in Section 2.5.2, accurate time-of-arrival and pulse width estimates of all received pulses over the full dynamic range of the receiver, where pulse width and time-of-arrival are defined at the -6 dB points.

TABLE 2.3-1

AIR-TO-AIR LINK POWER CALCULATION UNDER NOMINAL CONDITIONS AND AT A RANGE OF 10 NMI.

ITEM	UNITS	INTERROGATION LINK (1030 MHZ)	REPLY LINK (1090 MHZ)
1. Transmitter Power	dBm	57	57
2. Transmitter Cabling Loss	dB	3	3
3. Transmitter Mismatch Loss	dB	0	0
4. Transmitter Antenna Gain	dB	0	0
5. Free Space Path Loss	dB	118	118.5
6. Receiving Antenna Gain	dB	0	0
7. Receiving Mismatch Loss	dB	0	0
8. Receiving Cabling Loss	dB	3	3
9. Received Power	dBm	-67	-67.5
10. MTL	dBm	-77	-77
11. Nominal Margin (One Way)	dB	10	9.5
12. Nominal Margin (Two Way)	dB	9.5	

Notes:

Items 3 and 7, mismatch losses, refer to the differences, if any, that result when cables are attached to the antennas as compared with connections to perfectly matched loads. The nominal value is arbitrarily taken to be 0 dB.

Items 4 and 6 - the nominal value of aircraft antenna gain is arbitrarily taken to be 0 dB.

Item 5, free space path loss = $20 \log(4\pi R/\lambda)$ where R = range and λ = wavelength.

Item 9, received power equals the sum of items 1, 4 and 6 minus the sum of items 2, 3, 5, 7, and 8.

Item 10, MTL, denotes Minimum Triggering Level.

Item 11, nominal margin (one way) equals item 9 minus item 10. The small difference originates in free space path loss which is slightly different at the two frequencies.

Item 12, nominal margin (two way) is the lesser of the two values in item 11.

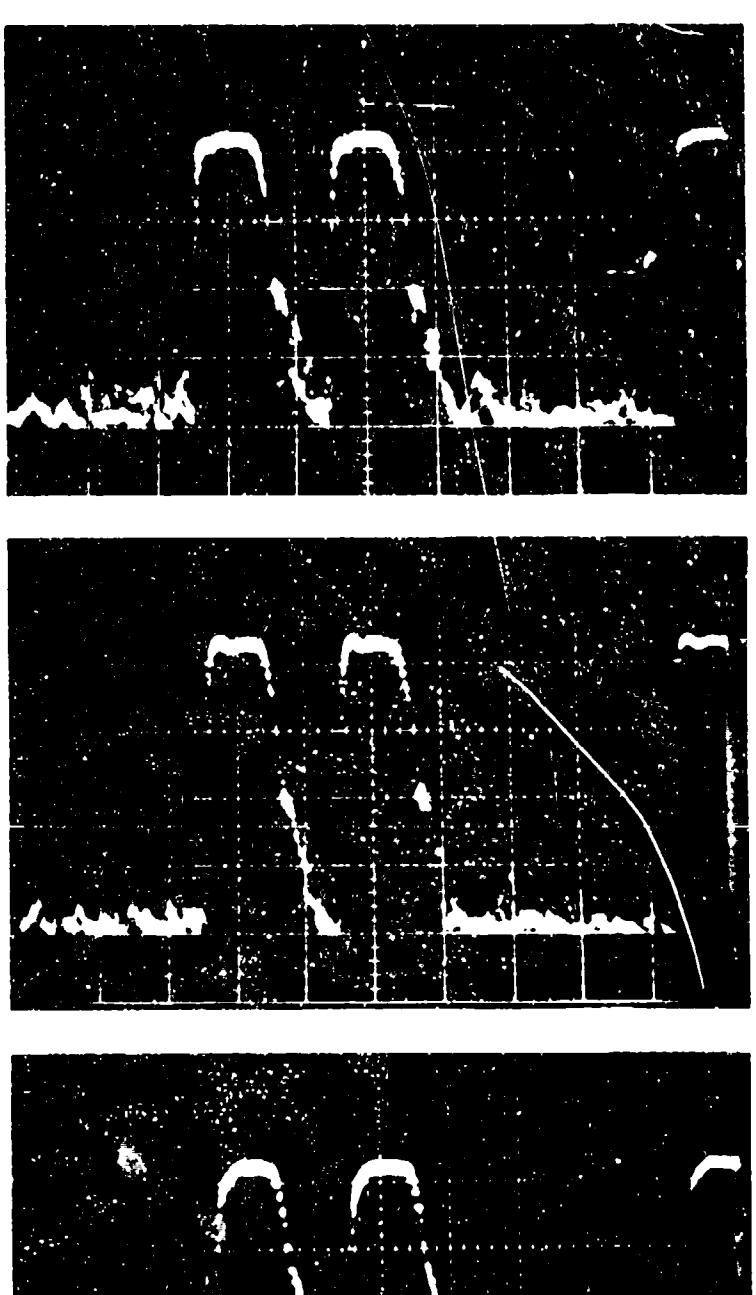


Fig. 2.3-6 Receiver log video response.

A simplified block diagram of the VPQ is shown in Fig. 2.4-1. Received log video pulses are amplified by A1 to produce a 40 mv/dB sensitivity at the A1 output. Slope detection is realized by subtracting a delayed version of the reply pulse (125-ns delay) from an undelayed version. As shown in Fig. 2.4-2, slope thresholds may be defined (48 dB per μsec) such that the outputs from comparators A6 and A7 contain edges which have fixed time relationships with the leading and trailing log video 6-dB points, independent of signal amplitude.

The log video signal strength is quantized by A8 by comparing the output of A1 against a threshold level. The threshold is determined by either the fixed minimum threshold level (MTL) or the dynamic minimum threshold level (DMTL) circuitry. These are both negative voltages, and the most negative is selected by the analog OR (implemented with diodes) as the threshold.

The MTL level is determined by a minimum MTL adjustment and a software controlled MTL setting. The minimum MTL is established to give 90% probability of successfully detecting ATCRBS bracket pairs at an input level of -77 dBm at the TEU receiver input port. It is adjusted by the potentiometer at the input to A4. The software may increase the MTL above this level (reduce the sensitivity) in 2-dB (80-mv) steps via the MTL control bits in the MCU control block. These control a D-to-A converter to bias the MTL voltage.

Dynamic MTL is provided to reduce the effect of multipath and low level fruit replies. It does this by raising the threshold level during a reply so that reply pulses will still be above the threshold but multipath due to that reply will normally be below the threshold. Of course, it is possible for multipath to be nearly as strong as the direct signal under some circumstances. DMTL cannot protect against such strong reflections.

DMTL is activated when the output of A1 exceeds the current quantized video threshold by 12 dB (480 mv). This is adjusted by means of the potentiometer at the input to A9. This triggers a timing network, Fig. 2.4-3. The track-and-hold module is forced into track mode (it may or may not have been holding a previous sample). 500 nsec afterward, it is set to hold mode. Since the input to the track-and-hold is delayed by 300 nsec, the signal level held is that 200 nsec following the point that triggered the DMTL.

Simultaneous with the signal level being held, a retriggerable one-shot is triggered. When processing ATCRBS replies, a 24-μsec interval is timed, corresponding to a reply length. At the end of this interval the track-and-hold is restored to track mode. When processing Mode S replies, a 7-μsec interval is timed, corresponding to the Mode S preamble length. At the end of this interval the track-and-hold is restored to track mode unless in the mean time the Mode S reply processor has detected a Mode S preamble. If so, it triggers another timing one-shot via the MSRP DMTL CONTROL line, and this keeps the track-and-hold in hold mode for an additional 120 μsec, corresponding to the length of a long Mode S reply plus an allowance for the multipath to die out. This two-step DMTL process for Mode S prevents ATCRBS fruit from triggering the DMTL for long intervals, which would desensitize the reply processing for Mode S.

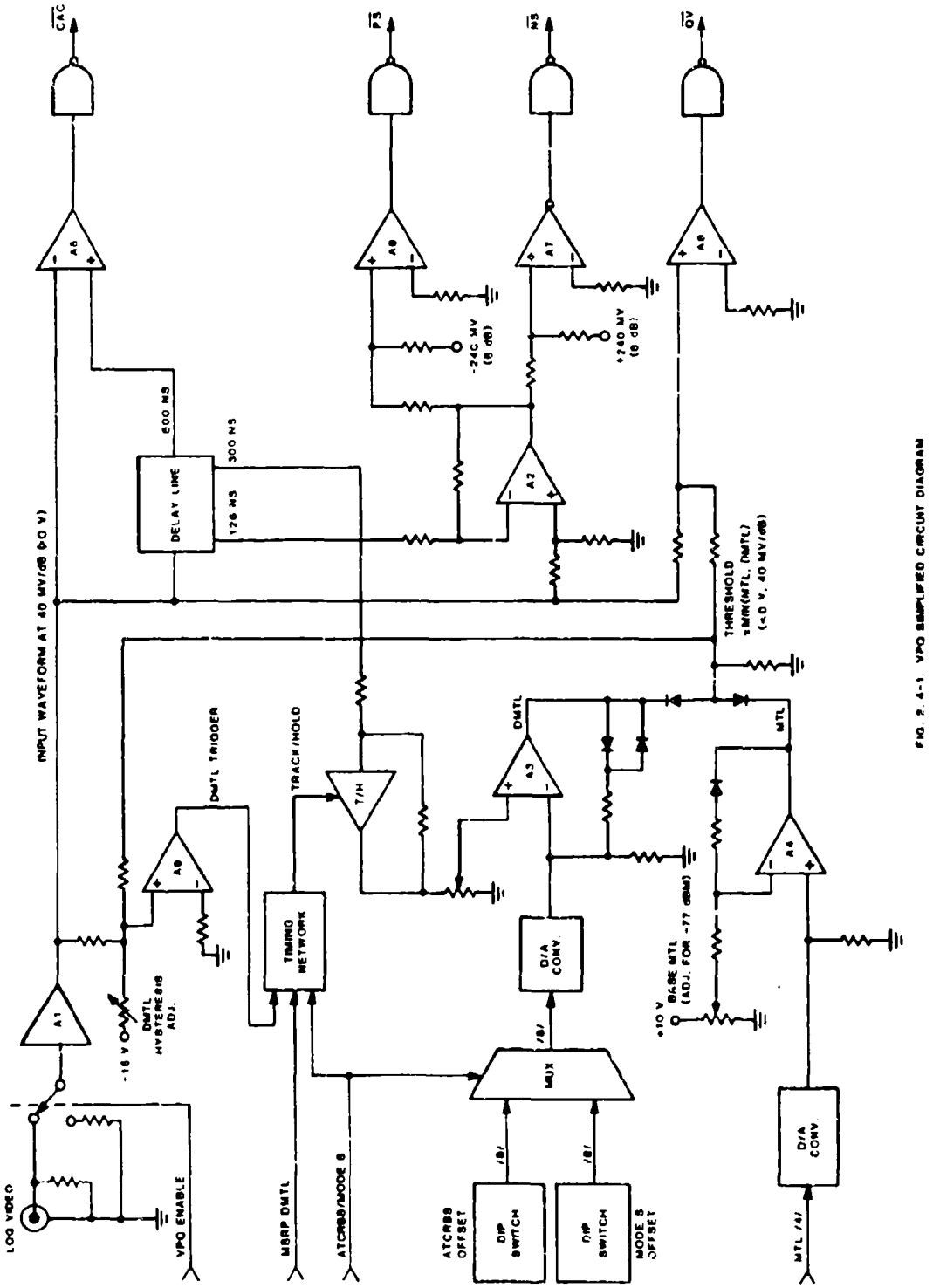


FIG. 2. 4-1. VPO SIMPLIFIED CIRCUIT DIAGRAM

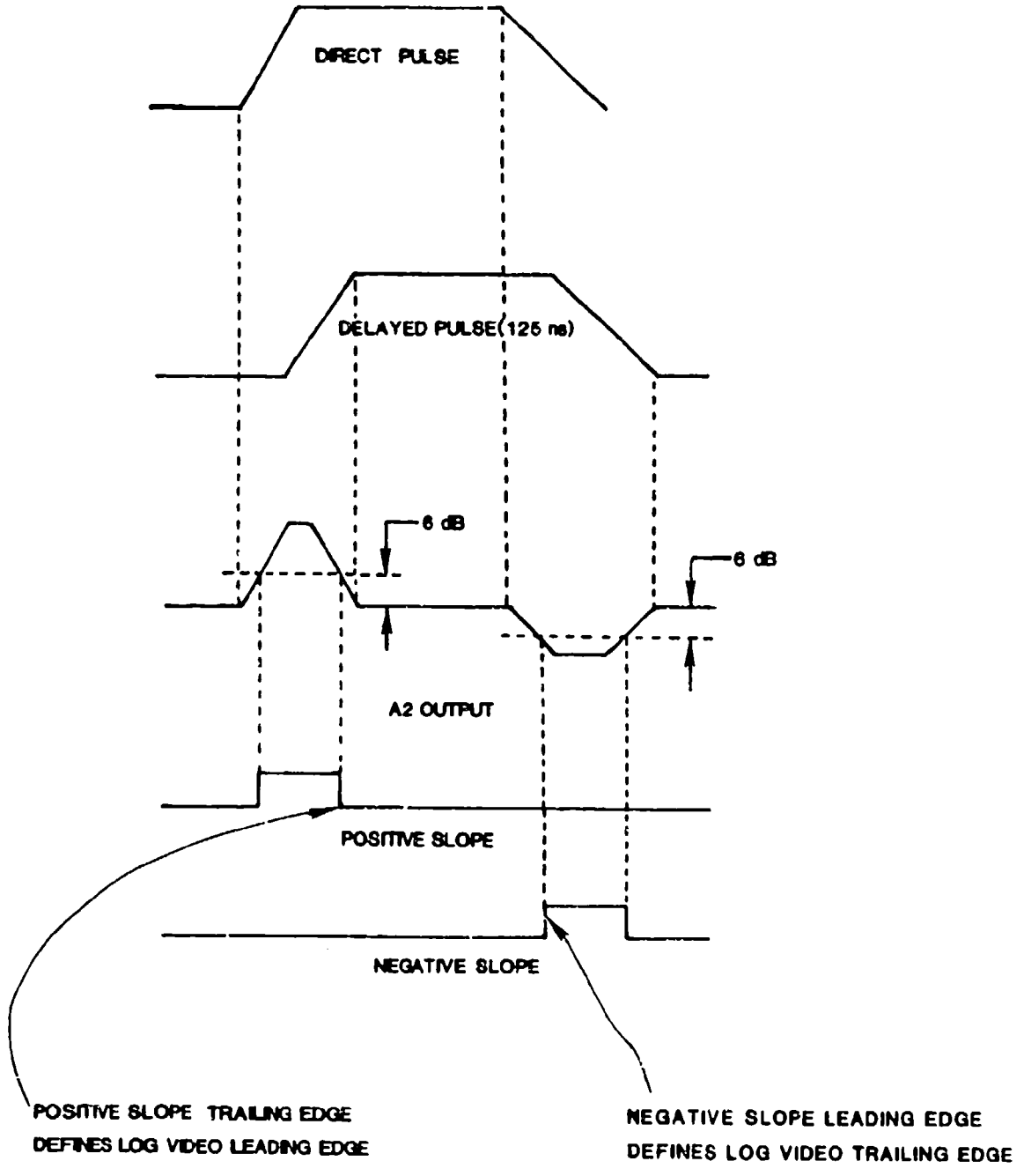


Fig. 2.4-2. VPQ Waveforms.

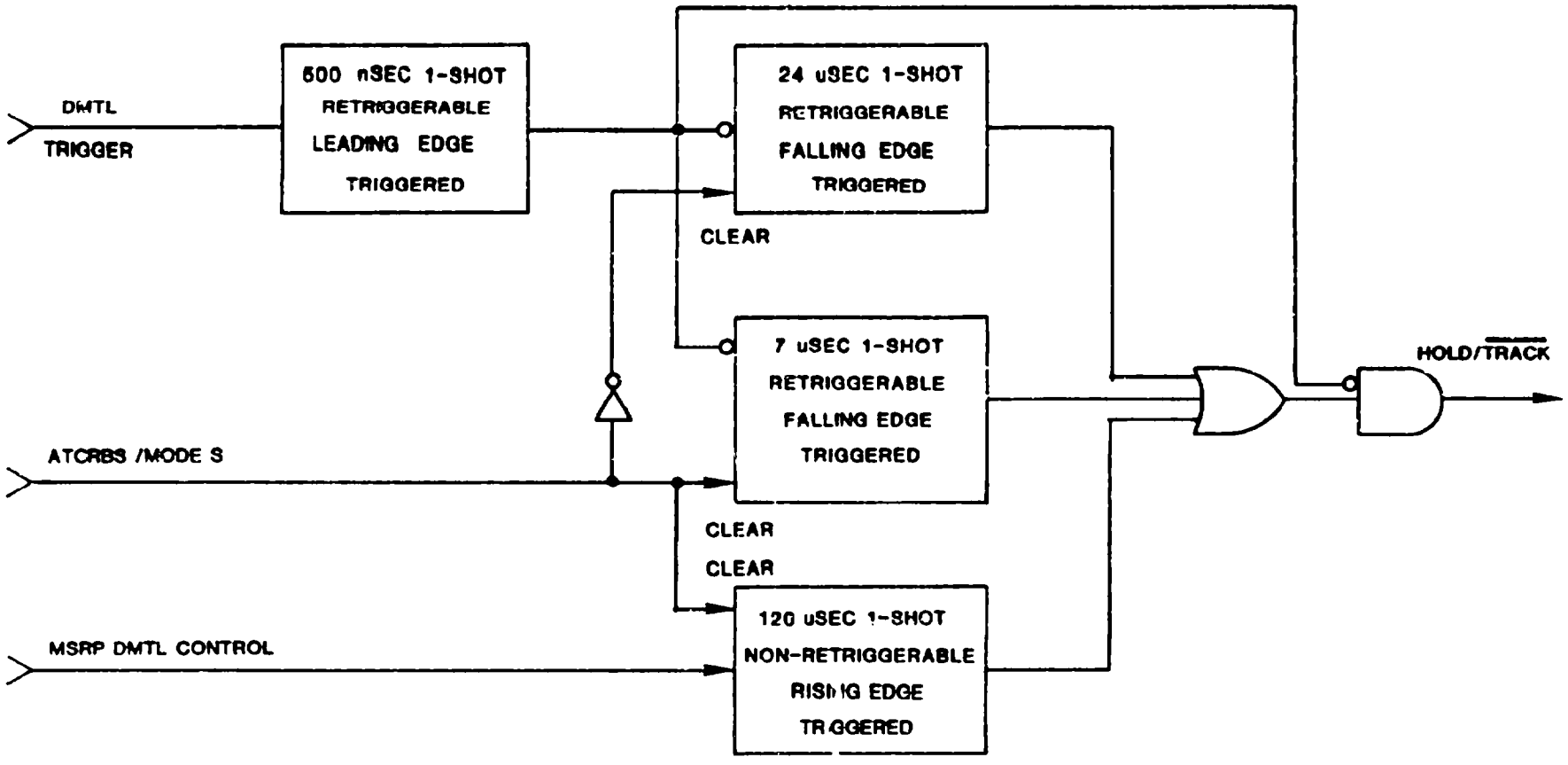


Fig. 2.4-3 VPQ DMTL timing network

The held signal level is then reduced in level by 9 dB (360 mv) for ATCRBS and 6 dB (240 mv) for Mode S. This is done by summing it with the output of a D-to-A converter. The inputs to the D-to-A converter come from one of two DIP switches selected by the multiplexor. The resulting threshold behavior is shown in Fig. 2.4-4 for ATCRBS and Fig. 2.4-5 for Mode S. In these figures the threshold is shown as a positive going signal to allow direct comparison with the input level.

Since the 7- sec and 24- sec one-shots are retriggerable, it is possible for the DMTL to be retriggered during these intervals if another pulse comes along which is at least 12 dB above the new threshold (in other words, at least 3 dB above a pulse which triggered ATCRBS mode DMTL, or 6 dB above a pulse which triggered Mode S DMTL). The advantage of this is that low level fruit replies or other interference sources cannot capture the DMTL and prevent it from functioning properly when a stronger desired reply comes along.

A disadvantage of DMTL is that strong fruit replies or other interference sources can prevent detection of weaker desired replies. It would normally also be true that strong desired replies (replies to TEU interrogations) would prevent detection of weaker overlapping ones. The use of the whisper-shout interrogation technique, however, tends to assure that all desired replies have about the same signal strength.

Decoding of the Mode S pulse position modulation (PPM) is provided for by the chip amplitude comparison (CAC) circuitry. Each 1- sec Mode S bit position is divided into two 500 nsec "chips". If the signal level in the first of these is greater than the level in the second a 1 is declared, and otherwise a 0 is declared.

2.5 Video Digitizer and ATCRBS Reply Processor

2.5.1 General Description

The ATCRBS reply processor detects ATCRBS replies, decodes their associated code pulses, and determines reply range. The processor also detects and eliminates phantom replies, and checks for the presence of overlapping code pulses belonging to other replies.

A block diagram of the ATCRBS reply processor is shown in Fig. 2.5-1. First, video data is digitized and used to find the leading edges of reply pulses. The Positive Slope (PS), Negative Slope (NS), and Quantized Video (QV) signals from the Video Pulse Quantizer (VPQ) are digitized to produce Actual Leading Edge (ALE), Actual Trailing Edge (ATE), and Sampled Quantized Video (SQV). If reply pulses overlap, they may be merged into a single wide pulse and the embedded leading edges will be missed. The leading edge generator tests the measured width of the SQV to determine if it exceeds the width of a standard ATCRBS pulse* and indicates if one or more leading edges have not been

*A standard ATCRBS pulse width is 450 100 ns. It therefore is sampled by between 2 and 5 clock pulse intervals, i.e., two to five 121-ns periods (12 samples per each 1.45-microsecond interpulse period).

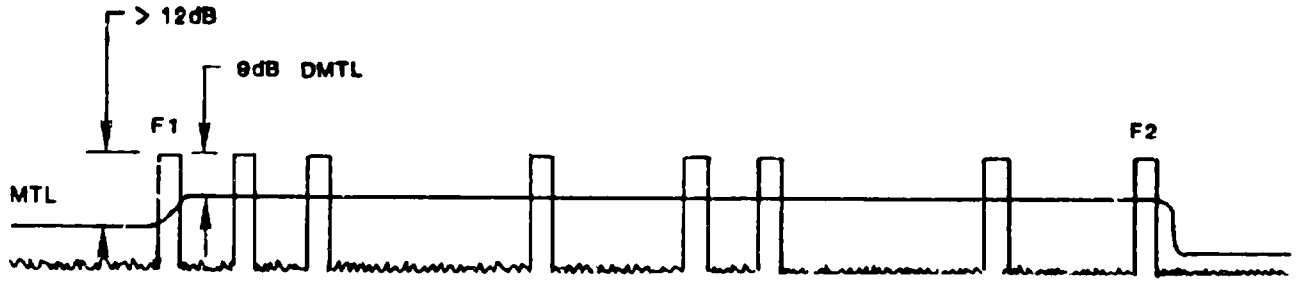
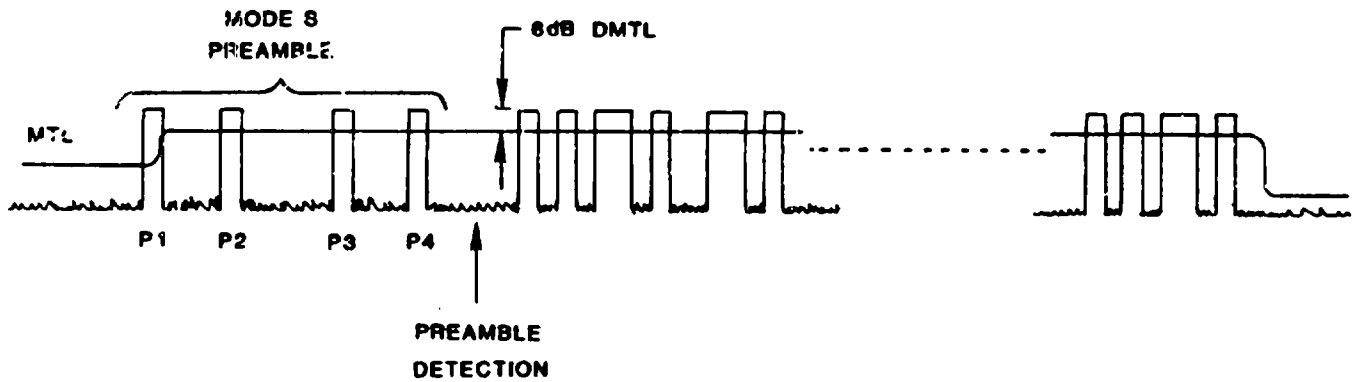


Fig. 2.4-4 ATCRBS DMTL waveforms.



DMTL= DYNAMIC MINIMUM THRESHOLD LEVEL

Fig. 2.4-5. Mode S DMTL Waveforms.

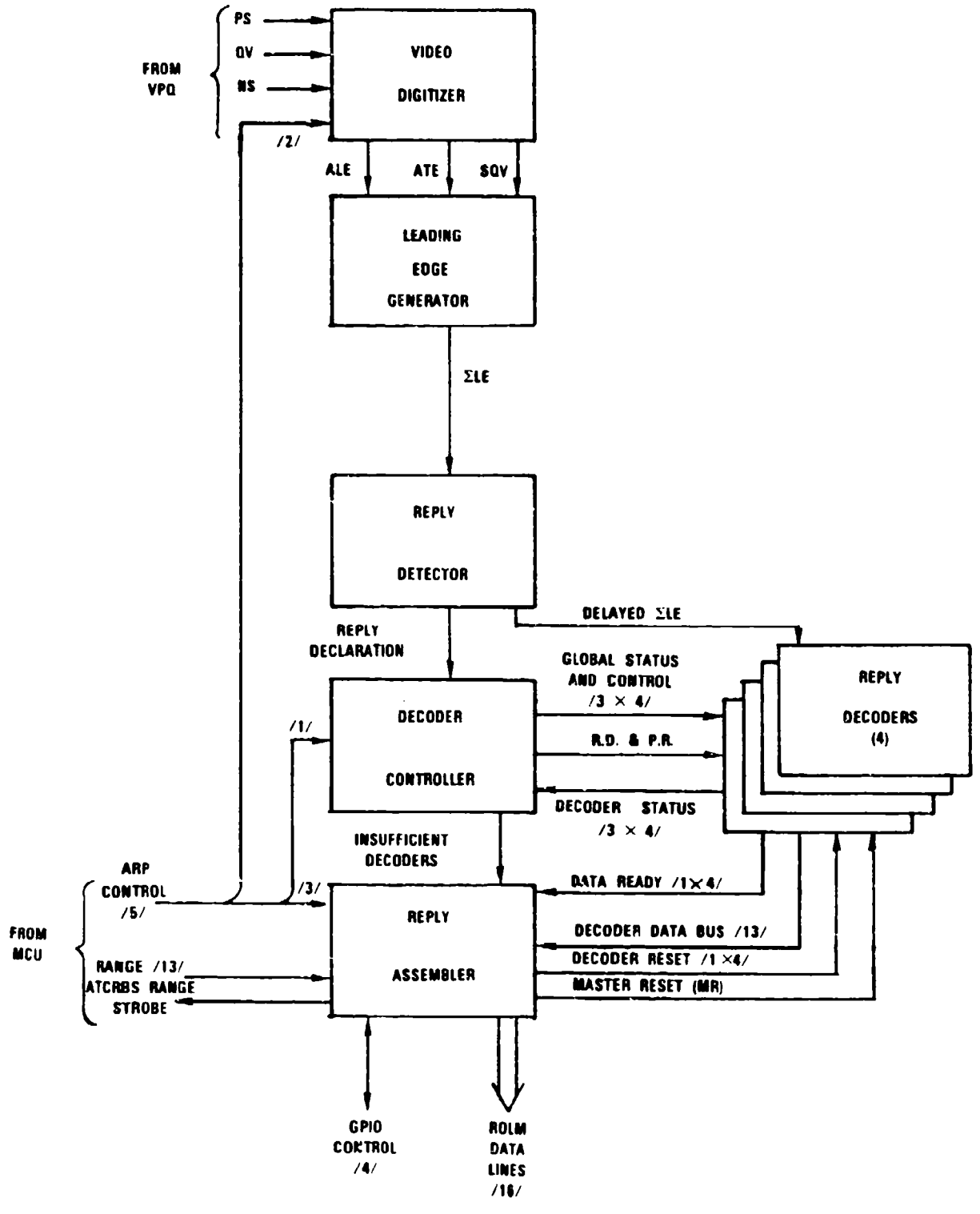


Fig. 2.5-1 ATCRBS reply processor block diagram.

detected. The leading edge generator then inserts additional leading edges at approximately the correct positions so as to form a sum leading edge (Σ LE) pulse stream. The remainder of the ATCRBS reply processing works solely from this leading edge pulse stream.

The ATCRBS framing pulses (F1-F2) are detected by delaying the Σ LE pulses by 20.3 μsec and comparing the delayed and undelayed Σ LE pulse streams for the simultaneous occurrence of a pulse on each line. When this condition occurs, the reply controller activates an idle reply decoder to process the corresponding reply pulses. Four independent decoders are employed so that up to four overlapping replies may be simultaneously decoded.

The reply processor contains logic to detect and eliminate phantom replies. A phantom reply is defined as a false reply whose F1 pulse is a valid code pulse or framing pulse of one reply and whose F2 pulse is a valid code or framing pulse of another reply. Figure 2.5-2 shows an example of two replies which are separated in time so that an A1 pulse of the earlier reply and a B4 pulse of the second reply are separated by the standard framing pulse pair spacing. The ATCRBS processor may be set to eliminate the phantom reply resulting from these two interfering replies. The reply processor does not eliminate C2-SPI phantoms (Fig. 2.5-2) whose framing pulses result from the spacing of the C2 and SPI pulses of a real reply. The elimination of phantoms is based on the assumption that no actual F1-F2 bracket pairs occur at the phantom pulse positions. The reply processor is also capable of flagging, but not eliminating, phantoms. The selection of phantom elimination or phantom flagging mode is made by the software and controlled via the PB bit of the MCU control block (Fig. 2.2-1).

As shown in Figure 2.5-2, overlapping replies may garble some of each other's code positions. The four reply decoders transmit global garble flags to each other indicating those time positions which belong to the reply they are processing. Note that the presence of a global garble flag only indicates that a reply decoder considers that time to be a valid code pulse position, not that a leading edge pulse was detected at that position. Thus the A2 bit of reply 1 will be flagged as being garbled due to the presence of the F1 of reply 2, but so will bits C4 through D4. Similarly, all bit positions of reply 2 through B2 will be flagged as garbled bits.

Phantom replies interact with the garble flagging logic to cause some bit positions to be mislabeled. This is due to the fact that phantoms are not eliminated until the second of the two real replies is detected. Until that time they are assigned to a decoder and can generate garble flags. These flags are not removed once the phantom is detected. Thus, in Fig. 2.5-2, bit positions A1 and C2 of reply 1 will be flagged as garbled even though no real reply is present to garble them.

When a reply is ready from the decoder, the reply assembler will forward 12 code bits, 12 garble bits, range, and status flags to the computer. The format is as shown in Fig. 2.5-3. The modified format used by TEUs with AOA is given in Section 2.7.

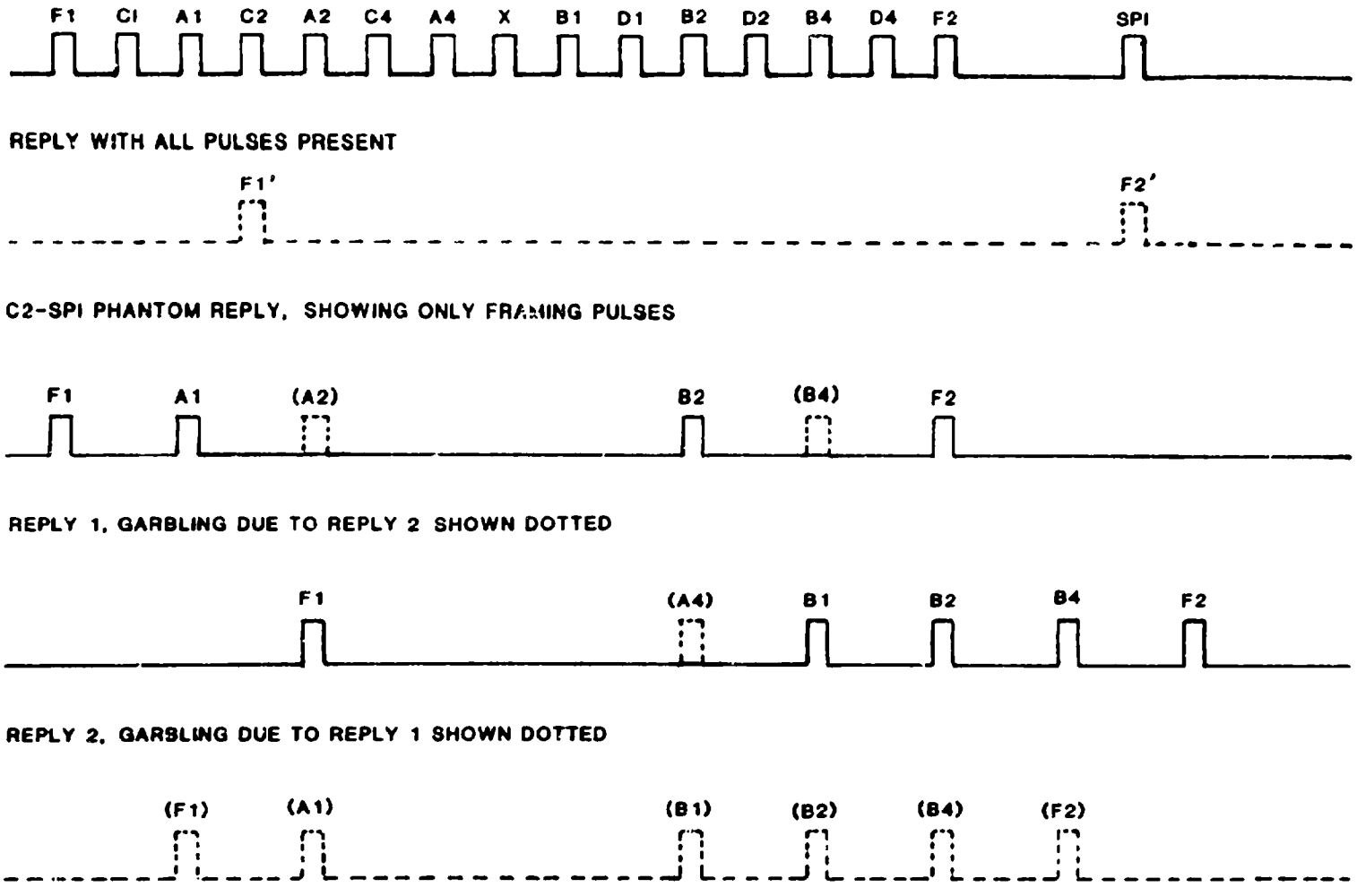
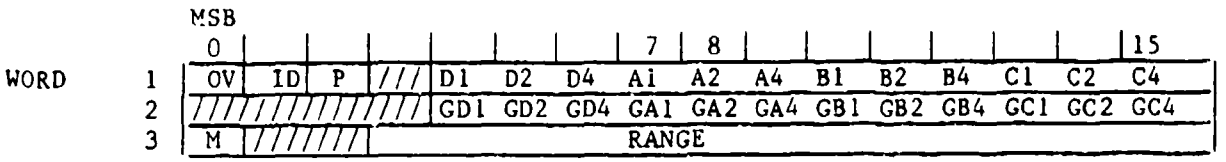


Fig. 2.5-2 Garbled reply and phantom reply examples.



OV — Buffer overflow bit. It is set ("1") when the buffer is full (contains four replies) and a decoder has a reply ready to be transferred to the buffer. This latter reply will be discarded. The OV bit is reset when the condition is reported to the CPU via the next reply placed into the buffer after the overflow condition clears.

ID — Insufficient decoders bit. This bit is set ("1") if a bracket is detected during a period when all four decoders are busy. It is reset when the condition is reported to the CPU via the first of the four replies-in-process that is transferred to the CPU.

P — Phantom bit. It is set ("1") when this reply is a potential phantom. If phantom elimination is enabled, a reply with the P bit set is not an actual phantom but did overlap with an earlier reply such that it was declared a potential phantom. If it had satisfied the overlap conditions with a later reply and been declared a phantom it would have been eliminated. If phantom elimination is turned off, a reply with the P bit set may be an actual phantom as well as a potential phantom.

D1 D2, ... C4 — ATCRBS data bits.

GD1, GD2, ... GC4 — Garble bit for each ATCRBS data bit.

M — Mode of interrogation. 1=A, 0=C.

Range — Round trip time plus transponder delay plus processor hardware delay. (LSB 125 ns). (Range count - 636.0) multiplied by 0.0101144 gives true range in nautical miles.

Fig. 2.5-3. ATCRBS reply processor data format.

2.5.2 Video Digitizer

The digitizer accepts slope and quantized video information from the VPQ and generates Actual Leading Edges, Actual Trailing Edges, and Sampled Quantized Video for the ATCRBS reply processor and Sampled Quantized Video for the Mode S reply processor. The digitizer clock, which determines the sampling rate, is run at 8.27 MHz in the ATCRBS mode and 8.00 MHz for Mode S. The mode is controlled by the CS-0 signal from the Mode S reply processor, as shown in Fig. 2.5-4.

The digitizer, shown in simplified form in Fig. 2.5-5, establishes actual leading edges whenever PS makes a 1→0 transition and QV=1. This defines the leading edge of the log video pulse.

Trailing edges are defined in several ways in order to account for garble situations. See Fig. 2.5-6. Note that Mode S replies evoke a sixth trailing edge case not used in ATCRBS processing.

The generation of SQV for a single ungarbled pulse is shown in Fig. 2.5-7. The width of SQV matches the width of the log video pulse at the 3- to 6-dB width. Variations will occur due to variations in the reply frequency which affect the pulse response of the receiver IF filter.

2.5.3 Pseudo Leading Edge Generation and Short Pulse Rejection

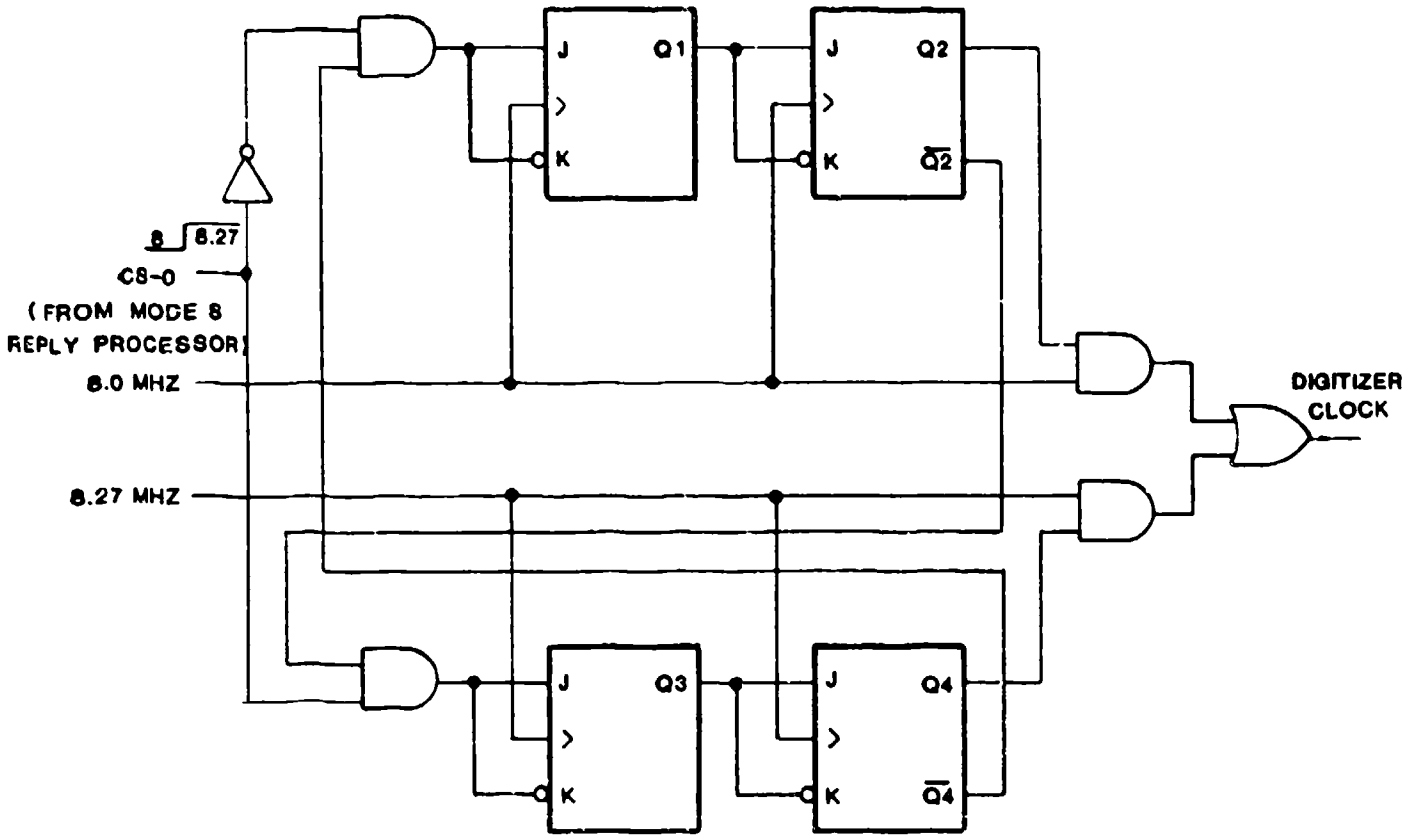
Overlapping replies will generate pulses having widths different than the standard code pulse width (2-5 samples at 121 nanoseconds per sample). To reject narrow pulses and to estimate the leading edge positions of pulses that have been combined in the TEU receiver, the logic shown in Fig. 2.5-8 is used.

The first pulse processing step eliminates narrow pulses having widths less than two samples. AOA units require the elimination of pulses narrower than three samples (see Section 2.7.4). Next, Counter-1 is used to artificially inject a pseudo leading edge four sample positions prior to the end of all pulses having widths greater than or equal to 6. See Fig. 2.5-9.

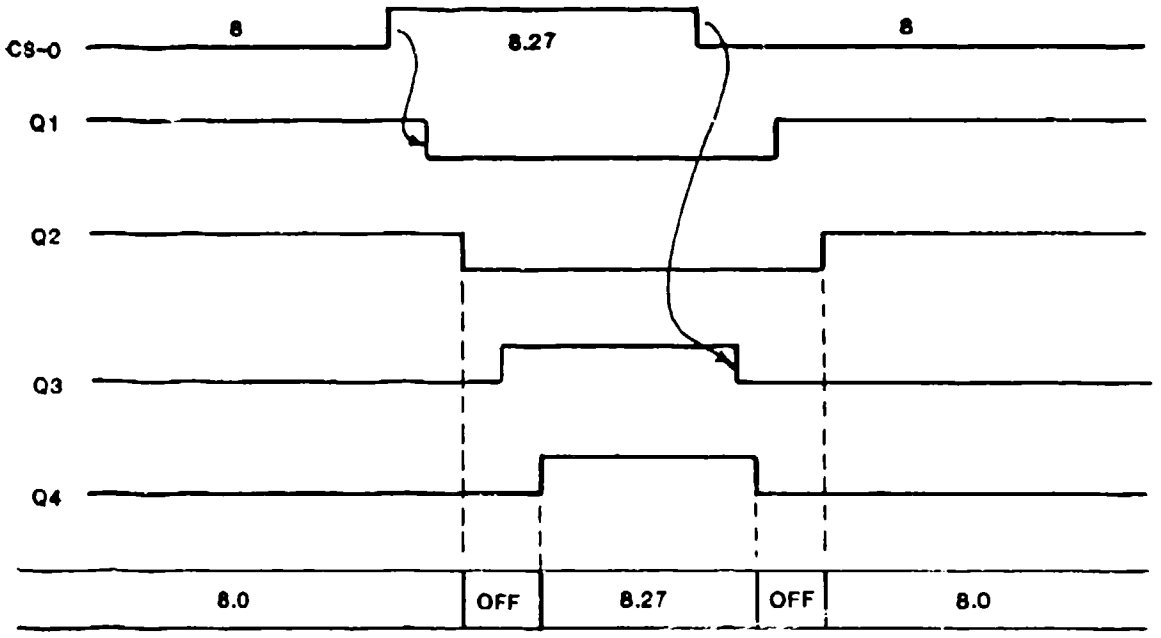
The rest of the logic in Fig. 2.5-8 analyzes pulses to determine whether extra leading edges need to be inserted (see Figs. 2.5-9 and 2.5-10).

A summary of the rules for pseudo and extra leading edge generation follows. When a leading edge declaration (ALE) is followed by a trailing edge declaration (ATE), PW is used to denote their separation in terms of sample intervals. LW is used to denote the spacing between two successively declared ALE's (no intervening ATE's) in terms of sample intervals.

(a) If PW = 1 then the associated ALE shall not be represented in the ΣLE data stream.

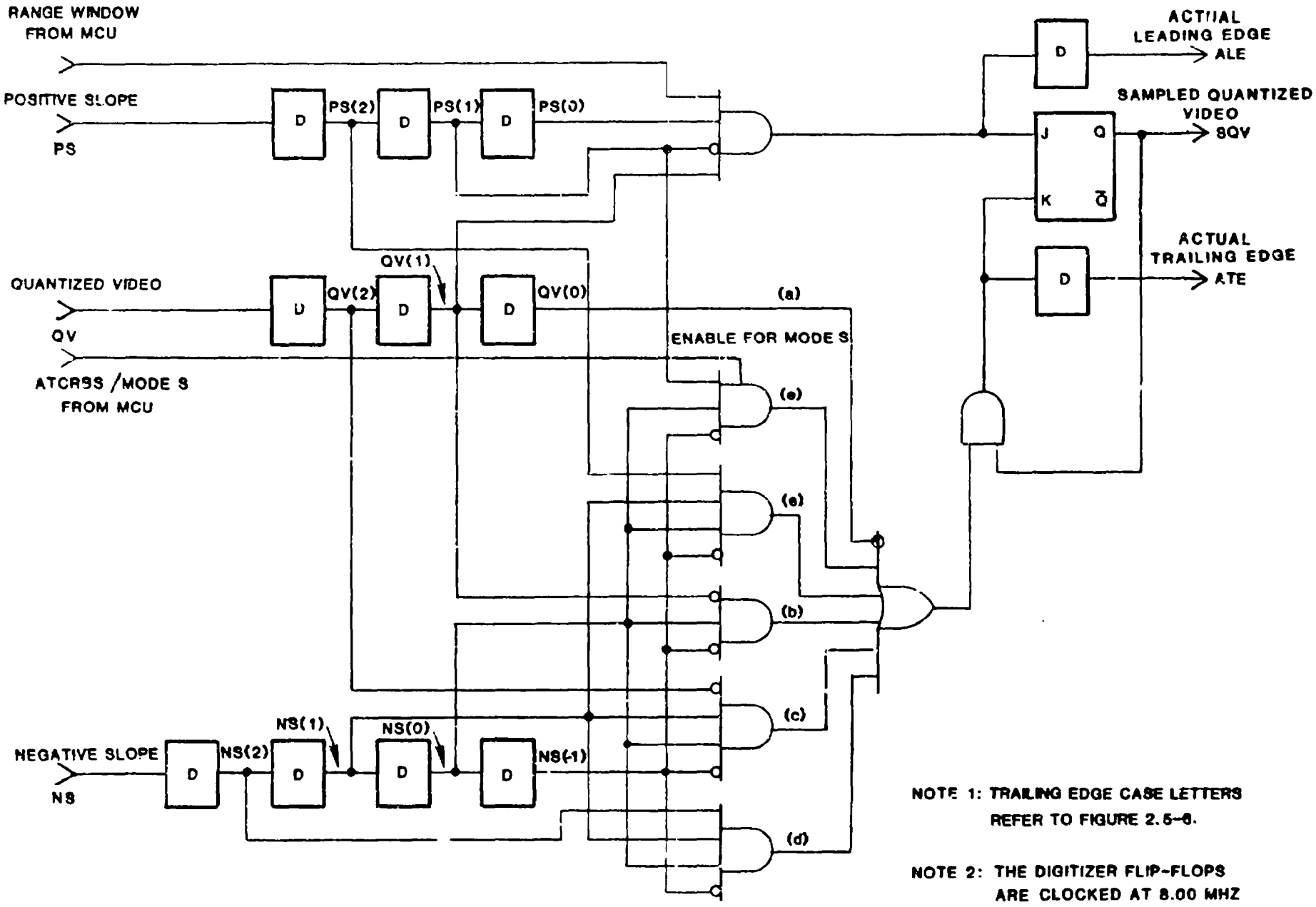


a) DIGITIZER CLOCK SWITCHING CONTROLLER



b) CLOCK SWITCHING TIMING

FIGURE 2.5-4 DIGITIZER CLOCK CONTROL



NOTE 1: TRAILING EDGE CASE LETTERS REFER TO FIGURE 2.5-6.

NOTE 2: THE DIGITIZER FLIP-FLOPS ARE CLOCKED AT 8.00 MHZ FOR MODE S AND AT 8.27 MHZ FOR ATCRBS.

Fig. 2.5-5 Digitizer logic.

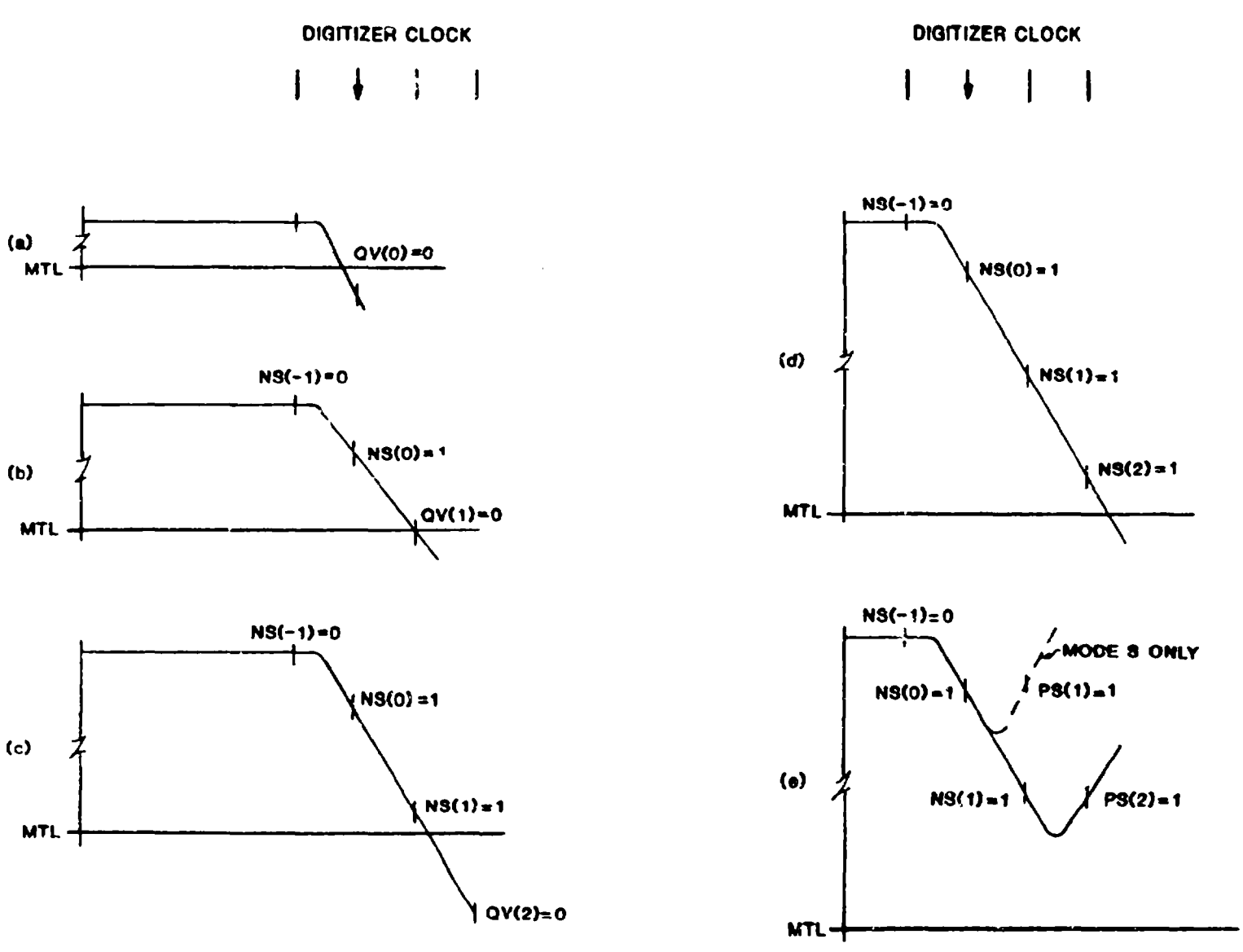


FIGURE 2.5-6 TRAILING EDGE CONDITIONS

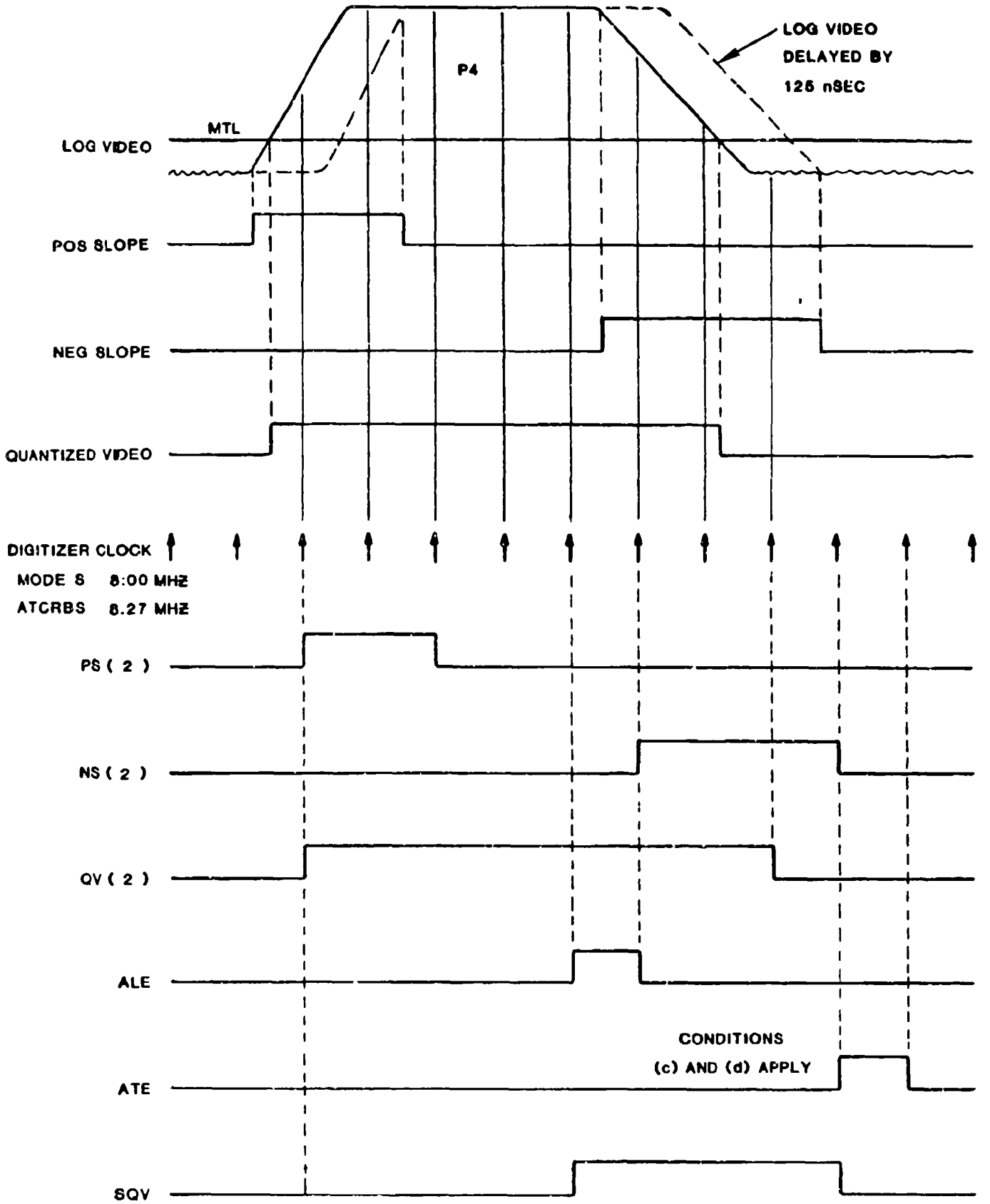


FIGURE 2.5-7 DIGITIZER TIMING

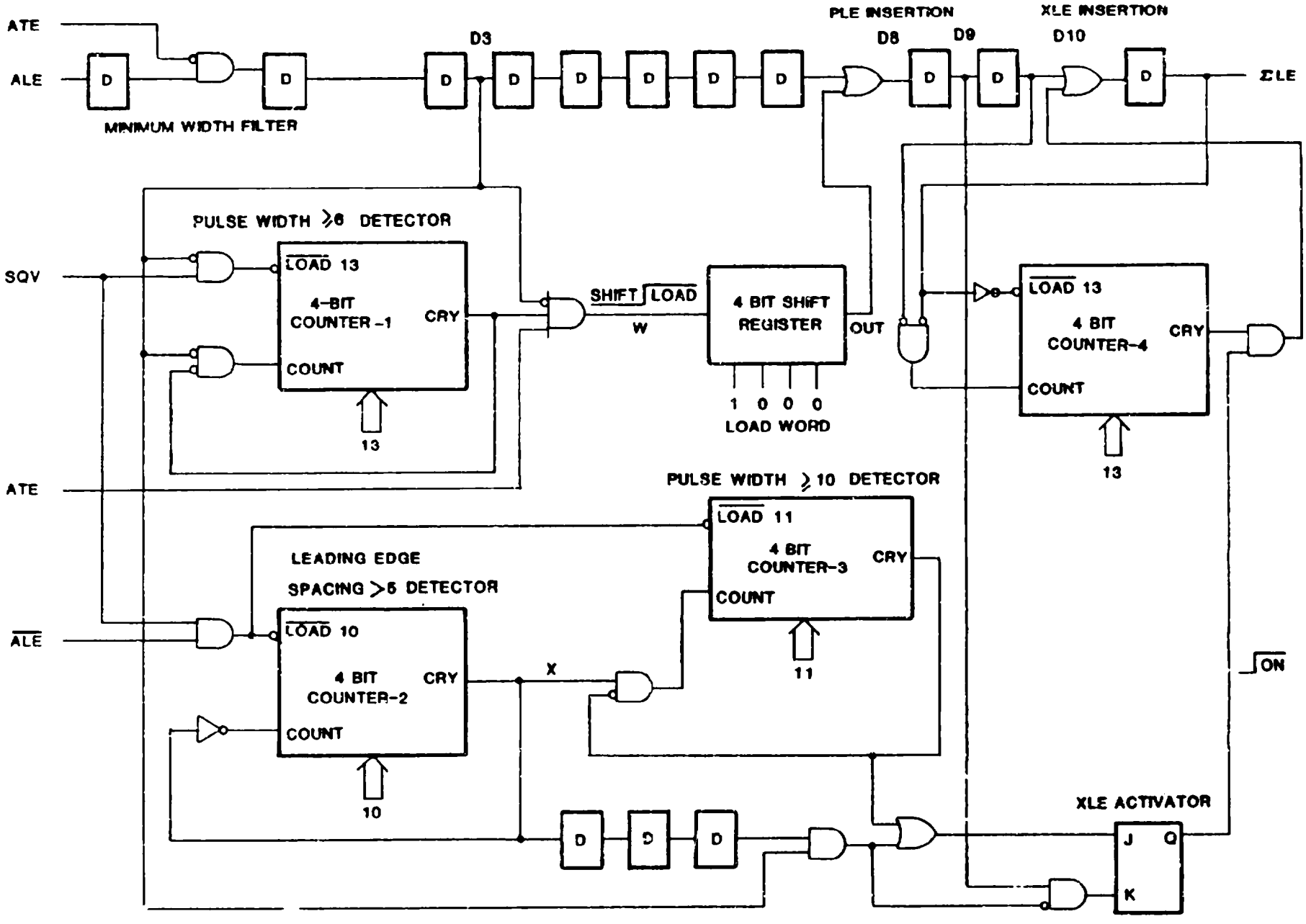


FIGURE 2.5-8 PSEUDO AND EXTRA LEADING EDGE GENERATOR

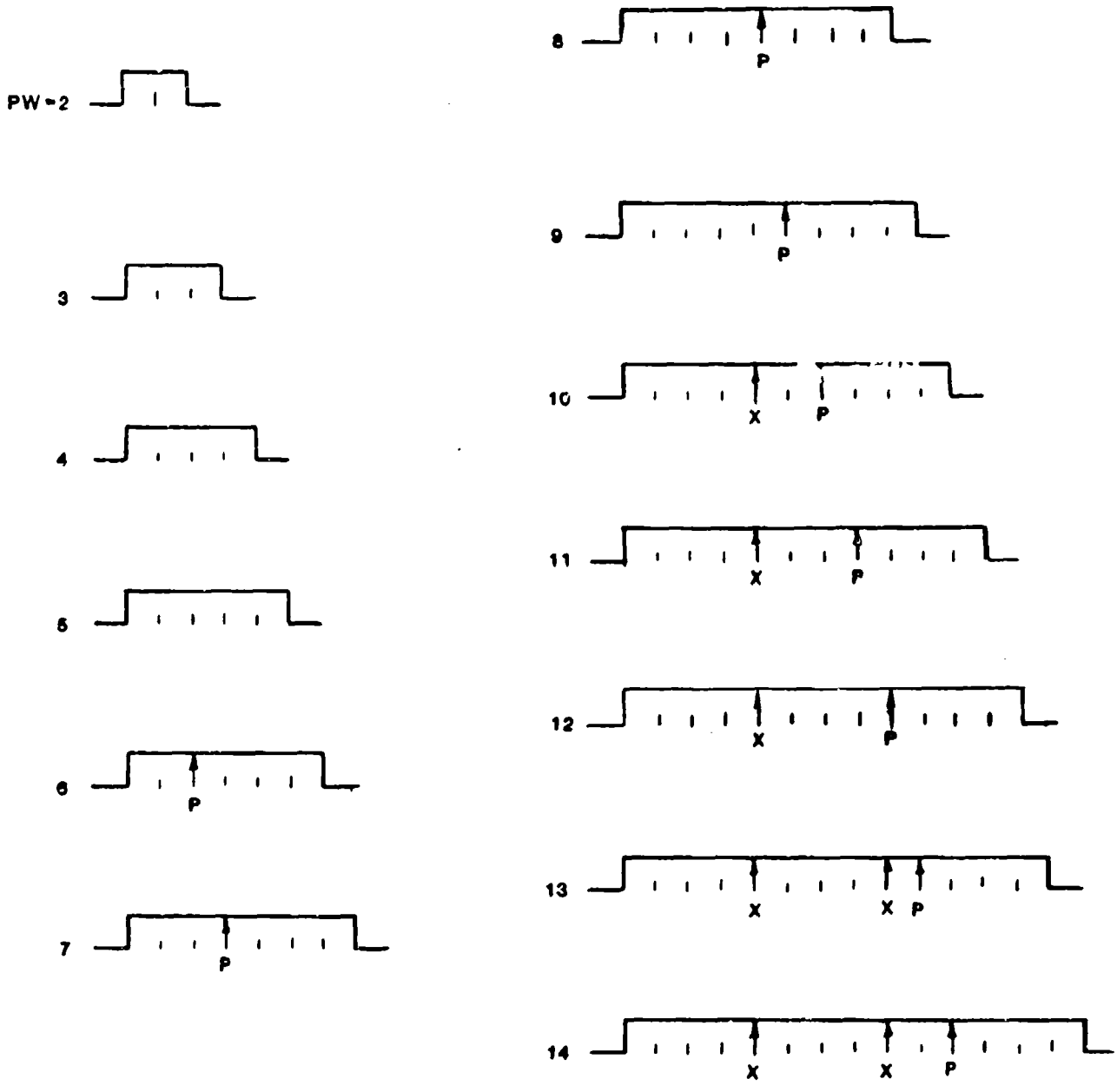
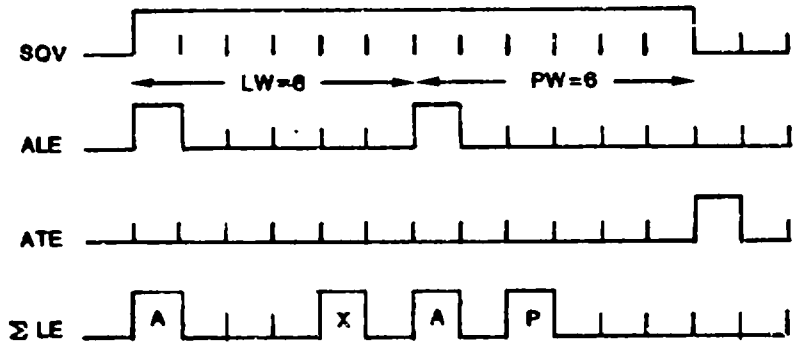


Fig. 2.5-9. Pseudo and extra leading edge pulse generation, no embedded ALE'S.



A = ACTUAL LEADING EDGE

X = EXTRA LEADING EDGE

P = PSEUDO LEADING EDGE

Fig. 2.5-10 Extra leading edge generation with embedded ALE's

(b) If PW > 6 then a pseudo-leading edge shall be declared at the sample time four sample intervals prior to the ATE. This is controlled by Counter-1.

(c) If PW > 10 then additional leading edges (called extra leading edges) shall be declared every fourth sample interval following the initial leading edge but prior to the pseudo-leading edge inserted in (b). This is controlled by Counter-3.

(d) If LW > 5 then extra leading edges shall be declared every fourth sample interval following the first leading edge but prior to the second leading edge. This is controlled by Counter-2 for LW=6-9, and by Counter-3 if LW > 10.

2.5.4 Reply Detection

When the Σ LE stream contains two leading edge declarations separated by 20.3 μ sec ± 1 sample, the logic shown in Fig. 2.5-11 will produce a reply detection pulse. The counter is preset to 88 and after 168 counts it returns to 88. The two RAM's act as 1-by-168 ping-pong buffers. The RAM write strobes (not shown in Fig. 2.5-11) are arranged so that when input Σ LE data is entered into RAM A, RAM B output is selected by the mux gate and vice versa. The design accounts for the ± 0.1-microsecond tolerance on F1-F2 spacing and prevents reply declarations from being spaced more closely than 3 clock intervals. This fact is used in the design of the reply assembler, which only has to read out one reply decoder at a time and has two clocks to do so before another reply decoder can be ready.

2.5.5 Reply Decoder Processing

When a reply detection occurs a free reply decoder is assigned using assignment logic contained in the Busy ROM in Fig. 2.5-12. If no decoder is free, the INSUFFICIENT DECODERS signal is sent to the reply assembler, where it is latched and transmitted to the CPU with the next reply transferred.

The reply decoder, shown in Fig. 2.5-13, is enabled by the $\overline{\text{SELECT}}$ pulse, which resets the counter to 51, and sets BUSY.

DELAYED Σ LE data is sampled using the Data flip-flop and the DATA FF CLEAR signal from the control ROM. As shown in Fig. 2.5-14, a leading edge occurring in any of three sample positions centered on the nominal leading edge position will cause a 1 to be declared.

In addition, a LOCAL GARBLE flag is generated and sent to all other reply decoders via the global garble logic shown in Fig. 2.5-12. The GLOBAL GARBLE flag, if true at the time it is sampled by the GARBLE STROBE, indicates that one or more of the other reply decoders is processing a reply which may have a pulse leading edge in any of the three positions currently being sampled. The nominal pulse leading edge positions of different replies must be separated by at least three clock intervals in order to not garble.

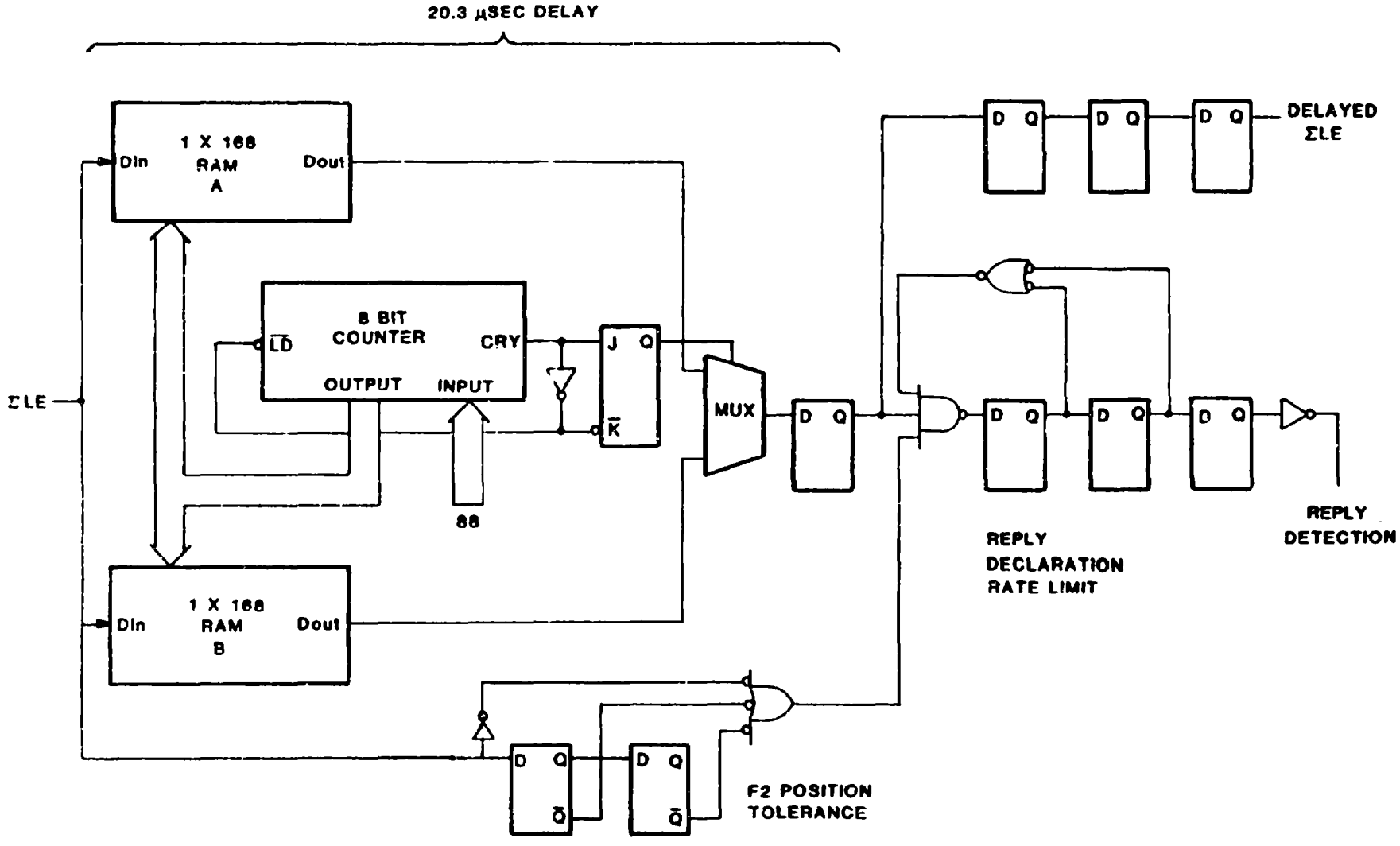


Fig. 2.5-11 Reply detector.

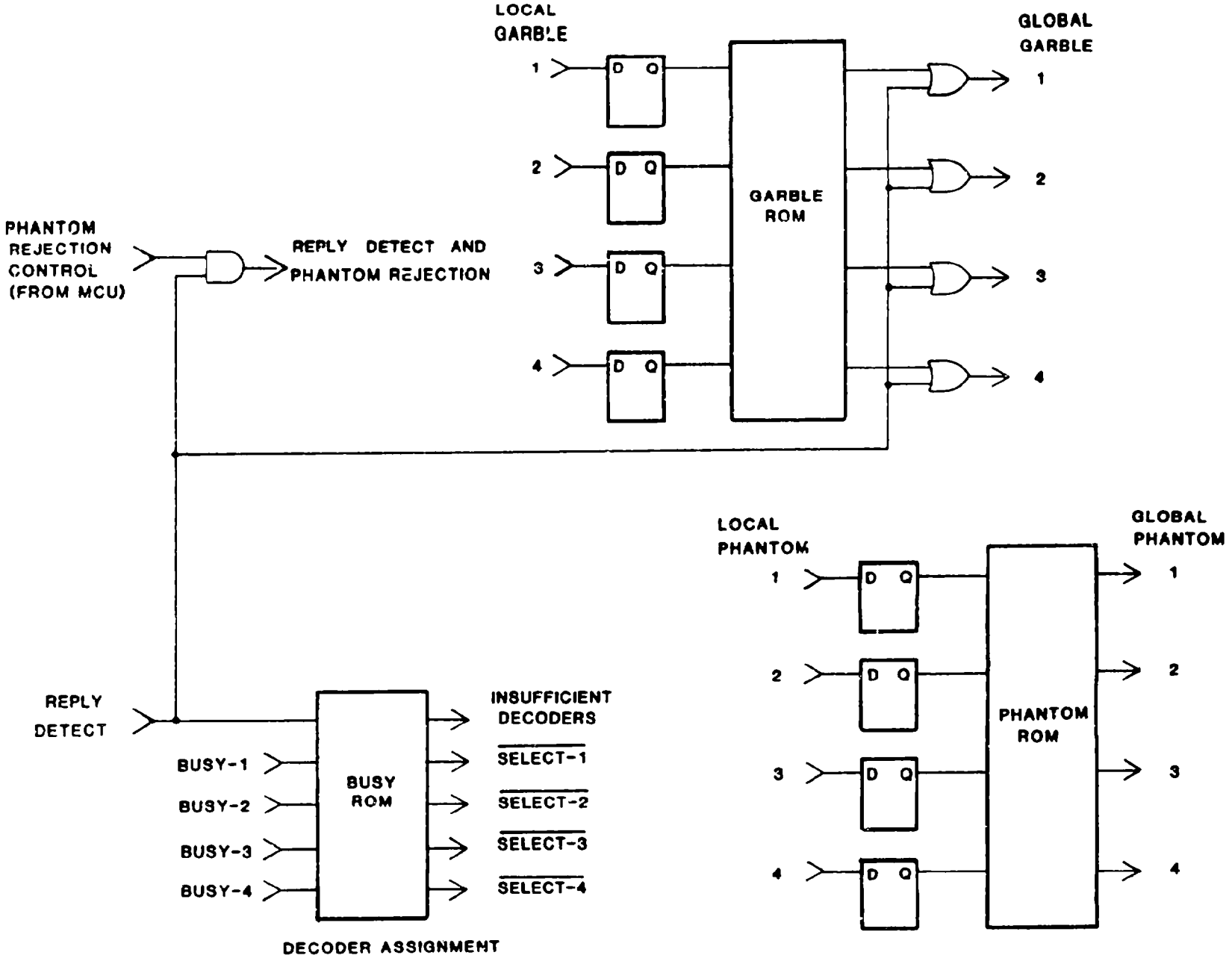


Fig. 2.5-12. Reply decoder control.

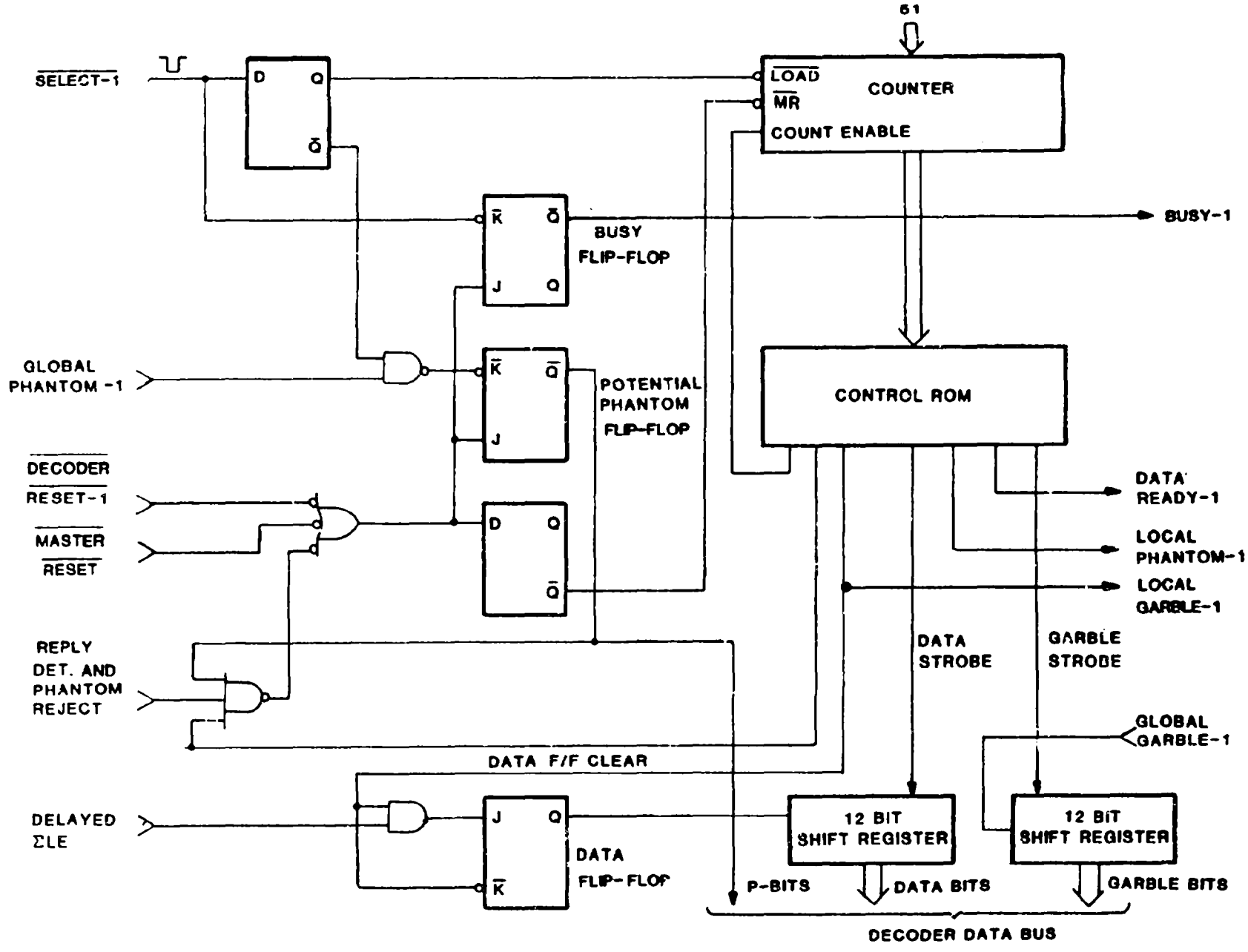


Fig. 2.5-13 Reply decoder number 1

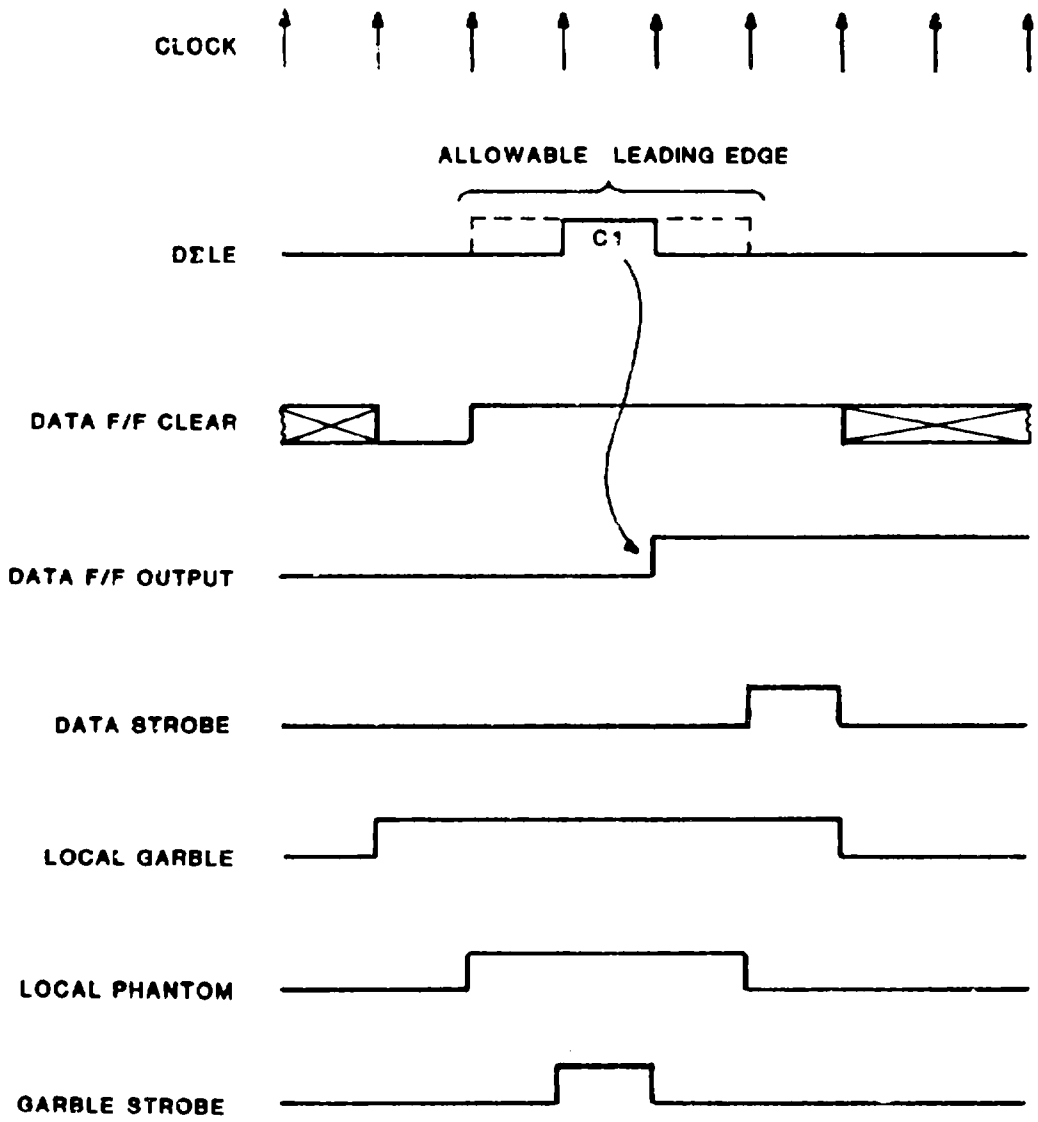


FIGURE 2.5-14 REPLY DECODER TIMING

Similarly, a local phantom signal is sent via the global phantom logic to all other decoders. See Fig. 2.5-12. This causes the P-bit (potential phantom) to be set if F1 (reply detection) has occurred in a time position which aligns with any of the data sample positions of another reply. A potential phantom becomes an actual phantom if the F1 leading edge of a subsequent reply aligns with one of the nominal leading edge positions of the potential phantom within ± 2 clock intervals. If this occurs and phantom elimination was enabled by the software, the decoder is reset. Otherwise the phantom is treated as a normal reply. Note that this method does not detect C2-SPI phantoms. The TEU does not process SPI pulses in any way.

When the reply code and garble data have been shifted into the holding registers, a DATA READY signal is sent from the decoder to the reply assembler. The Data register is gated to the DECODER DATA BUS on the clock following DATA READY and the Garble register is gated to the bus on the next clock.

2.5.6 Reply Assembler

When a DATA READY pulse is received by the reply assembler, it latches the decoder data on the next clock and the garble data on the subsequent clock and stores them in buffer memories (see Fig. 2.5-15). There can be no conflict among decoders due to the restriction imposed by the reply detector that reply detections are separated by at least 3 clock intervals. The DATA READY pulse is also used to generate the ATCRBS RANGE STROBE to the MCU, and the resulting RANGE data is sampled along with the state of the MODE A/C line. Finally, a DECODER RESET pulse is sent to the decoder to clear it.

A count is kept of the number of replies stored. The buffer memory can hold up to four replies. If a DATA READY pulse is received when the buffer is full, a BUFFER OVERFLOW condition is latched and held until loaded into the buffer memory with the next reply stored. The data from the decoder signalling DATA READY is lost, no ATCRBS RANGE STROBE is generated, and the decoder is reset.

If the reply decoder controller sets the INSUFFICIENT DECODERS line this is latched by the reply assembler and held until loaded into the buffer memory with the next reply stored.

The transfer of replies to the CPU is governed by the state controller shown in Fig. 2.5-16, with state diagram as shown in Fig. 2.5-17. When a RESET pulse is received from the MCU the controller is set to state 0. This causes a MASTER RESET to be sent to all the decoders and to certain flip-flops within the reply assembler.

When the reply buffer contains one or more replies, as determined by a non-zero reply count, the controller begins a DMA transfer using the burst mode (see the description of DMA transfer in Section 2.9.4). First, the reply count is sampled through a reply-to-word-count translation ROM to obtain the number of words to be transferred in this burst. All replies in the buffer at

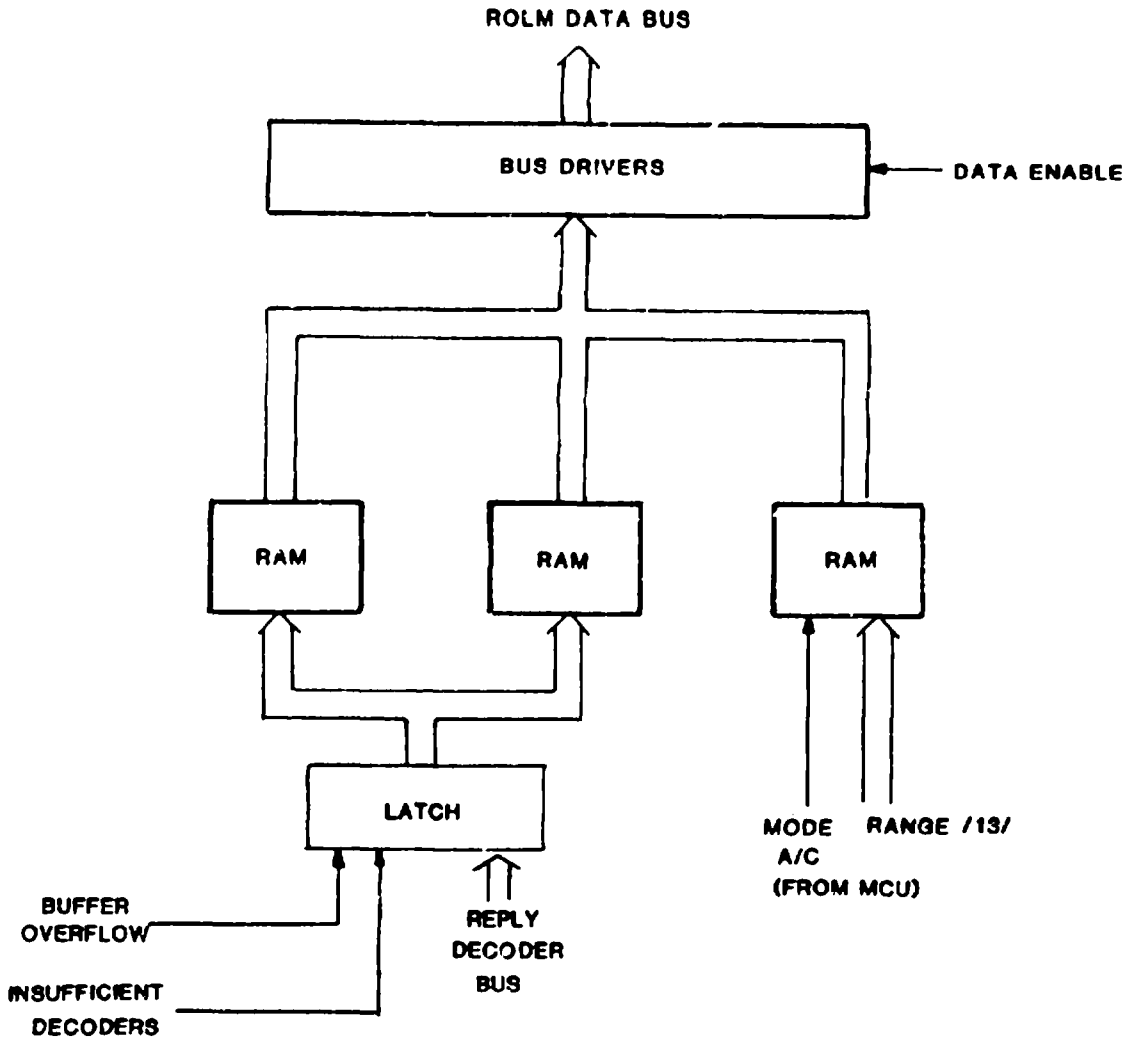


Fig. 2.5-15 Reply data output logic

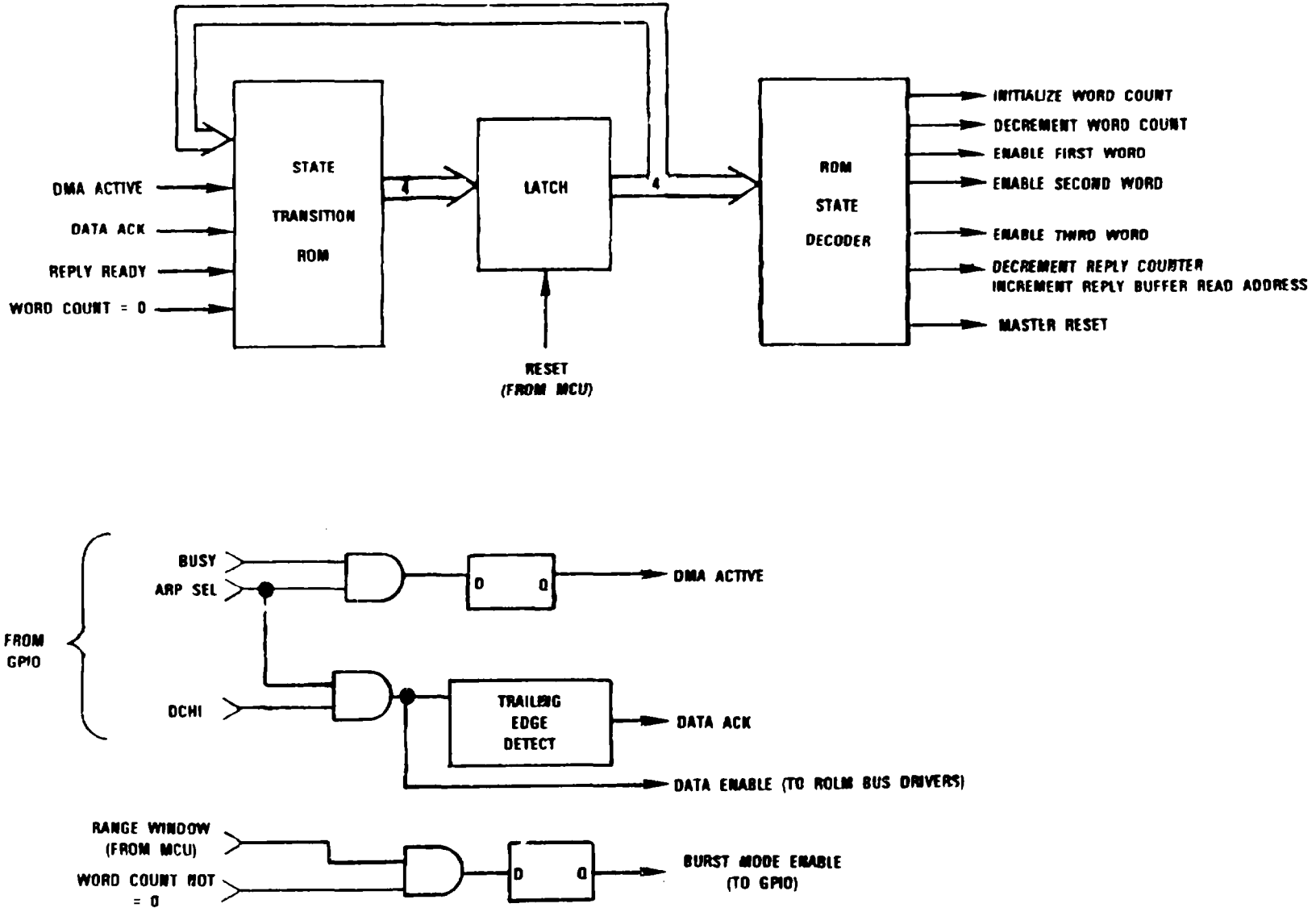


Fig. 2.5-16 Reply assembler state controller and associated circuitry.

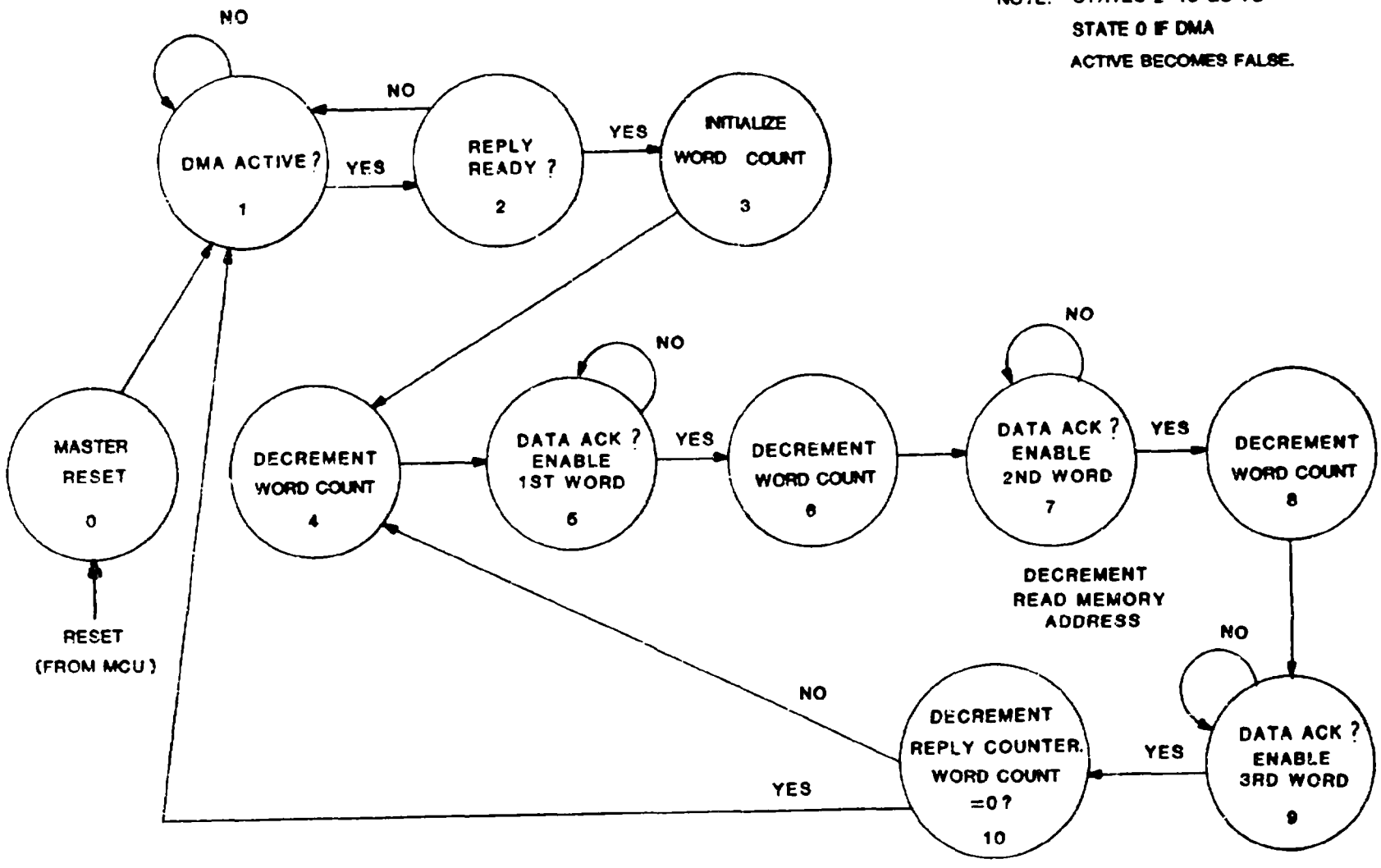


Fig. 2.5-17 State diagram of reply assembler controller.

the time of this sampling will be transferred. The number of words sampled is then transferred in groups of three, with the reply count being decremented after each group. Finally the controller returns to state 1 to see if more replies are ready for transfer.

Not shown in Fig. 2.5-17 are transitions from all states to state 0 if DMA ACTIVE becomes false. This is to take care of the case where the GPIO word count for the DMA transfer goes to zero and BUSY becomes false. This would cause the state controller to "hang up" in some state if each state did not test DMA ACTIVE.

2.6 Mode S Reply Processor

2.6.1 General Description

The TEU Mode S Reply Processor (MSRP), Fig. 2.6-1, processes both Mode S squitters and discrete replies, determining the range of discrete replies and decoding message blocks for both type of replies. In addition, for discrete replies, it checks the decoded address against the expected address and discards replies which do not decode correctly. Reply error correction is not performed.

Sampled Quantized Video (SQV) from the video digitizer is used for Mode S processing, except that the clock frequency used by the digitizer is 8.00 MHz instead of 8.27 MHz. The sliding window detector samples the SQV signal and provides valid pulse and clear leading edge signals to the preamble detector. A valid pulse is defined to be one in which three-out-of-four successive samples (125-ns samples) of the SQV are "1". A clear leading edge is defined as a valid pulse for which the SQV sample preceding the four samples of the pulse is a "0" and the first of the four samples of the pulse is a "1".

A Mode S preamble is declared by the preamble detector when a valid pulse is followed by three additional valid pulses located 8, 28, and 36 samples after the position of the first pulse and, in addition, at least two pulses have clear leading edges. The number of valid pulses and clear leading edges required for preamble detection is controlled by a ROM.

The detection of a Mode S preamble causes the following events to happen:

(1) The range time is recorded.

(2) The message processor logic is started to strobe the Chip Amplitude Compare (CAC) signal.

After each 16 samples, the data is stored in a 16-bit by 16-word Mode S reply RAM buffer. As the address of the reply becomes known (decoded by the parity decoder), it is compared with the expected address. If a discrete reply is expected and the reply address does not match the expected address the reply is discarded. The reply address is also stored in the reply RAM.

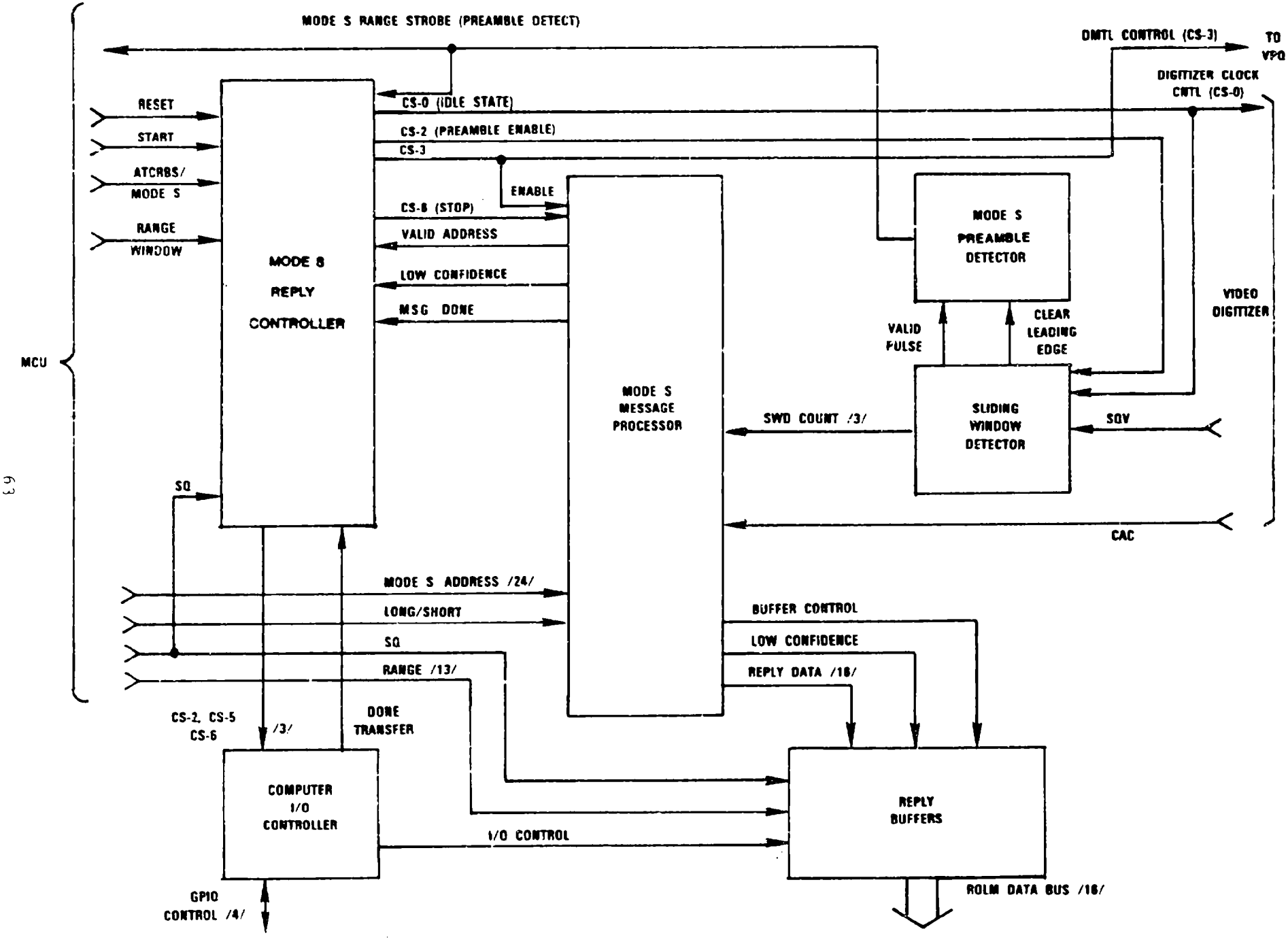


Fig. 2.6-1. Mode S reply processor block diagram.

Confidence bits are also computed using CAC data and the SWD COUNT provided by the sliding window detector. If the difference in number of SQV samples between the first and second chips exceeds 2, and the sign of the difference is consistent with the data bit (CAC sample), high confidence is declared. If not, a low confidence counter is incremented. If any message received during a squitter interval has more than 34 low confidence bits, the reply is discarded. This is a method for detecting and discarding squitter replies which are badly garbled by multipath such that their Mode S address fields are probably invalid.

An invalid message counter is set to zero at the beginning of each range window and incremented once whenever a message has more than 34 low confidence bits. The count is reported whenever a successful reply occurs and then reset to zero.

Each time a successful reply (valid discrete address or a squitter with less than 35 low confidence bits) is detected, a nine-word DMA transfer to the ROLM 1650 computer memory via the general purpose I/O interface (GPIO) will occur.

At the end of each range window a nine-word test reply is also transferred.

The formats used to transfer Mode S replies and test replies to software are as shown in Fig. 2.6-2.

2.6.2 State Controller

The Mode S reply processor is controlled by the state controller shown in Fig. 2.6-3 with state diagram as shown in Fig. 2.6-4.

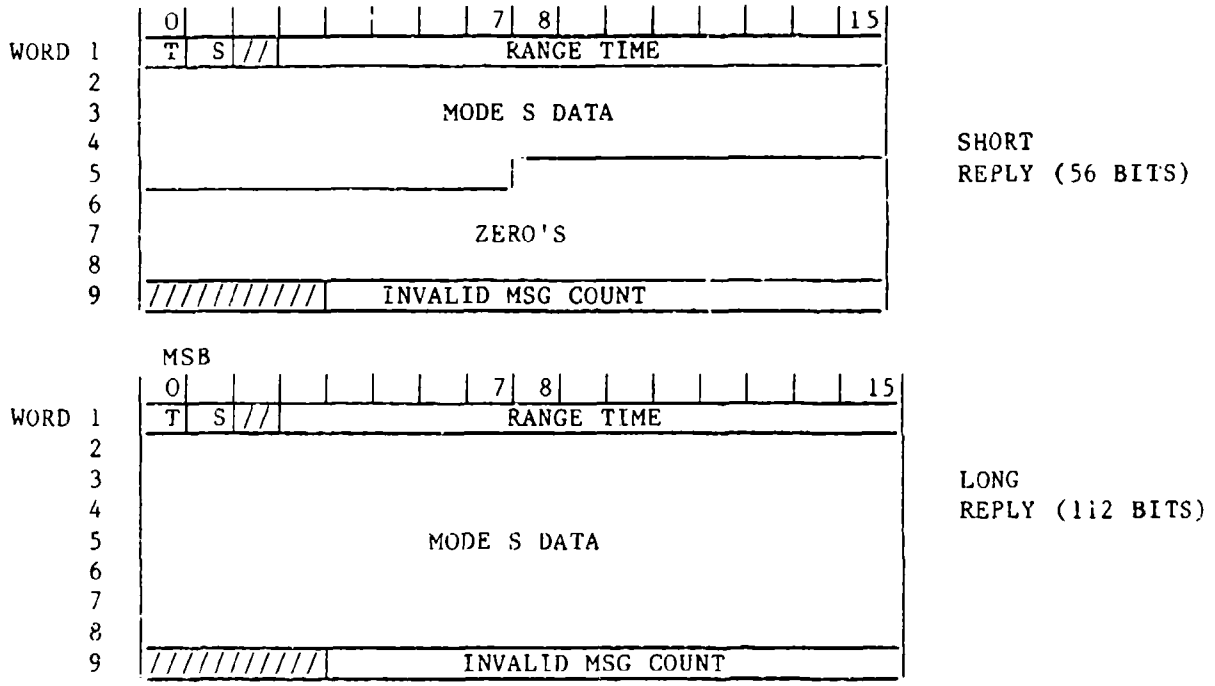
When power is first applied, the RESET signal from the MCU forces the controller to Control State Zero (CS-0). When a Mode S interval is required, the MCU establishes desired values of all mode bits and issues a START pulse. This causes a transition to CS-1 (a delay state). Departure from CS-0 changes the video digitizer clock from 8.27 to 8.00 MHz, required for Mode S processing.

Upon the opening of the range window, CS-2 occurs, which enables the Mode S preamble detector. The remaining control states may be traced using Fig. 2.6-4. Note that:

(a) Range window (RW) is continually monitored. At the termination of squitter mode, SQ=0 will cause RW=0 (see Fig. 2.2-8).

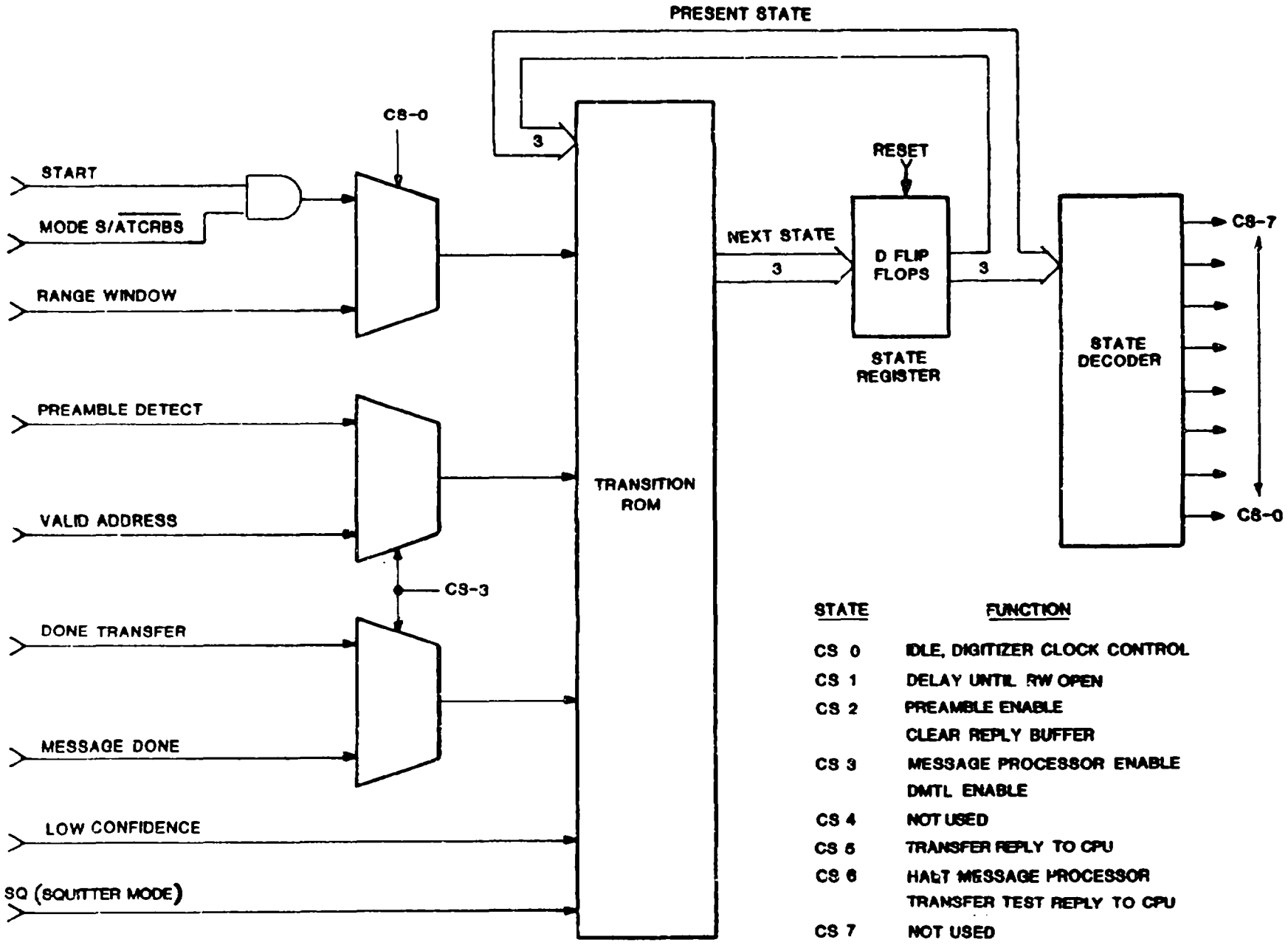
(b) If RW closes prior to the complete detection of a reply (MESSAGE DONE set), that reply is discarded.

(c) A test reply is only transferred at the end of a discrete (SQ=0) interval.



RANGE TIME	Round trip time + Mode S transponder delay + hardware delay. LSB = 125.0 ns. (Range count-1316) multiplied by 0.0101144 gives true range in nautical miles.
MODE S DATA	32 (88) bit Mode S reply data and 24 bit decoded address
S	1 = Reply processor in squitter listening mode. 0 = Reply processor in discrete reply mode.
T	Test Bit 0 = Real Reply 1 = Test Reply
INVALID MESSAGE COUNT	Number of preamble detections which were followed by messages having too many low confidence bits. Counter is reset each time a successful reply is transferred to software.

Fig. 2.6-2. Mode S reply processor data formats.



STATE	FUNCTION
CS 0	IDLE, DIGITIZER CLOCK CONTROL
CS 1	DELAY UNTIL RW OPEN
CS 2	PREAMBLE ENABLE
	CLEAR REPLY BUFFER
CS 3	MESSAGE PROCESSOR ENABLE
	DMTL ENABLE
CS 4	NOT USED
CS 5	TRANSFER REPLY TO CPU
CS 6	HALT MESSAGE PROCESSOR
	TRANSFER TEST REPLY TO CPU
CS 7	NOT USED

Fig. 2.6-3. State controller block diagram.

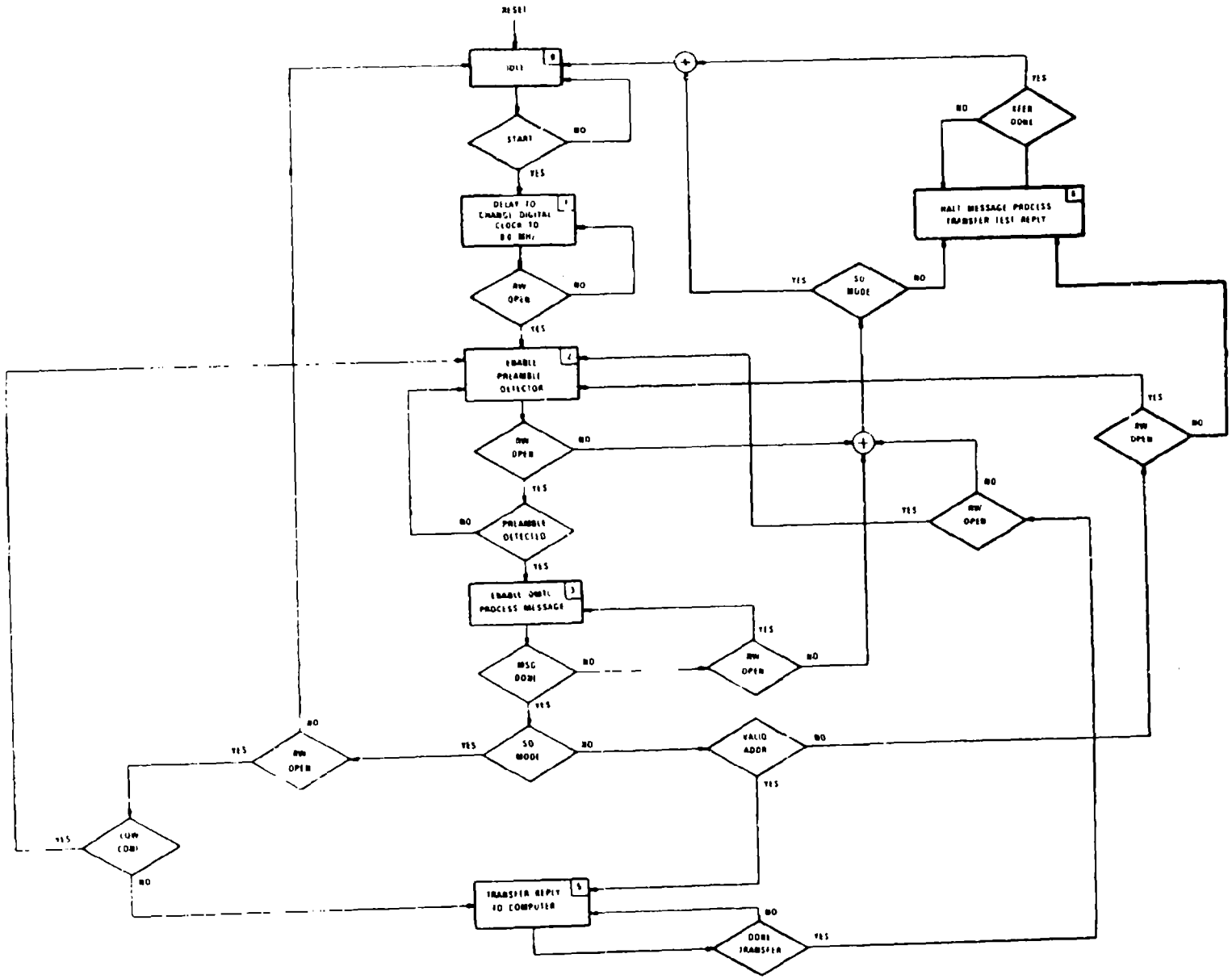


Fig. 2.6-4. State controller state diagram.

(d) A separate state controller is used during CS-5 and CS-6 to manage the transfer of replies to the computer buffer (see Section 2.6.9). If MODE S REPLY PROCESSOR SELECT becomes false during a transfer, the transfer state controller (Fig. 2.6-12) will immediately halt and issue a TRANSFER DONE. This may result in a partial transfer of a reply message.

(e) The software controlling the Mode S reply processor must provide a minimum of 80 microseconds following the end of a discrete range window to allow for the transfer of a discrete reply and the test reply.

(f) The software controlling a squitter interval may immediately terminate the range window and computer interface channel if desired. However, this may result in the partial transfer of a squitter reply received just prior to the end of the interval.

(g) Mode S preamble detections are only enabled during control state 2. Control States 4 and 7 are not used.

2.6.3 Sliding Window Detector

Mode S preamble detection and confidence declaration require that pulse energy be estimated. This is realized using a sliding window detector which continuously examines SQV and notes when SQV=1 in at least 3 of the 4 samples in the window. The logic diagram and timing for P4 (the last preamble pulse) is shown in Figs. 2.6-5 and 2.6-6 respectively. Note that the preamble detector is enabled at the gate which declares clear leading edges (i.e., the input to the preamble detector). This is necessary since message data would otherwise be present in the shift registers used in preamble detection immediately following the end of the message processing interval (the message data could easily synthesize a false preamble). Note also that the sliding window detector is enabled whenever the state controller is not in control state 0.

2.6.4 Preamble Detector

The preamble detector is shown in Fig. 2.6-7. It is enabled when clear leading edge (CLE) declarations are enabled at the sliding window detector (see Section 2.6.3). Two D flops spread the CLE signal to account for timing tolerances. Fig. 2.6-6 shows pertinent timing. Note that the detection of a preamble enables a transition to CS-3 which disables further preamble detection and starts the message processor module (see Fig. 2.6-4).

2.6.5 Message Processor

The detection of a preamble causes a transition from control state 2 to 3. This starts the message processor, shown in Fig. 2.6-8, which includes logic to perform the following:

(a) CAC sampling to obtain the data bits, Di.

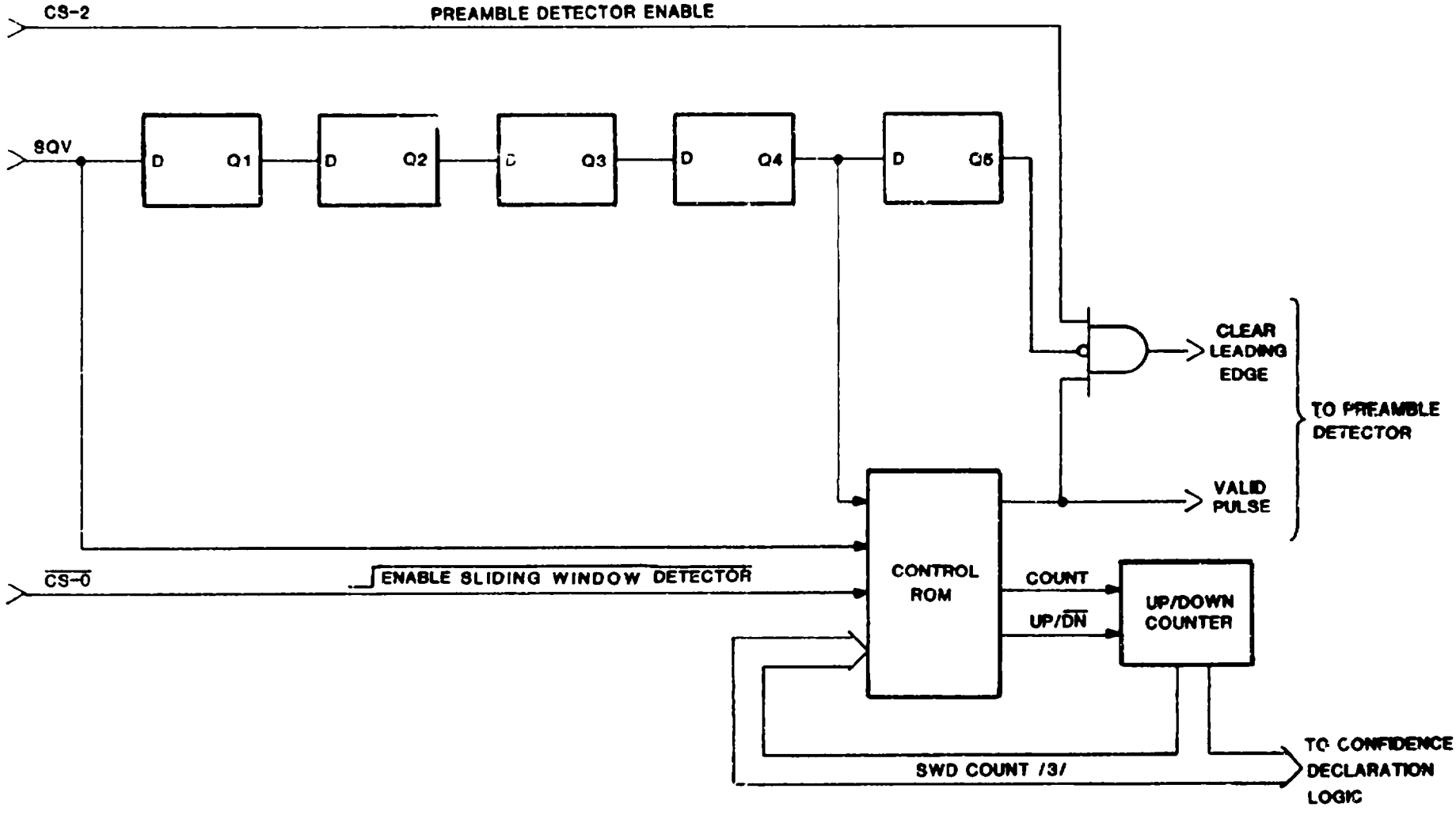


Fig. 2.6-5 Sliding window detector.

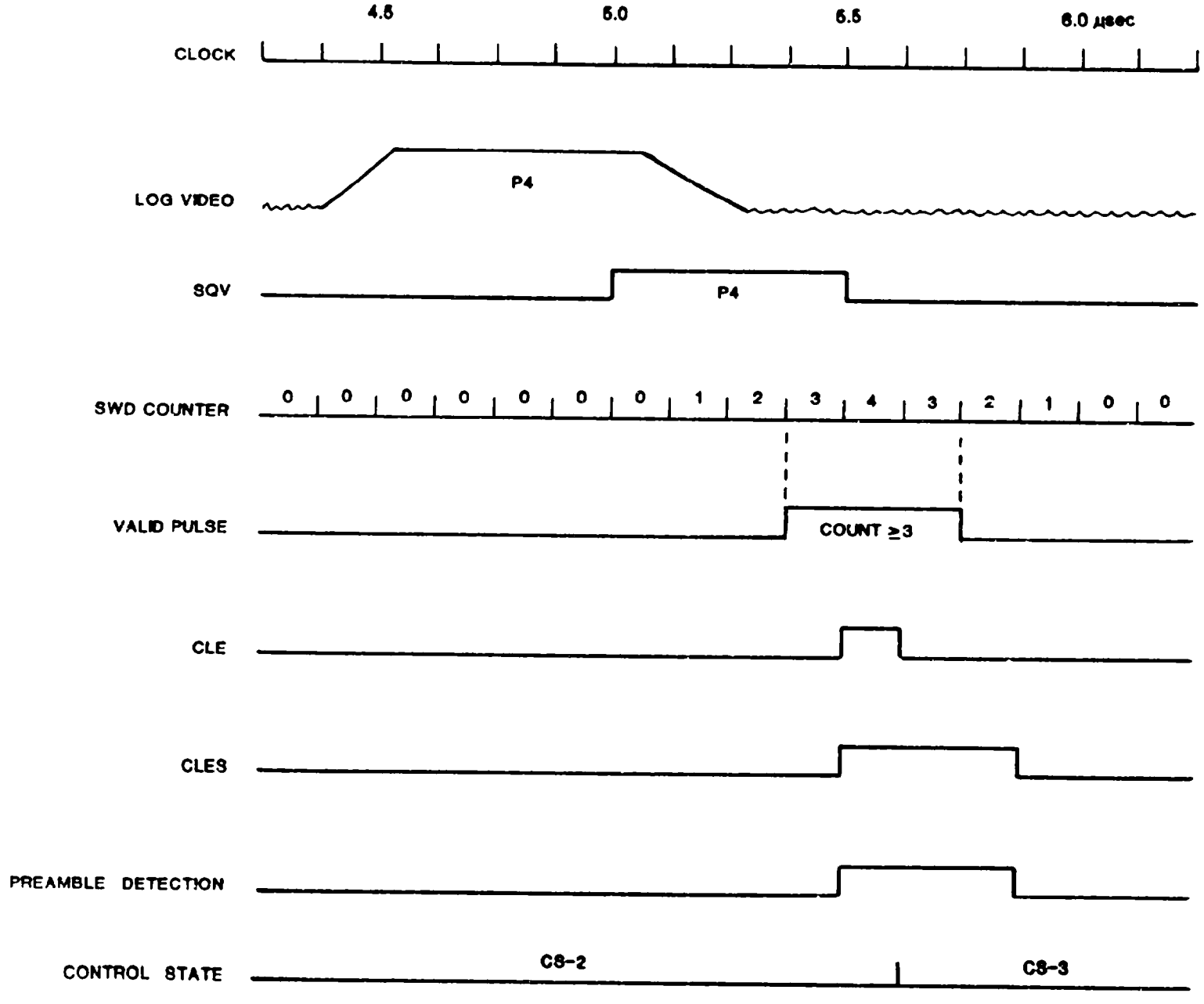


Fig. 2.6-6. Mode S preamble detection timing.

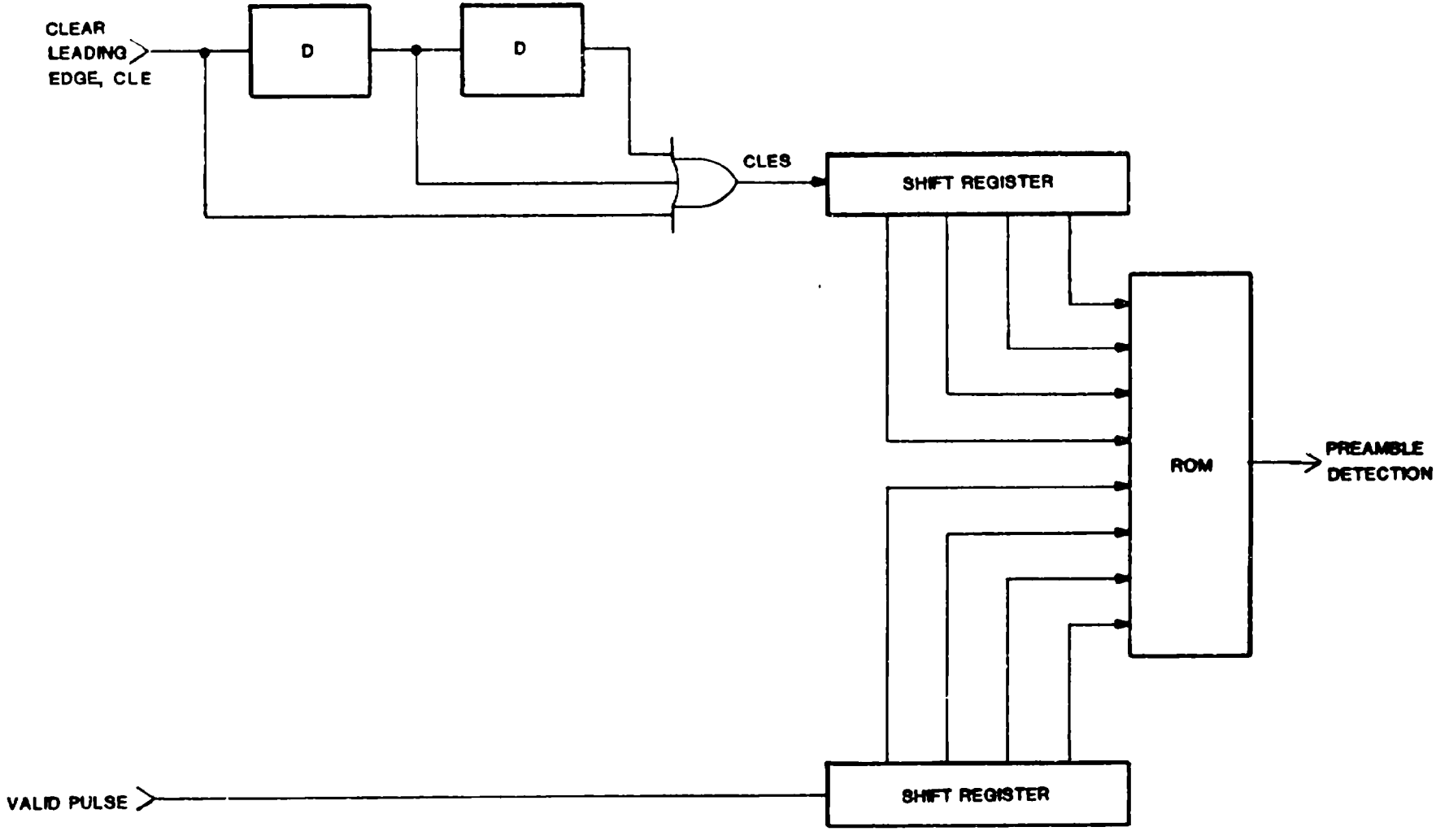


Fig. 2.6-7. Preamble Detector

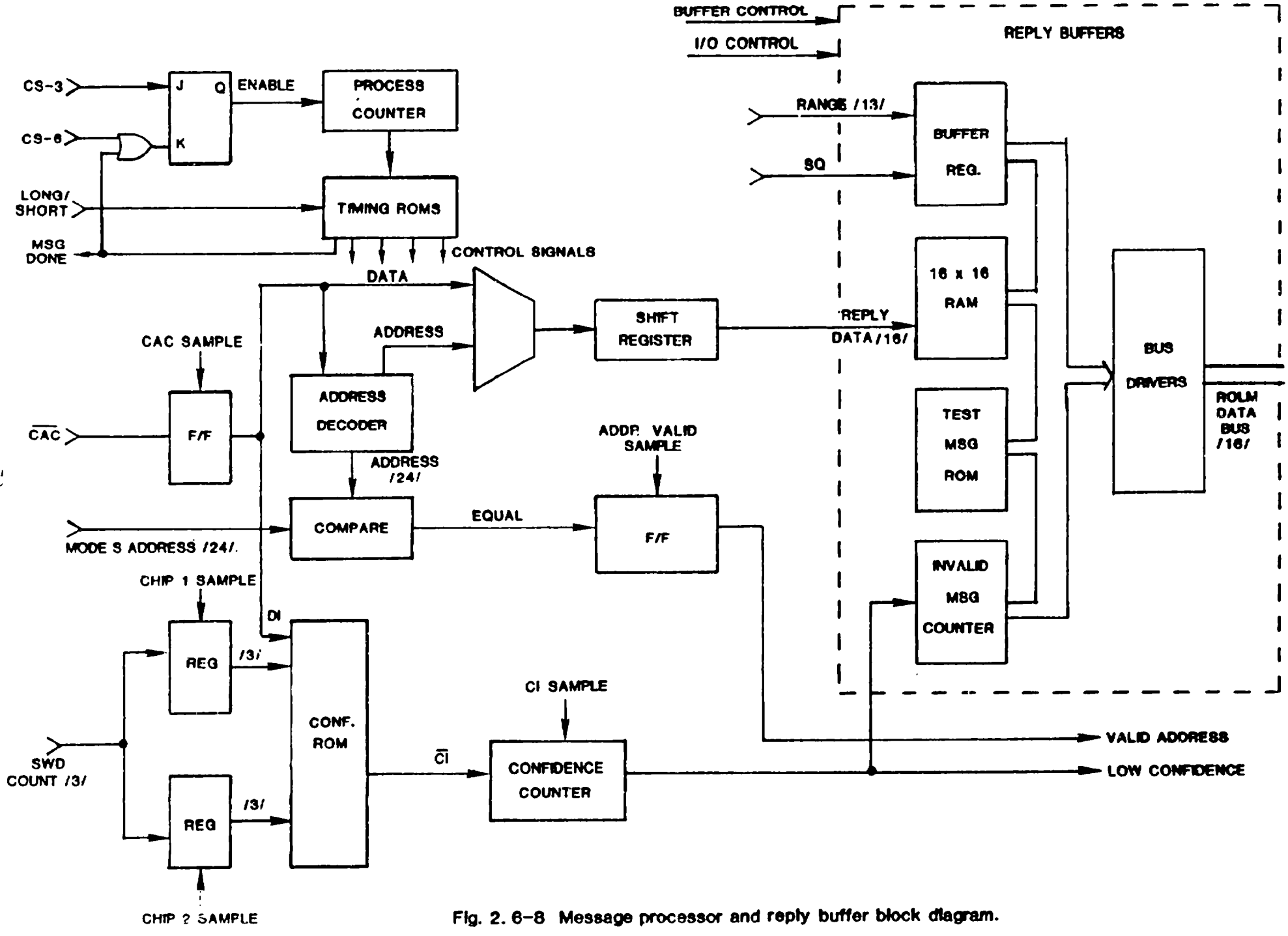


Fig. 2.6-8 Message processor and reply buffer block diagram.

(b) Store Di in 16-bit words in a 16 x 16 RAM.

(c) Compute the address from the received data and address/parity fields.

(d) Compare the decoded address with the expected address (if in discrete mode).

(e) Compute the confidence bits and declare invalid messages based on an excessive number of low confidence data bits (if in squitter mode).

(f) Count invalid messages.

The message processor accomplishes these functions using timing signals derived from a set of timing ROMs. Figure 2.6-9 shows the timing sequence for the first part of a data block. Conventional control signals shift the data and, later, the decoded address into a 16-bit shift register, as shown in Fig. 2.6-10 for a short reply. Long replies are transferred similarly, one transfer for each 16-bit segment.

The address decoder is an implementation of the Mode S reply error detector. As shown in Fig. 2.6-11, it consists of a feedback register with control gates. The following should be noted:

(a) The complete message (data and address/parity fields) is shifted into the address decoder. Feedback is enabled continuously during this interval.

(b) The decoded address, present after the last message bit input shift, is first compared with the expected address and the address valid flag set if appropriate. Next the address is transferred to the reply RAM by disabling the feedback and applying 24 additional shifts. A 2-to-1 multiplexer selects the decoded address instead of input data during this period.

(c) The initial value of the decoder shift register must be zero. This is achieved automatically using CS-2 (preamble detect enable state) which disables feedback, forces a zero at the decoder input and enables the shift registers continuously (8 MHz vs. 1 MHz).

As shown in Fig. 2.6-4 replies not completely processed are discarded if the range window terminates whether discrete or squitter. If however the message process sequence has been completed, the reply will be transferred to a software buffer provided one of the following is true:

(a) In a discrete mode, a reply is received having an address which matches the expected address.

(b) In a squitter mode, a reply is received having less than 35 low confidence bits.

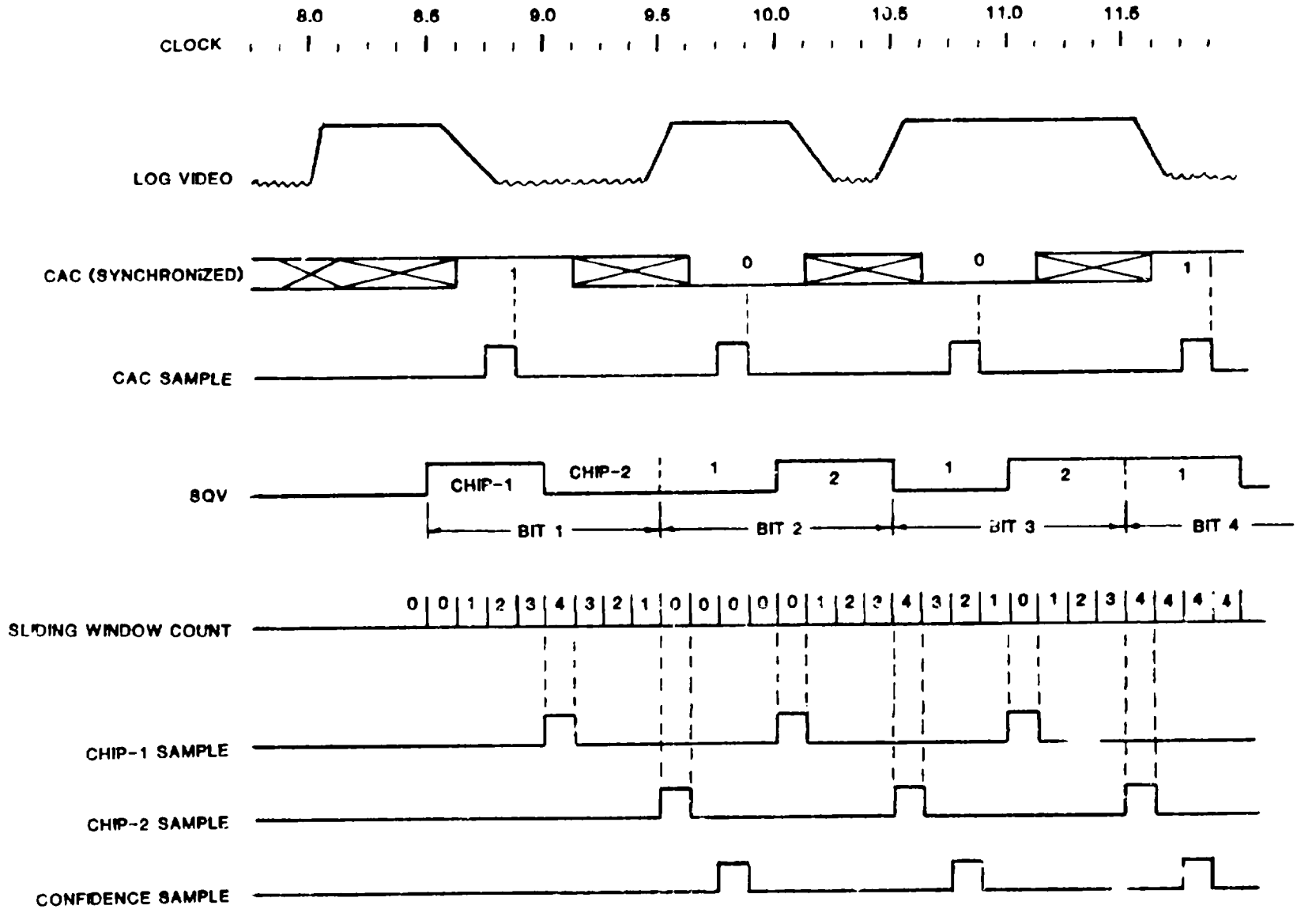


FIGURE 2.6-9 MESSAGE PROCESSOR TIMING.

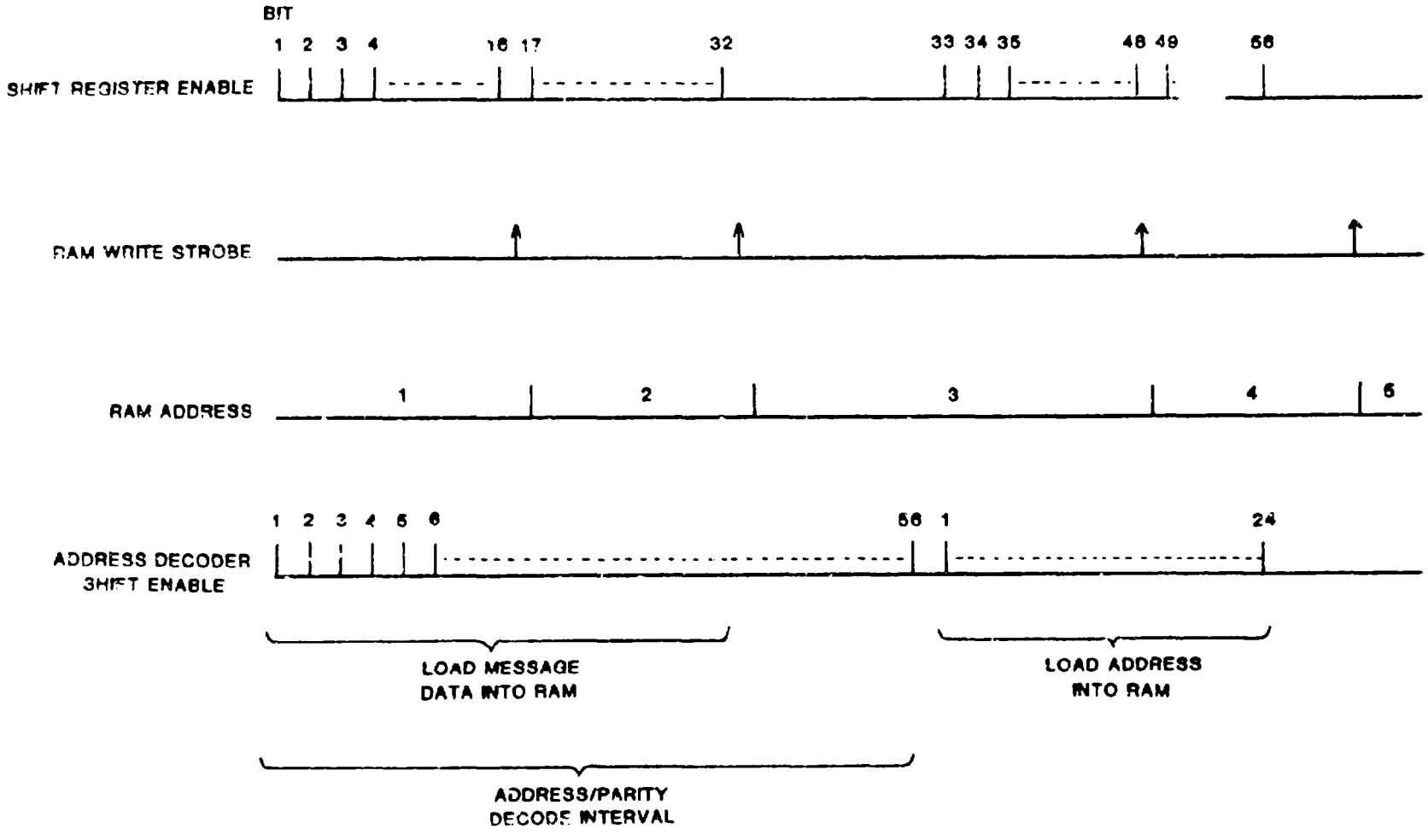


FIGURE 2.6-10 MESSAGE PROCESSOR TIMING

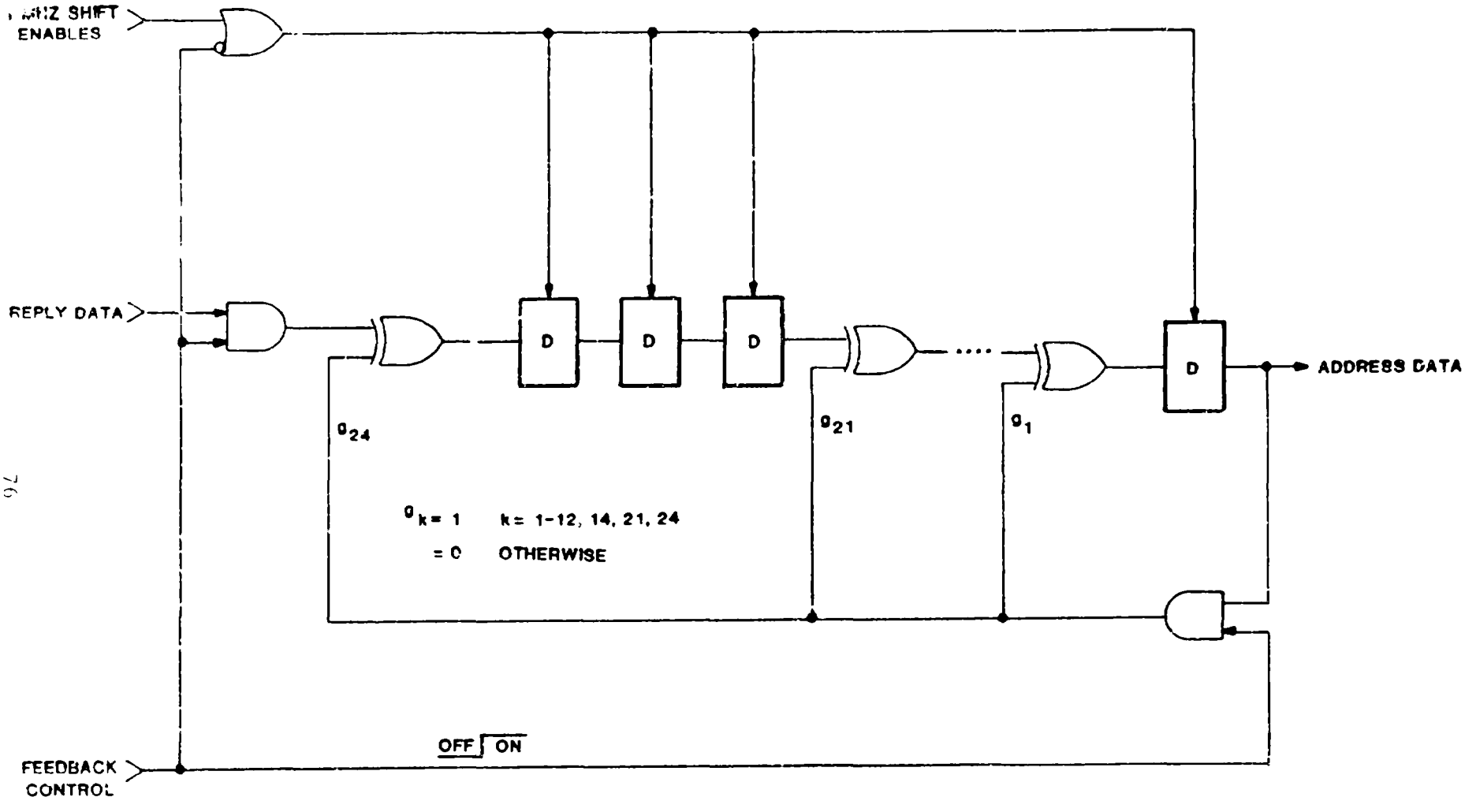


FIGURE 2.6-11 ADDRESS DECODER

2.6.6 Computer Interface Controller

The transfer of a discrete, squitter, or test reply from the MSRP to a software buffer is controlled by the computer interface controller, shown in Fig. 2.6-12. A state control diagram for this controller is shown in Fig. 2.6-13. A detailed timing sequence is shown in Fig. 2.6-14.

2.6.6.1 Discrete or Squitter Reply Transfer

The transfer of data begins when the Mode S state controller assumes state 5 or 6. (Note: the BUSY and MODE S REPLY PROCESSOR SELECT signals must be true when CS-5 or CS-6 occur). The computer interface controller then changes to state 1 in order to reset the RAM address counter. The controller next changes to state 2, which enables the range word. The RAM reset signal (state 1) is used after a delay as the first DONE signal. When a DCHI is received, indicating the first word has been transferred, the trailing edge of DCHI is used to: (a) increment the reply RAM to address 1 (address zero is not used), and (b) change to computer interface controller state 3. The first DCHI trailing edge pulse is also used as the second DONE.

The process is repeated such that during computer interface controller states 3-9, RAM words containing reply data are transferred, and during state 10 the invalid message count word is transferred. State 11 occurs next in order to generate a transfer done signal and to inhibit the 9th DCHI from generating an unwanted 10th DONE signal.

2.6.6.2 Test Reply Transfer

This process is identical to that described in Section 2.6.9.1 except that data is read from a set of test ROMs.

2.7 Angle-of-Arrival (AOA) Subsystem

In order to study the feasibility of providing TCAS traffic advisories based upon intruder bearing, as well as range and altitude, two TEUs were equipped with an angle-of-arrival receiver capable of measuring the angle-of-arrival of the reply from an intruder aircraft's transponder. The goal was to provide bearing accuracy of $\pm$15 degrees (1/2 clock position) or better in order to assist pilots in visually locating threatening aircraft. A much greater accuracy would be required before AOA data could be used in generating horizontal collision avoidance commands.

A block diagram of the TEU AOA augmentation is shown in Figs. 2.7-1 and 2.7-2. Two antenna arrays are used, one array on top of the TCAS aircraft, and one on the bottom. The four signals from each array are combined to form a Σ and Δ signal for each array. Either of the two arrays may be selected by means of a diversity switch. Interrogations are transmitted via the L port of the selected array, and replies are received on both the Σ and Δ ports. The AOA receiver has three outputs: one output is proportional to the sine of the phase difference between the Σ and Δ signals (this phase difference is linearly related to the angle-of-arrival of the incoming reply), one output is proportional to the cosine of this phase difference, and one output is log video.

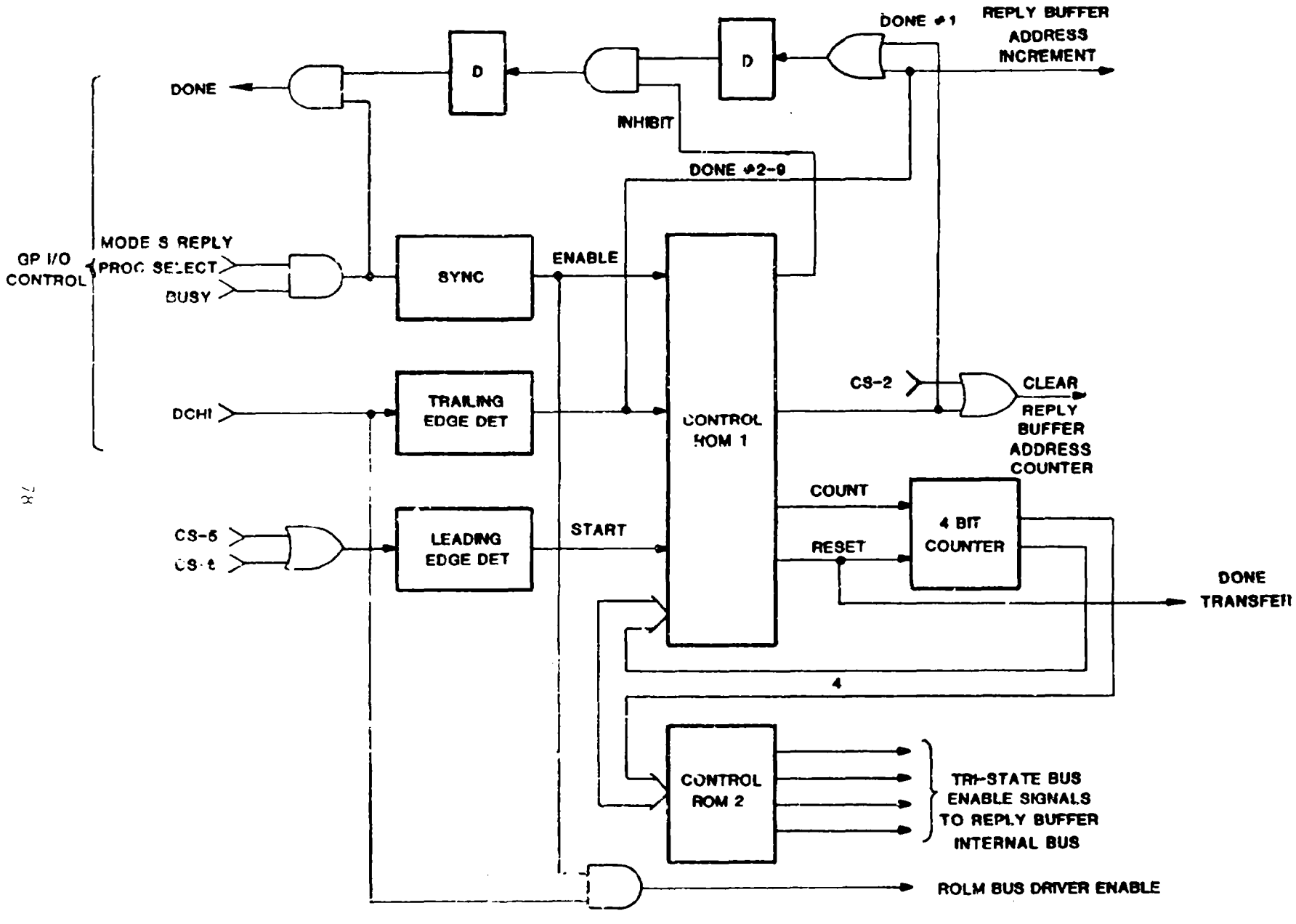


Fig. 2.6-12 Mode S I/O controller.

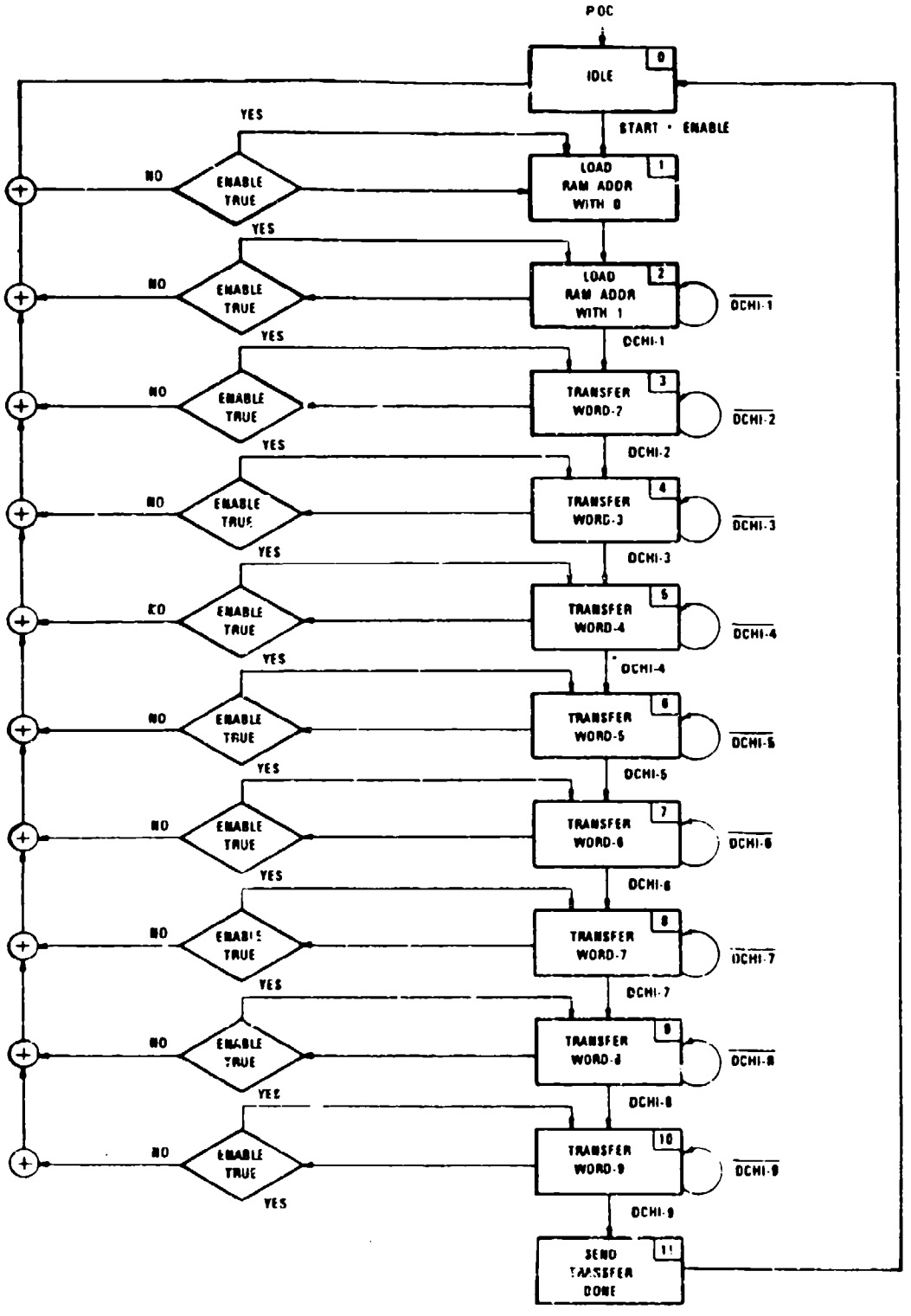


Fig. 2.6-13 MODE S I/O controller state diagram.

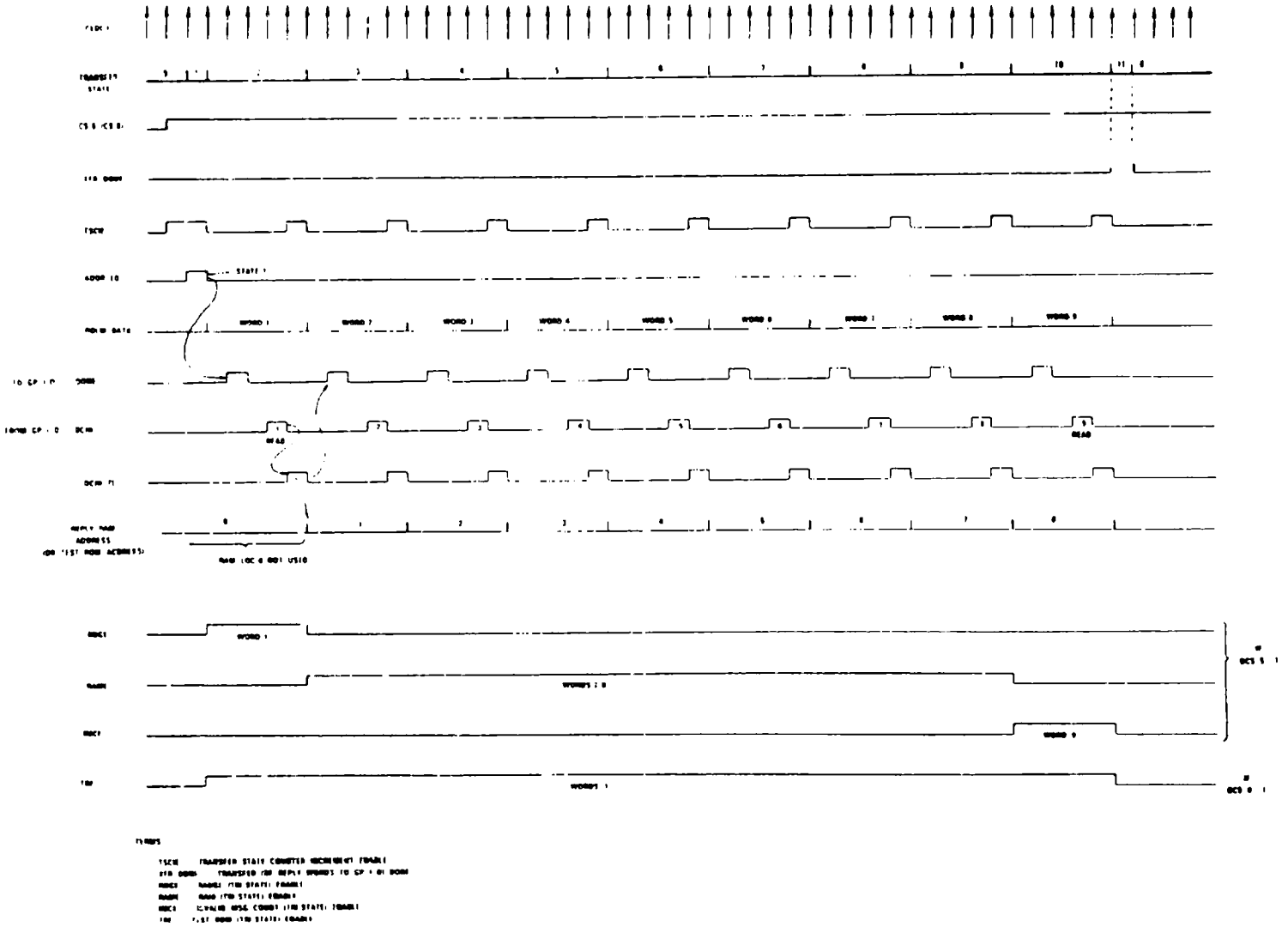


Fig. 2.6-14. Mode S I/O controller timing diagram.

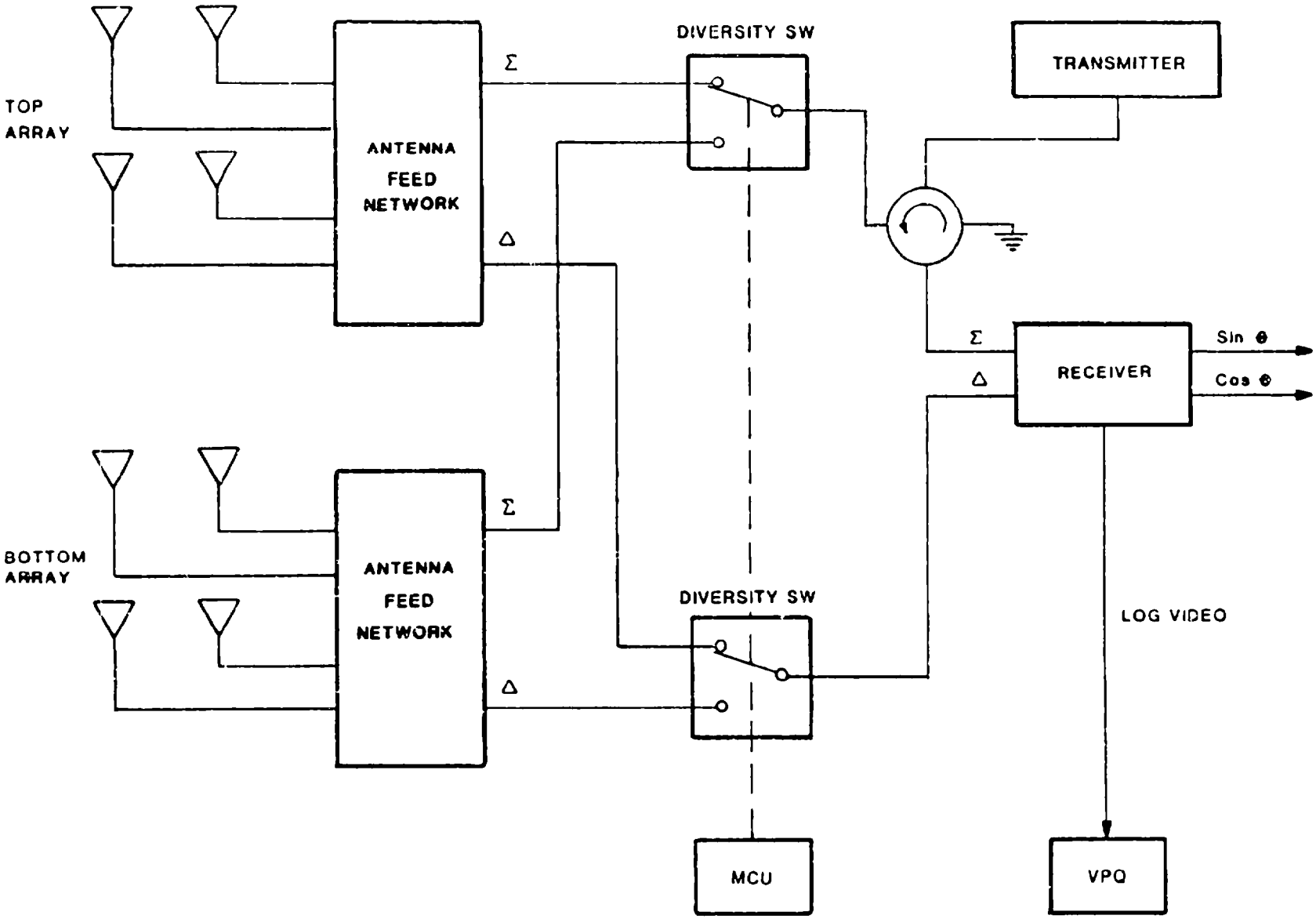


Fig. 2.7-1. TEU bearing augmentation - RF components.

100869

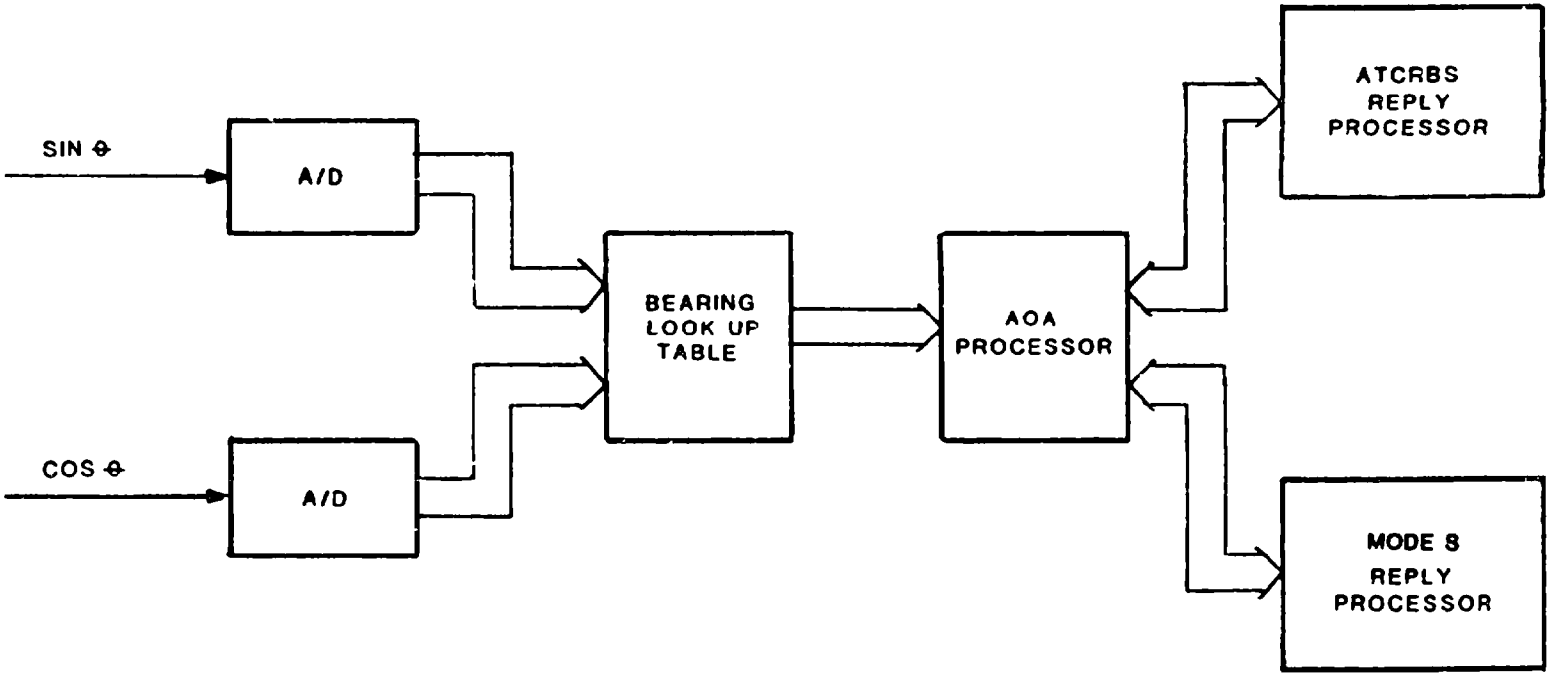


Fig. 2.7-2. TEU bearing augmentation – digital components.

The sine/cosine outputs of the receiver are converted to digital form by high speed A/D converters, and then a table lookup is performed to determine the corresponding angle-of-arrival. The angle-of-arrival measurements for individual pulses are associated with incoming replies by the AOA processor which passes its results to the Mode S reply processor and to the ATCRBS reply processor for transfer to the computer, along with the range, altitude, and other information for that reply.

A more detailed description of the angle-of-arrival hardware is presented in Sections 2.7.1 through 2.7.4. Section 2.7.1 deals with the AOA antennas, Section 2.7.2 with the AOA receiver, Section 2.7.3 with the AOA digitizer, and Section 2.7.4 with the AOA processor.

2.7.1 Antenna and Antenna Feed Network

The angle-of-arrival antenna consists of a square array of four L-band monopole elements spaced a quarter-wavelength apart. Figure 2.7-3 shows the orientation of the array elements. In the Cessna 421 aircraft the antenna elements are individually mounted on the aircraft surface with a separate pressurized RF bulkhead connector for each monopole. The antenna feed network, shown in Fig. 2.7-4, is located directly inside the aircraft skin. It consists of a set of four L-band hybrids connected as shown. There are no active elements located at the antenna. The Σ and Δ outputs are fed to the angle-of-arrival receiver through a pair of phase-matched RF cables.

The antenna pattern for such an antenna is shown in Fig. 2.7-5 (based on measurements in an anechoic chamber, using a circular ground plane of 3-ft. diameter). It can be seen from this pattern that the phase difference between Σ and Δ is an approximately linear function of the angle-of-arrival of the received signal. The amplitude of the Σ channel is nearly independent of the azimuth angle-of-arrival of the received signal.

Antenna performance as a function of elevation angle is presented in Fig. 2.7-6. The gain pattern is essentially the same as that of a single monopole antenna, except about 3-dB lower in absolute gain. It can be seen that the phase and amplitude responses of the antenna appear to be reasonable for the purposes intended at elevations up to about 40°.

2.7.2 AOA Receiver

A second receiver was added to the existing TEU RF Box to process the angle-of-arrival signals. The block diagram of the twin receivers is shown in Fig. 2.7-7. The IF outputs of both receivers are fed to a pair of matched limiting amplifiers to ensure that their amplitudes are identical over the entire dynamic range of the receiver. The limited IF outputs are then phase detected, the output of the phase comparator being a pair of video signals whose amplitudes are proportional to the sine and cosine of the angle-of-arrival of the reply. These video pulses are digitized by the AOA digitizer and then processed by the TCAS AOA processor.

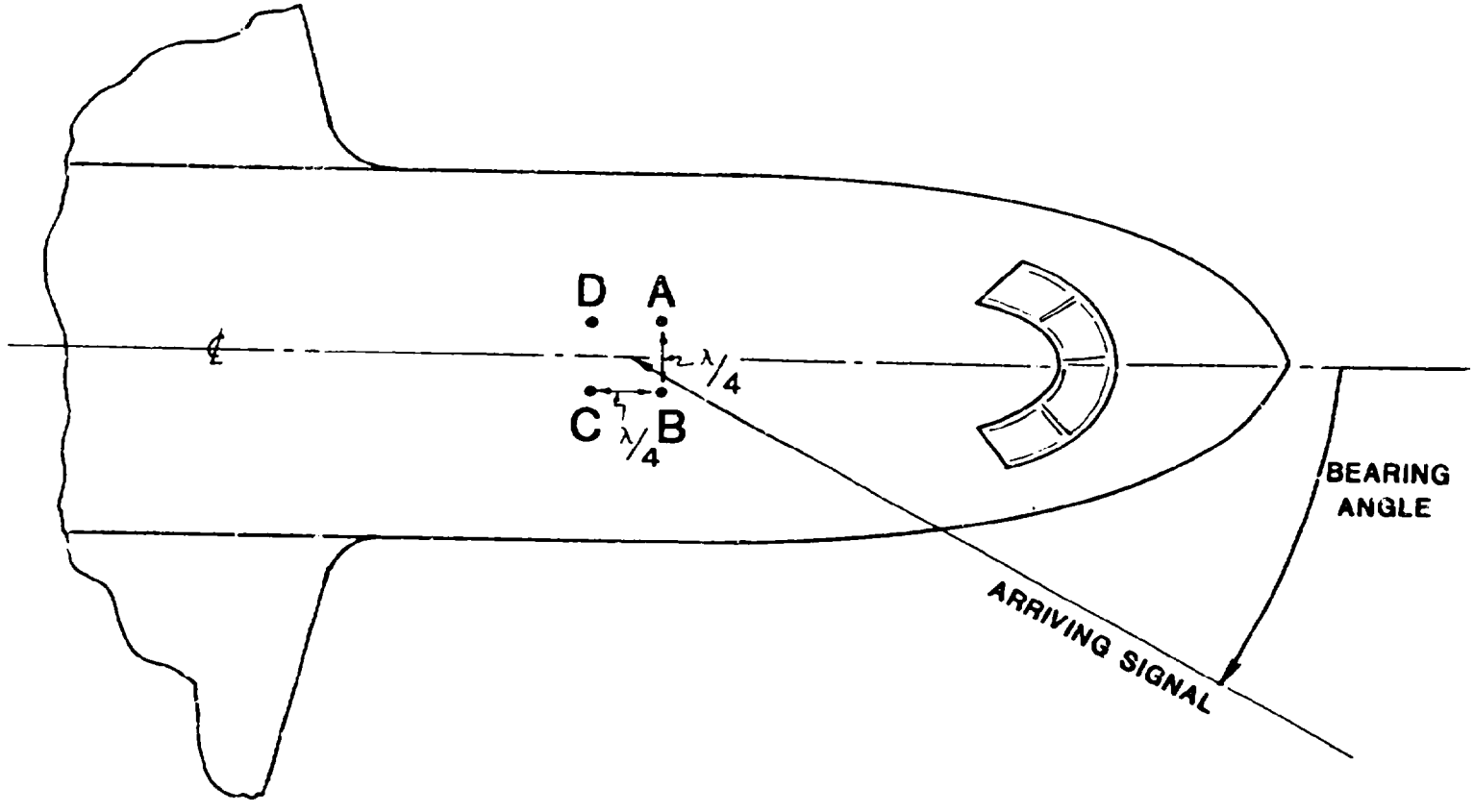


Fig. 2.7-3 AOA antenna orientation.

100870

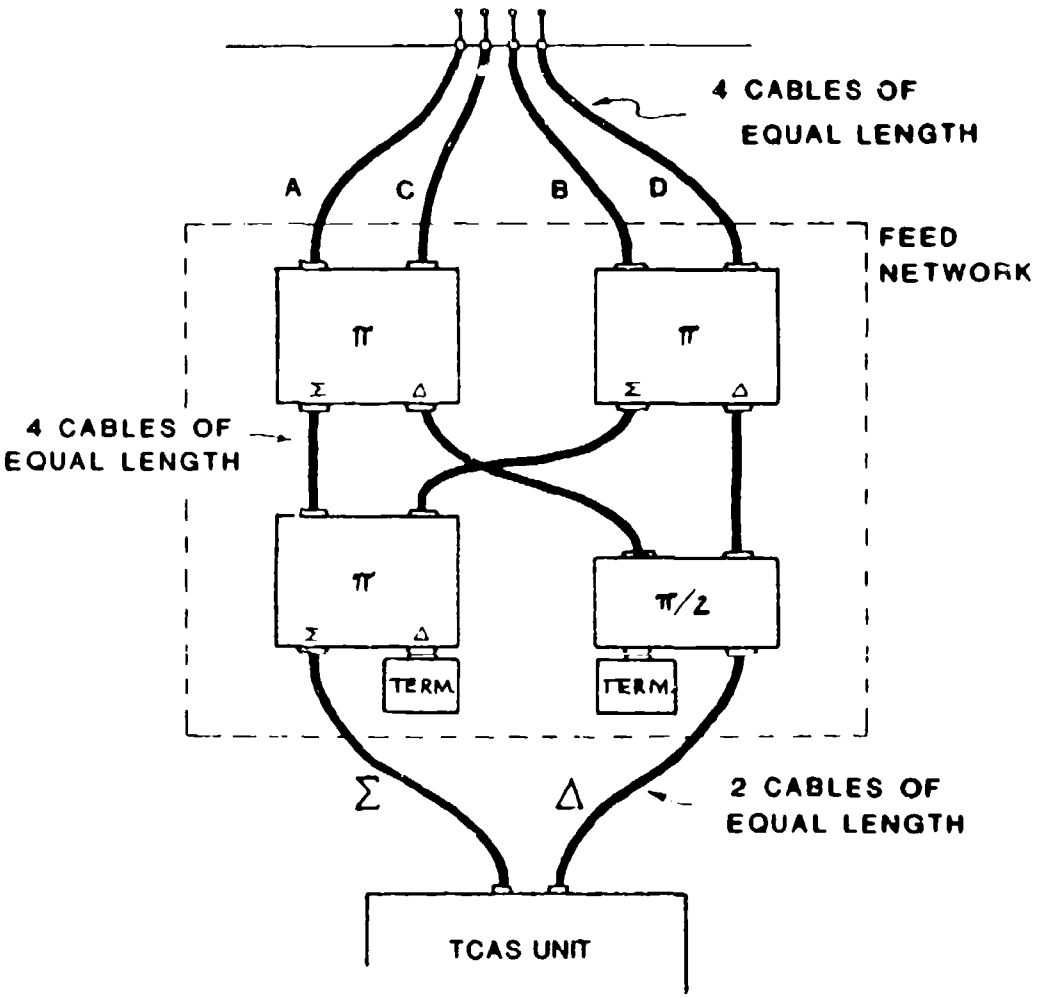


Fig. 2.7-4. AOA antenna feed network.

100868

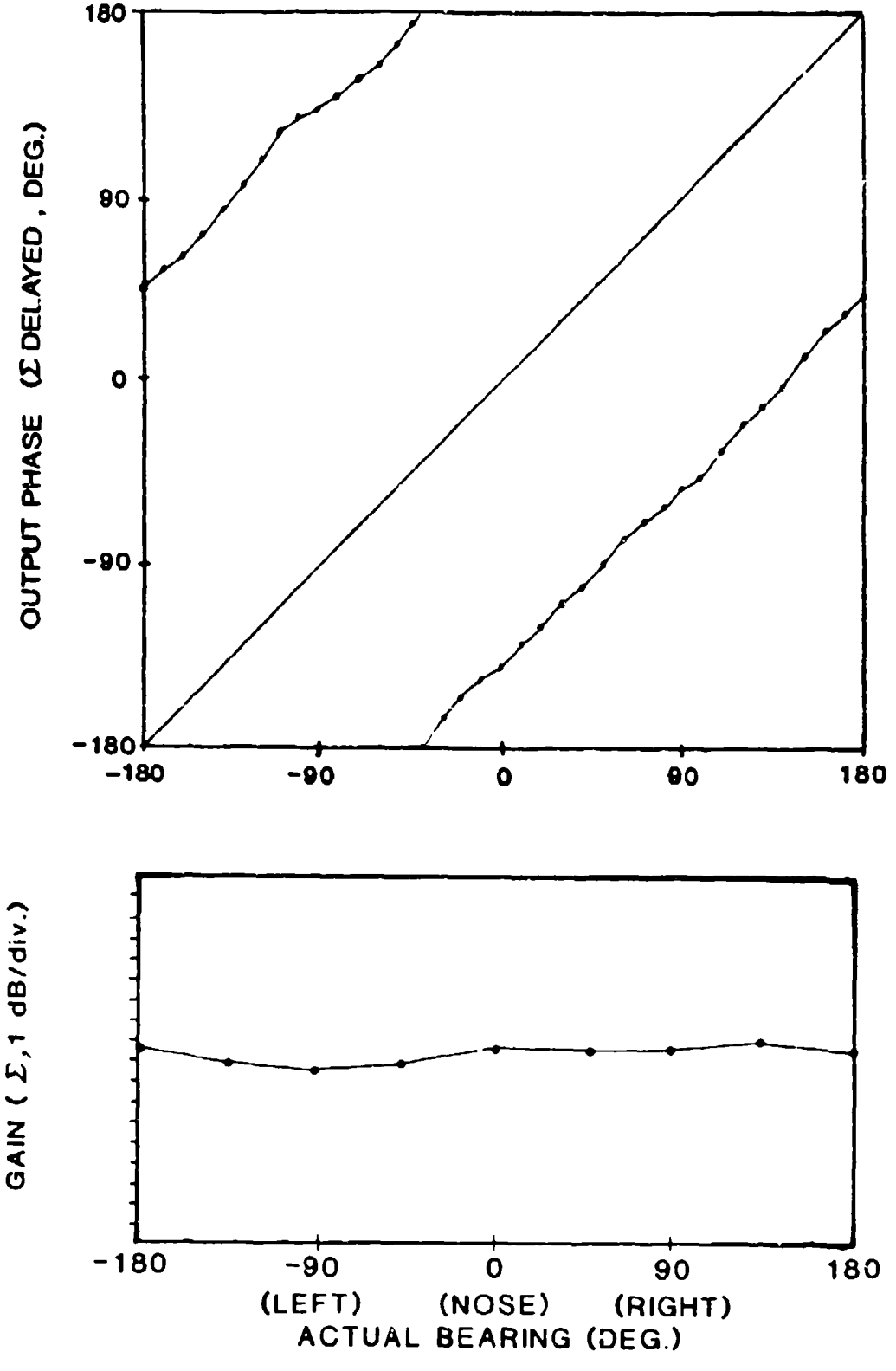


Fig. 2.7-5. Phase and amplitude response of the AOA antenna versus azimuth. 100867

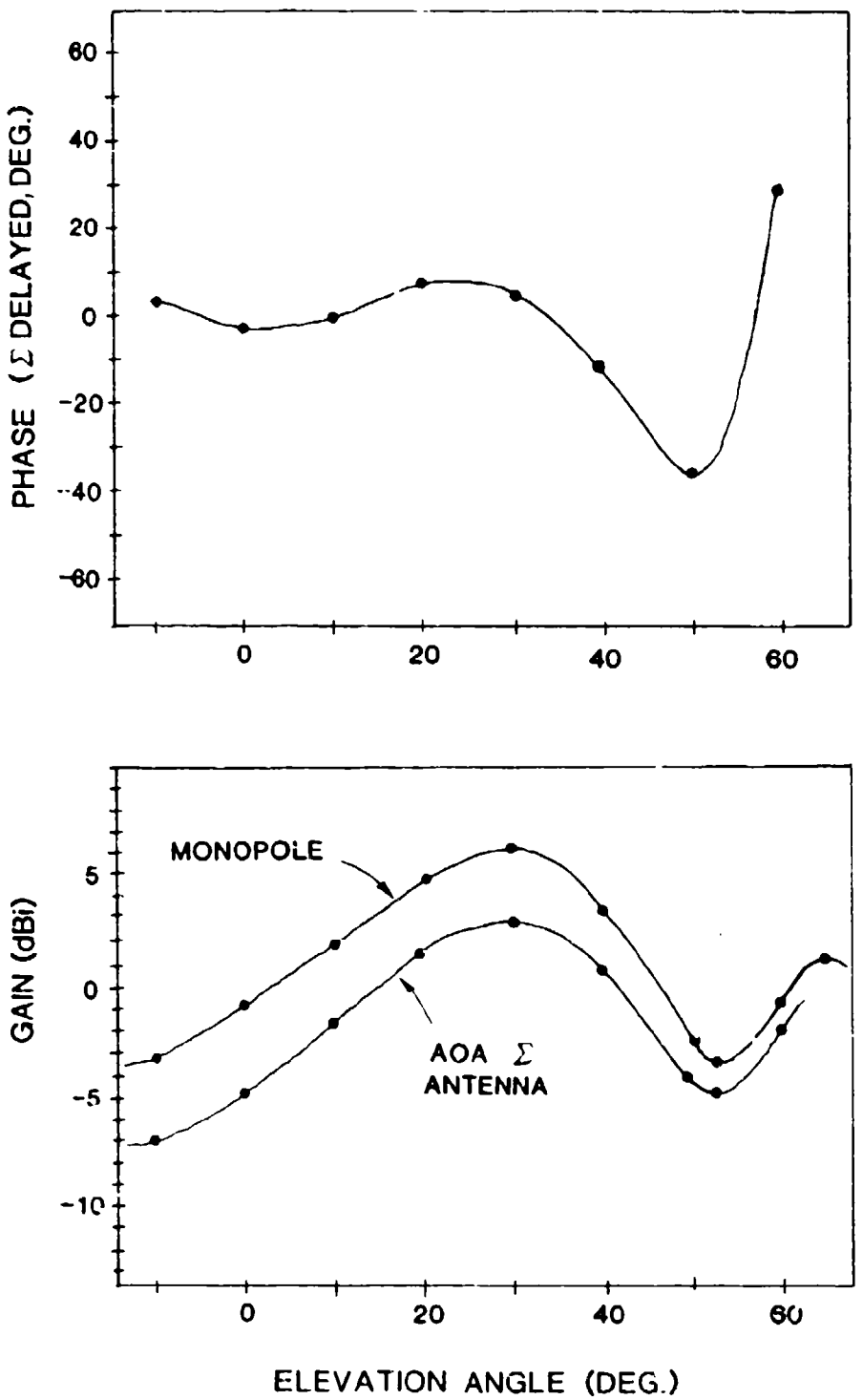


Fig. 2.7-6. Phase and amplitude response of the AOA antenna versus elevation angle.

100881

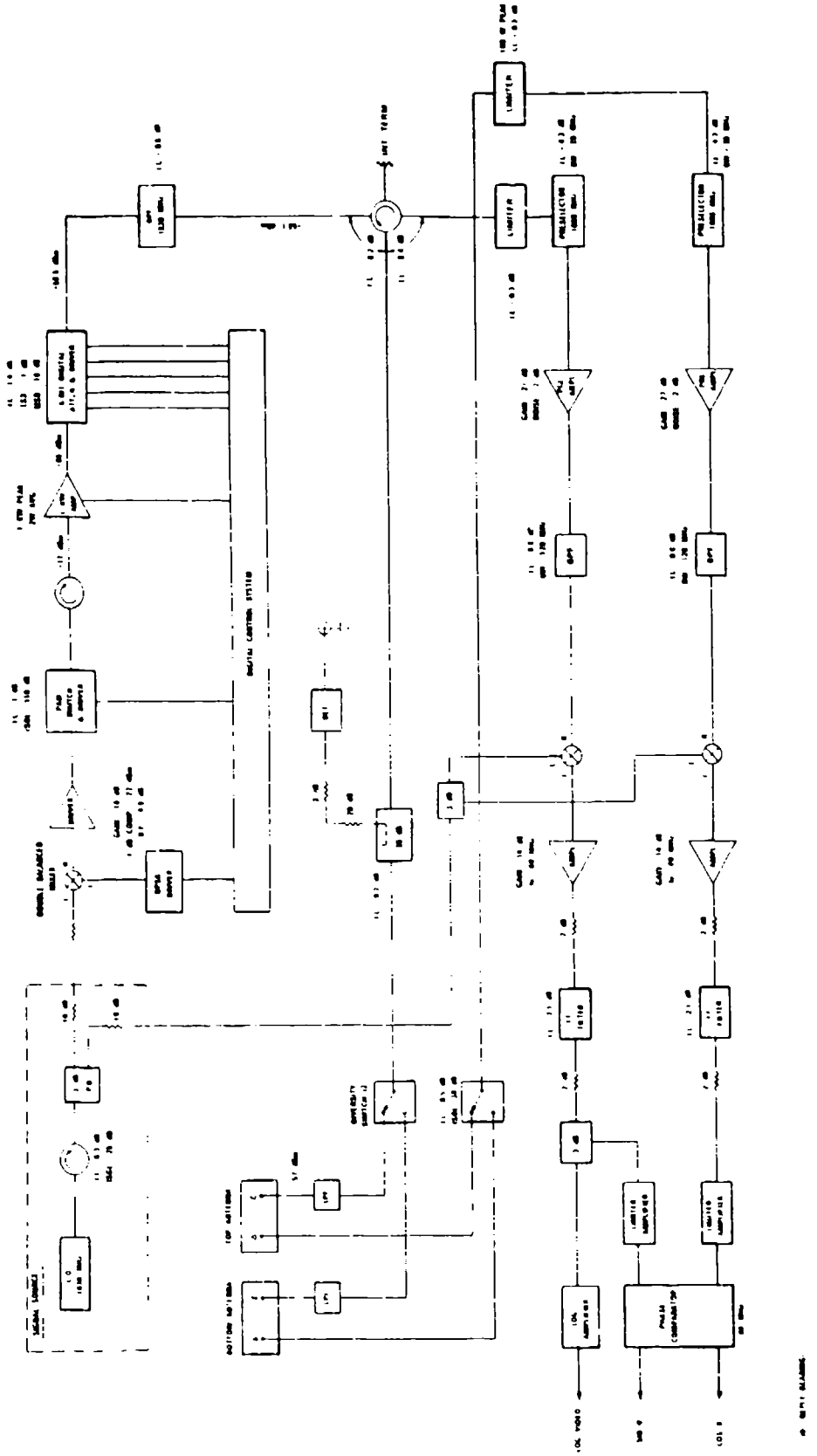


FIG. 2.7-7. TEU receiver block diagram.

The two principal sources of systematic receiver error are the phase comparator and the limiting amplifiers. The phase comparator has an error component which varies as a function of both the phase difference being measured, and the frequency of the signal being received. The phase comparator used in the AOA receiver is specified to have an error of less than ± 5° at all phases and frequencies within the receiver bandwidth. For the limiting amplifiers, the insertion phase varies with signal level, and any mistracking of insertion phase between the two amplifiers will result in a differential phase error. For this reason, matched amplifiers were used in the AOA receiver, and these are specified to have a differential phase error of less than ± 2.5° over their entire dynamic range. Therefore, a realistic figure for the total receiver phase accuracy would be about ± 5.6°, ignoring phase bias.

2.7.3 AOA Digitizer

A block diagram of the AOA digitizer is shown in Fig. 2.7-8. The sine and cosine video from the receiver is low-pass filtered to reduce high frequency noise and then buffered by high-speed operational amplifiers which invert the signal and transform the -400-mv to +400-mv signal from the phase comparator into a 0- to -1-volt signal for A/D conversion. Typical waveforms before and after buffering are shown in Fig. 2.7-9. A/D conversion is accomplished by a pair of 8-bit 30-MHz flash converters operating at the TEU digitizer clock frequency (8 MHz Mode S, 8.27 MHz ATCRBS). One sample of each digitized sine and cosine is available each clock period.

2.7.4 AOA Processor

A block diagram of the AOA processor is shown in Fig. 2.7-10. The digitized sine and cosine video pulses are used to address a prestored angle code in the bearing look-up table. Since the digitized sine and cosine video is available continuously, it is necessary to read the code precisely when a reply pulse is present. The AOA control logic does this in conjunction with the ATCRBS and Mode S reply processors. These reply processors independently examine the received pulse streams to determine the proper time to sample each code pulse. Since it is necessary to wait for the reply processors to perform reply "bracket" detection (ATCRBS) and preamble detection (Mode S) before the position of the code pulses is known, the strobe to sample the angle-of-arrival pulse will occur long after that pulse is present. For this reason, the angle-of-arrival samples coming from the lookup table are delayed by an amount equal to the processing lags in each reply processor. This delay is accomplished digitally, with a different delay value for ATCRBS processing than for Mode S processing. The AOA control logic must also discern the identity of the pulse being sampled, that is, to which reply it belongs since up to four interleaved replies can be processed simultaneously by the ATCRBS Reply Processor. This is a trivial process for a Mode S reply since only one reply at a time may be processed. This identification is denoted DA in Fig. 2.7-10, and consists of a 2-bit binary code, a unique code for each of the four reply processors (the arbitrary code 01 is used for a Mode S reply.) When a pulse has been correctly sampled, and its ID is known and it is determined that the pulse is ungarbled, the SEN line is pulsed to enable the remainder of the AOA processor.

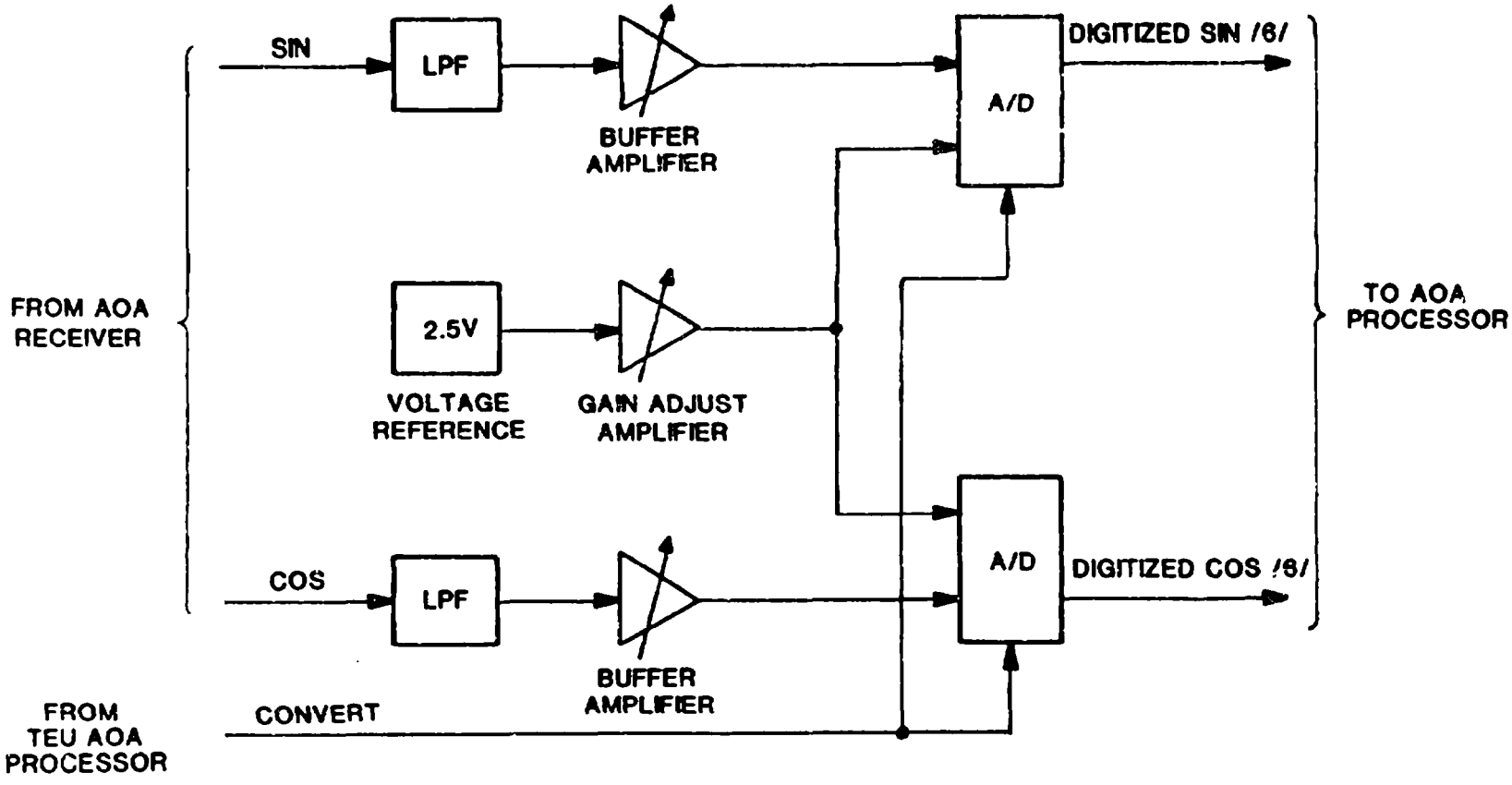
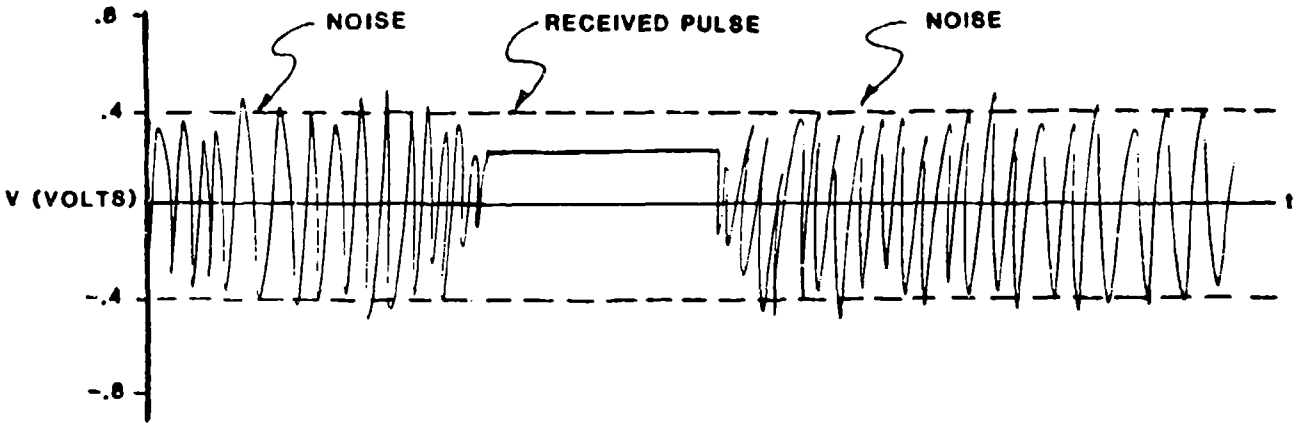
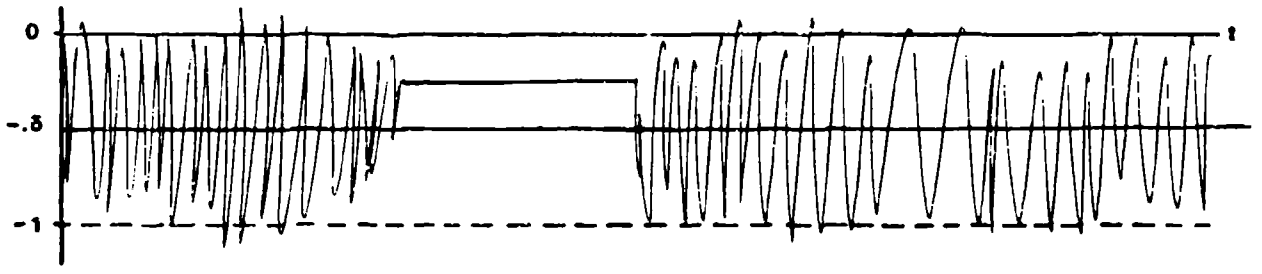


FIGURE 2.7-8 TEU AOA DIGITIZER



a). Sin, Cos Waveforms input to digitizer



b). Buffered Sin, Cos Waveforms for A/D Converter

FIGURE 2.7-9 AOA DIGITIZER WAVEFORMS

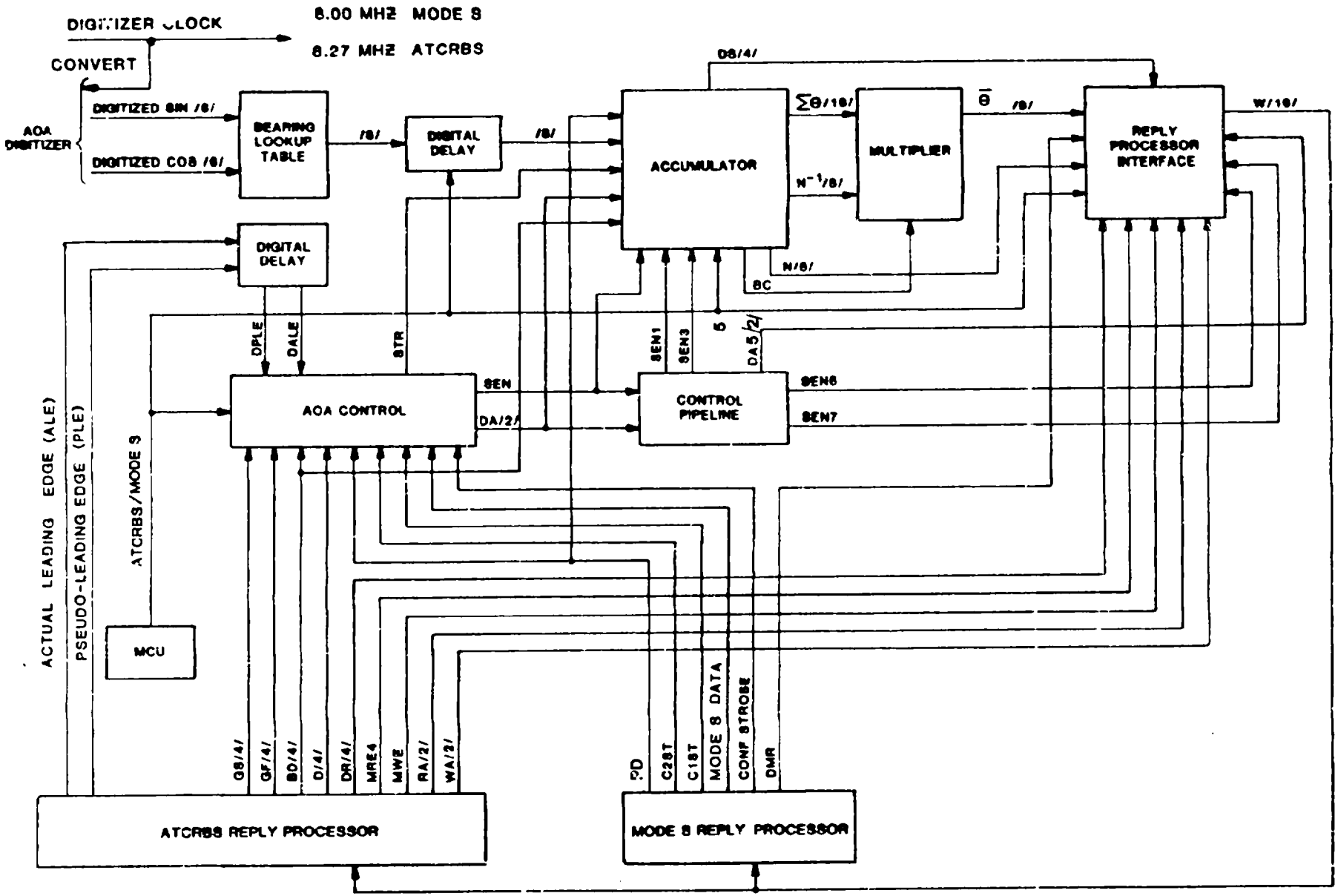


Fig. 2.7-10. TEU AOA processor.

Initiated by the SEN pulse, the accumulator adds the new angle sample to the running sum for the reply indicated by DA which is output as $\Sigma\theta$ in Fig. 2.7-10. It will also increment a counter to keep track of the number of pulses included in the sum of the angle samples. This count is output to the reply processor interface to be included in the data transfer to the appropriate reply processor (N in Fig. 2.7-10). The accumulator also keeps track of whether any pulses have been sampled for a given reply, and outputs this information to the reply processor interface (via the four DS lines in Fig. 2.7-10, one line indicating the status of each reply). The pulse count (N) is inverted (N^{-1} in Fig. 2.7-10) and passed to the LSI multiplier. This high speed, 16-bit device multplies $\Sigma\theta$ by N^{-1} (in essence dividing $\Sigma\theta$ by N) computing the average of the angle samples for the reply being processed, based on as many pulses as it has received at that point. This average is passed to the reply processor interface to be included in the data transfer to the appropriate reply processor. The current angle estimate ($\bar{\theta}$ and N) is stored in the reply processor interface for each reply, and is continually updated as new pulses for each reply are processed. When a reply processor finishes processing a reply, it requests the angle-of-arrival data from the reply processor interface which then transfers a 16-bit word containing the most current values of $\bar{\theta}$ and N, and a flag indicating whether any sample was made for this reply. This 16-bit data word is denoted by W in Fig. 2.7-10.

Control signals from the AOA control unit that control the accumulator and reply processor interface are propagated down the control pipeline which delays them by the processing lags in the accumulator, multiplier, and reply processor interface so that they are current at each step in processing.

2.7.4.1 AOA Control

The principal function of the AOA control unit is to determine which pulses should be sampled, and when these samples should be taken. This is accomplished by listening to the reply processors to determine when pulses are present and which replies they belong to. A block diagram of the AOA control unit is shown in Fig. 2.7-11. It is divided into two main sections, one for ATCRBS replies (top half) and one for Mode S replies (bottom). Only one control section is active at any given time, based on the ATCRBS/MODE S signal coming from the MCU. The two control sections will be discussed separately below.

2.7.4.1.1 AOA ATCRBS Controller

Eighteen control signals from the ATCRBS reply processor are used by the AOA ATCRBS controller. The actual leading edge stream and the pseudo-leading edge stream, appropriately delayed (DPLE and DPLE) are used to locate the position of a pulse. Note that so-called extra leading edges are not processed. After a leading edge declaration, the BD signal will indicate whether the leading edge was responsible for a bracket declaration. There are four BD signals, one from each reply decoder indicating which decoder was assigned to process the reply. These correspond to the SELECT signals, Fig. 2.5-13, after delay by the D flip-flop at entry to the decoder.

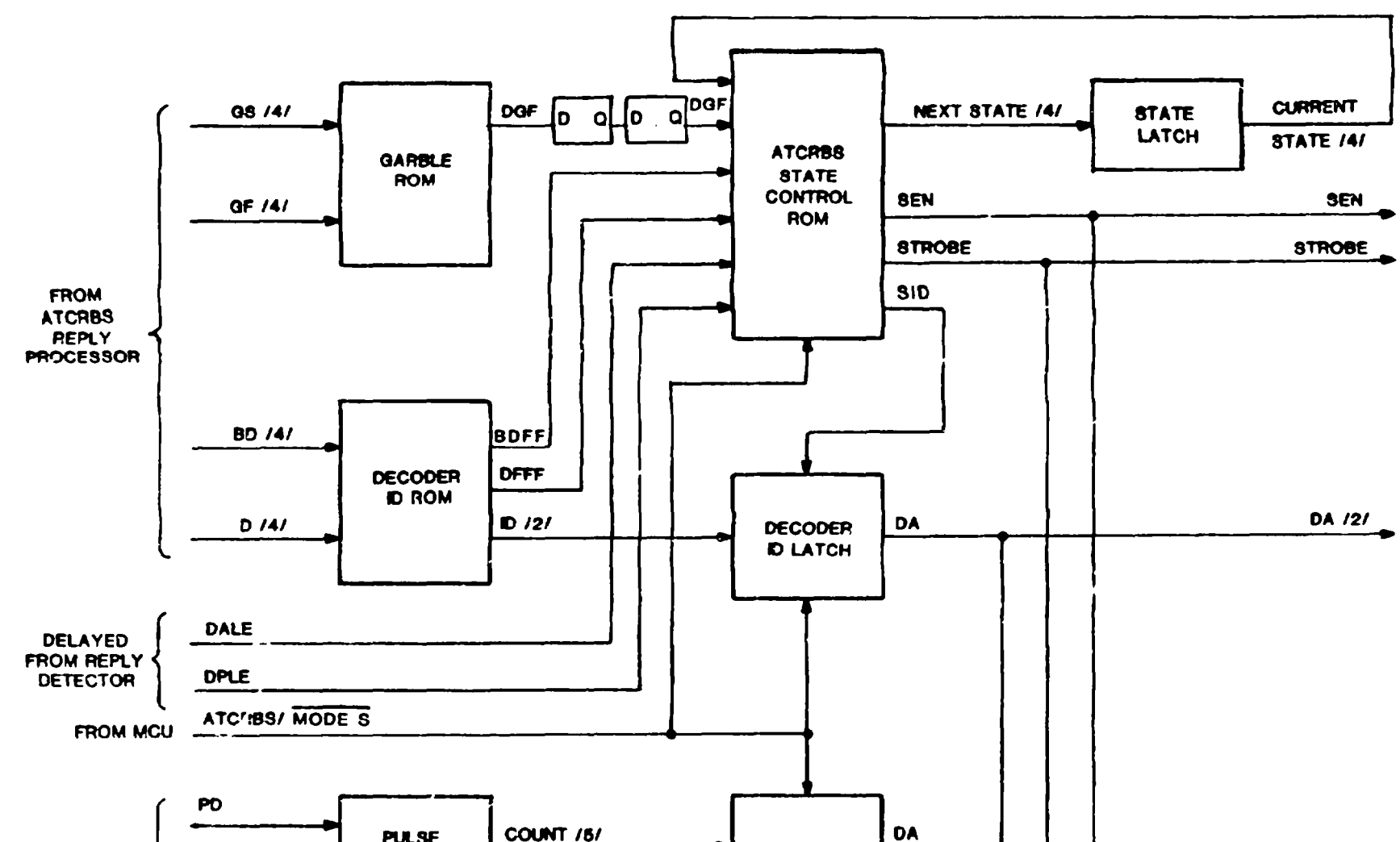


Fig. 2.7-11. TEU AOA processor-AOA controller.

Similarly, the four D lines indicate whether the leading edge was claimed by a reply decoder as a data bit for the reply it was processing. These correspond to the DATA STROBE signal, Fig. 2.5-13. The four BD lines and the four D lines are decoded by the decoder ID ROM to form a BDFF signal which is the logical "or" of the four BD lines, a DFFF line which is the logical "or" of the four D lines, and two ID lines which are encoded with a binary code to indicate which reply decoder had set either its BD line or its D line.

Garbling situations are declared by the ATCRBS reply processor via the four GF lines and the four GS lines (see Fig. 2.5-13). The GF lines are global garble flags for each reply decoder and indicate that the pulse position in question may be garbled by another reply. The GF signals are strobed by the GS signals which indicate when the GF signals are valid. The garble ROM performs the logical "or" over "and" function $\sum_{i=1}^{4} (GF_i \cdot GS_i)$ and DGF (delayed garble flag) indicates when the pulse being sampled may be garbled.

The timing of these signals is shown in Fig. 2.7-12. A DALE or DPLE pulse triggers the processing by the AOA ATCRBS controller. If the DALE or DPLE pulse caused a bracket detection, this will be indicated on the following clock tick by one of the BD lines. If the DALE or DPLE pulse is claimed by one or more reply decoders as a data pulse, the corresponding D lines will be pulsed during one of the three clock intervals. However, if more than one decoder claims the pulse the DGF signal will be high during one or more of the same three clock intervals. If both DFFF and BDFF are set simultaneously this indicates that the F1 pulse of a reply is garbled by another reply. The controller treats this as if the DGF line was set for that pulse.

The DALE, DPLE, BDFF, DFFF, and DGF signals are fed to a state controller which generates the control signals SEN, STROBE, and DA based on these inputs. The state diagram for the controller is shown in Fig. 2.7-13. Following the receipt of a leading edge, a four clock-tick cycle is entered to process the input information. If either a BDFF or a DFFF is declared during the first three cycles, without a DGF, the fourth cycle is used to generate the SEN signal to enable processing of that pulse. If the DGF signal is present, the controller times out the full four clock ticks, but no SEN is generated. Whenever BDFF or DFFF is present, the SID signal is used to save the reply decoder ID in the Decoder ID latch. The angle sample is taken on the second clock tick following a DALE pulse, or on the third clock tick following a DPLE pulse, by setting the STROBE signal. This causes the accumulator to latch the angle sample from the stream coming out of the digital delay. Under certain conditions, when processing wide ATCRBS pulses, the controller will abandon the DALE it is processing in favor of a DPLE. Figure 2.7-14 is a diagram of all wide-pulse situations (excluding cases with embedded ALE's), and indicates how these are sampled. Note that since not all edges are sampled, there can be situations where no SEN pulse results even though no garbling is present. The ATCRBS/MODE S signal is used to tri-state the ATCRBS SEN, STROBE, and DA lines during Mode S reply processing.

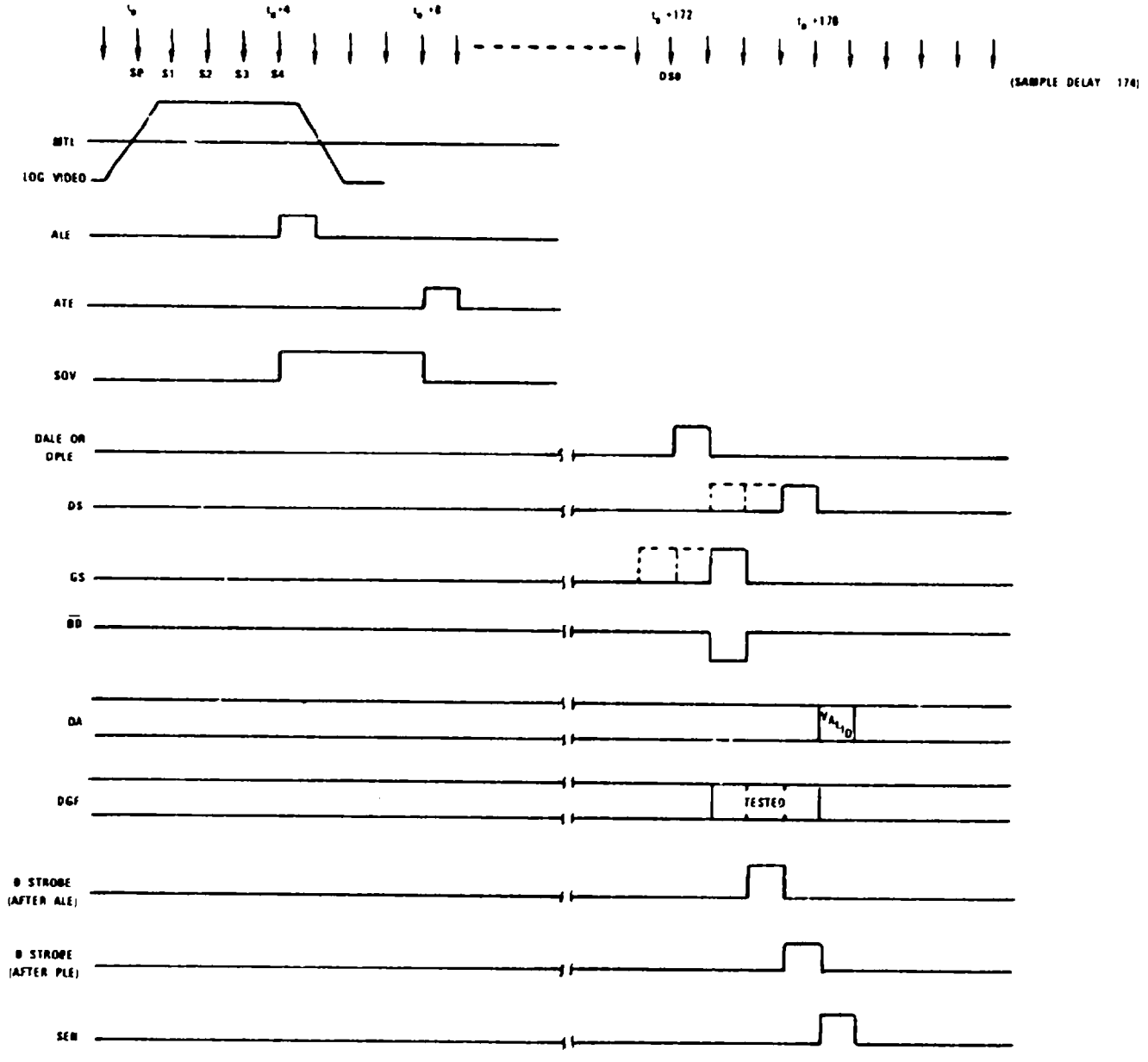


Fig. 2.7-12. TEU AOA ATCRBS timing diagram.

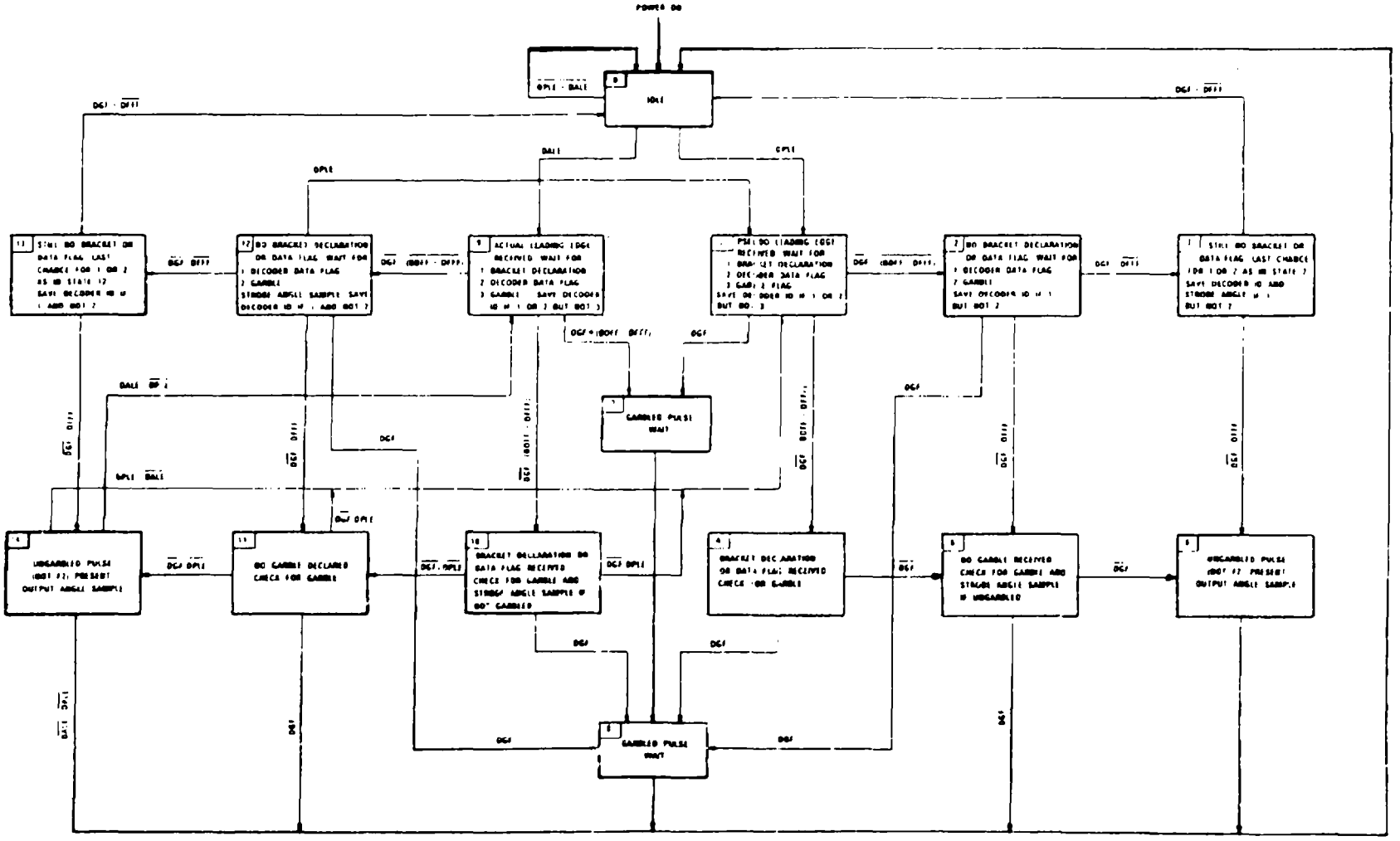


Fig. 2.7-13. TEU AOA ATCRBS State control diagram.

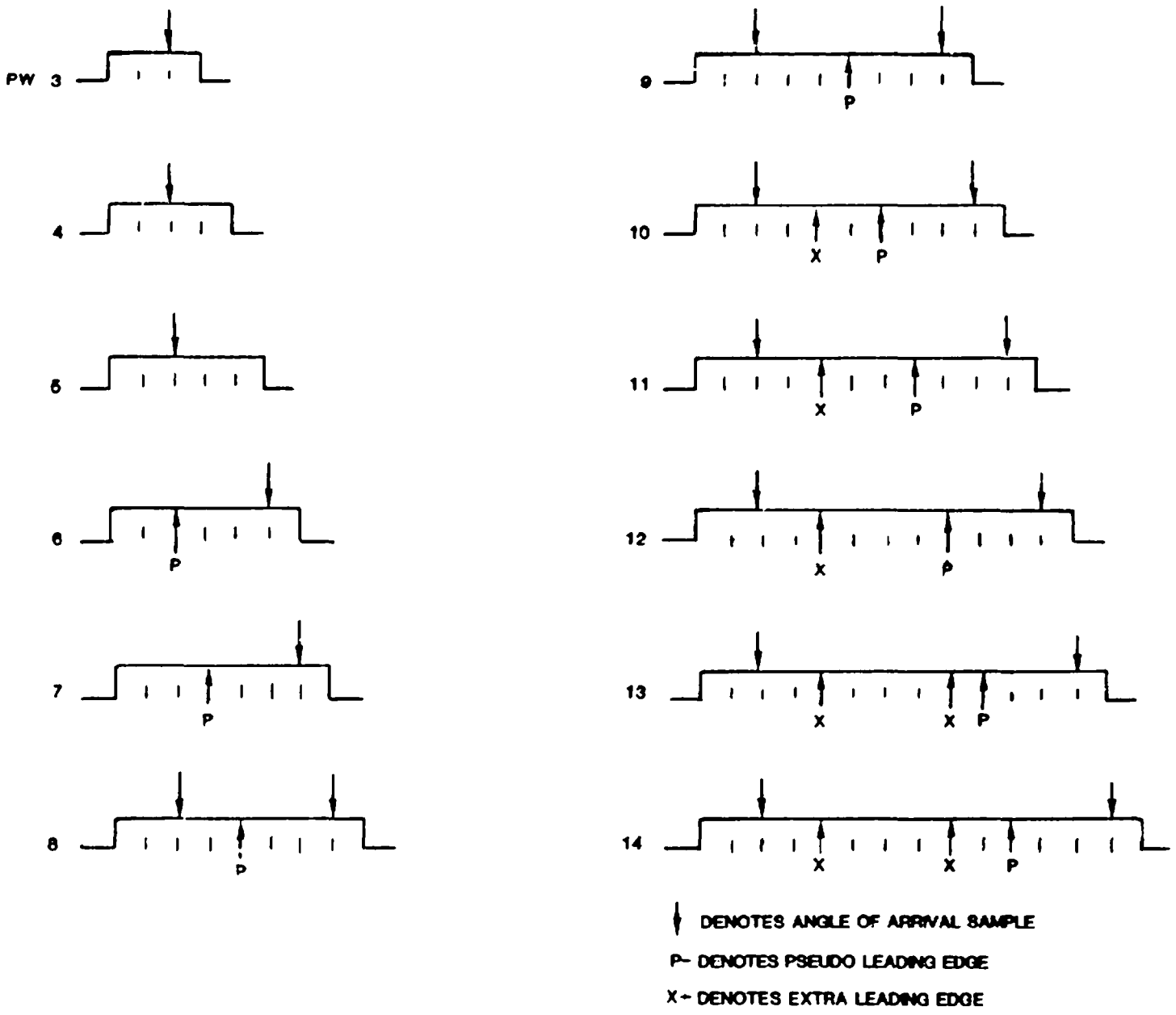


Fig. 2.7-14 AOA sampling of overlapped ATCRBS pulses

Note that the AOA processor as implemented assumes that valid ATCRBS pulses are at least 3 clock intervals long. This corresponds to the initial TEU design, but disagrees with the narrow pulse criterion presented in Section 2.5.3 which allows two-clock-wide pulses. TEU's with AOA retain the earlier design and reject two-clock-wide pulses.

2.7.4.1.2 AOA Mode S Controller

Enabled by ATCRBS/MODE S, the Mode S AOA controller provides the DA, STROBE, and SEN signals during Mode S reply processing. Since only one Mode S reply at a time may be processed, DA is arbitrarily chosen as the code "01". STROBE and SEN are provided by the Mode S control ROM based on five control signals from the Mode S Reply Processor in such a manner as to cause the accumulator to average the bearing samples from the first 32-bits of the message block. C1ST, the chip-1 strobe and C2ST, the chip-2 strobe are used to strobe the angle sample by setting STROBE. Either C1ST or C2ST is used, depending on whether the data pulse is in chip-1 or chip-2, as indicated by the MODE S DATA signal. The pulse counter is used to determine whether the pulse defined by C1ST/C2ST is one of the first 32 in the reply. The counter is cleared by the detection of a preamble (PD), and incremented by the confidence strobe (CONF STR) following each data bit.

2.7.4.2 AOA Accumulator

The AOA accumulator, shown in Fig. 2.7-15, is responsible for accumulating the sum of the bearing estimates for each reply being processed, as well as the number of pulses that comprise that sum, so that the multiplier can compute the average bearing estimate for each reply. This function is carried out by 4 bit-slice microprocessors, each four-bits wide, in a four clock-tick processing cycle. Eight registers of the micro processors are used to store $\Sigma\theta$ and N for each of four decoders. The four clock-tick processing cycle is divided into two halves by the accumulator control. During the first half (SEN,SEN1) the N counter for the reply decoder addressed by DA is incremented. If this is the first pulse processed for this decoder, N is set to 1. The N count is latched by the N-LATCH at the end of the first half-cycle, and its inverse, N^{-1}, is available at the output of the inverse lookup ROM. During the second half (SEN2,SEN3), the new bearing estimate, $\hat{\theta}$, is added to the old $\Sigma\theta$ for the reply decoder addressed by DA. If this is the first pulse, $\Sigma\theta$ is set to the new $\hat{\theta}$. The updated value of $\Sigma\theta$ is latched into the multiplier at the end of the second half-cycle, along with N^{-1}.

The Decoder Status Register (DSR) is used to keep track of whether this is the first pulse for a given reply decoder. The DSR bit corresponding to a given decoder is cleared by BD, in the ATCRBS mode, and by PD in Mode S. At the end of each accumulator processing cycle, determined by SEN3, that status bit corresponding to the decoder indicated by DA is set. Thus, a decoder status bit will be cleared at bracket detection, and set again when the first AOA sample from that decoder is processed. Note that the F1 pulse may not be

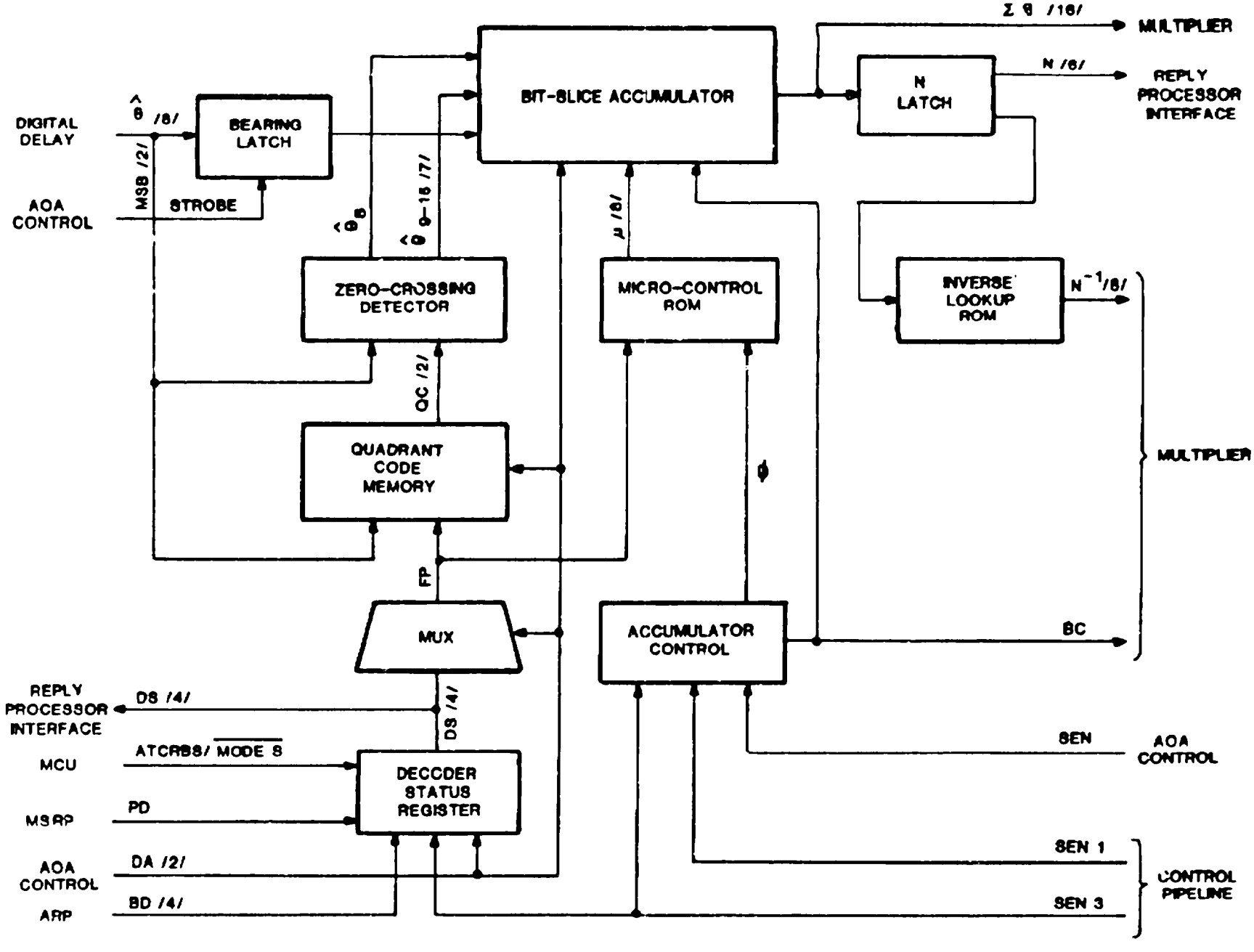


FIGURE 2.7-15 AOA ACCUMULATOR

the first AOA sample due to garbling or the presence of a following pseudo-leading edge. A multiplexor selects the DS line corresponding to the appropriate reply decoder for the current AOA sample. The output, FP, indicates whether the pulse being processed is the first for the selected reply decoder (if it is, the DS line will not yet have been set).

The clock phases (SEN, SEN1, etc.) from the control pipeline are used by the accumulator control in conjunction with the system clock to generate BC, the 2-cycle clock for the microprocessors and the multiplier (CK, DCK in Fig. 2.7-16), and ϕ, which indicates which half of the 2-cycle is active. ϕ and FP are used by the micro-control ROM to generate the proper micro-instructions for the microprocessors.

A problem arises when averaging angles between 0° and 360° in that for a reply near 0° when some of the pulses are at 0+ (0°, 5°, etc.) and some are at 0- (355°, 350°, etc.) a straight arithmetic average will not yield the correct result. In order to correct this problem, the accumulator divides the 0°-360° azimuth range into four 90° quadrants indicated by the two MSBs of the new angle sample ($\hat{\theta}$). It stores the quadrant code for the first pulse of each reply in the quandrant code memory. For each subsequent pulse, the zero-crossing detector compares the quadrant of the new angle sample with the quandrant code, QC, for the first sample of that reply and detects any zero crossing situation. If the first pulse was 0-, and a subsequent pulse is 0+, 360° is added to $\hat{\theta}$ by setting $\hat{\theta}_8$. If the first pulse was 0+ and a subsequent pulse is 0-, both $\hat{\theta}_8$ and $\hat{\theta}_{9-15}$ are set, changing $\hat{\theta}$ to $-\hat{\theta}$ to correct the zero crossing.

The timing diagram for the accumulator, multiplier, and reply processor interface is given in Fig. 2.7-16. Note that the four clock-tick cycle of the AOA accumulator depends on the AOA state controller's corresponding four clock-tick cycle to guarantee that the latched data inputs do not change before they are read.

2.7.4.3 AOA Reply Processor Interface

A block diagram of the AOA reply processor interface is shown in Fig. 2.7-17. The purpose of the reply processor interface is to save the updated average bearings from the accumulator/multiplier, and to buffer this data for transfer to the appropriate reply processor.

The first AOA Output Buffer (as shown in Fig. 2.7-17) stores the current values of $\bar{\theta}$ and N by the appropriately delayed reply decoder address (DA5) on the sixth phase of the SEN signal (SEN6). These values are updated every time a pulse is added to the estimate for $\bar{\theta}$. When the reply has been completely processed, the most current values of $\bar{\theta}$ and N are transferred to the second output buffer, along with the appropriate DS flag to indicate whether the data in the first buffer is valid (AEV in Fig. 2.7-17).

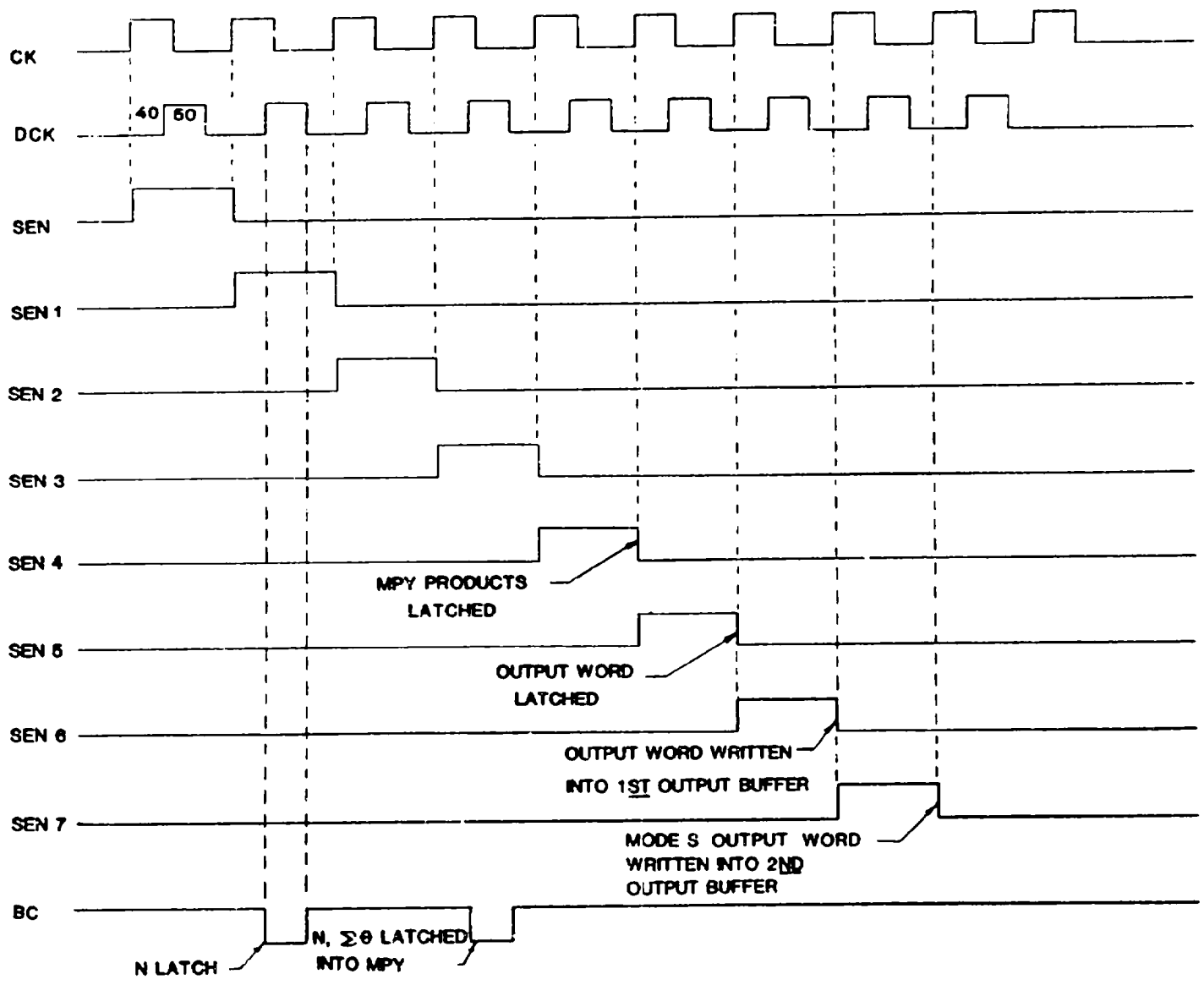


Fig. 2.7-16. TEU ACA accumulator timing diagram.

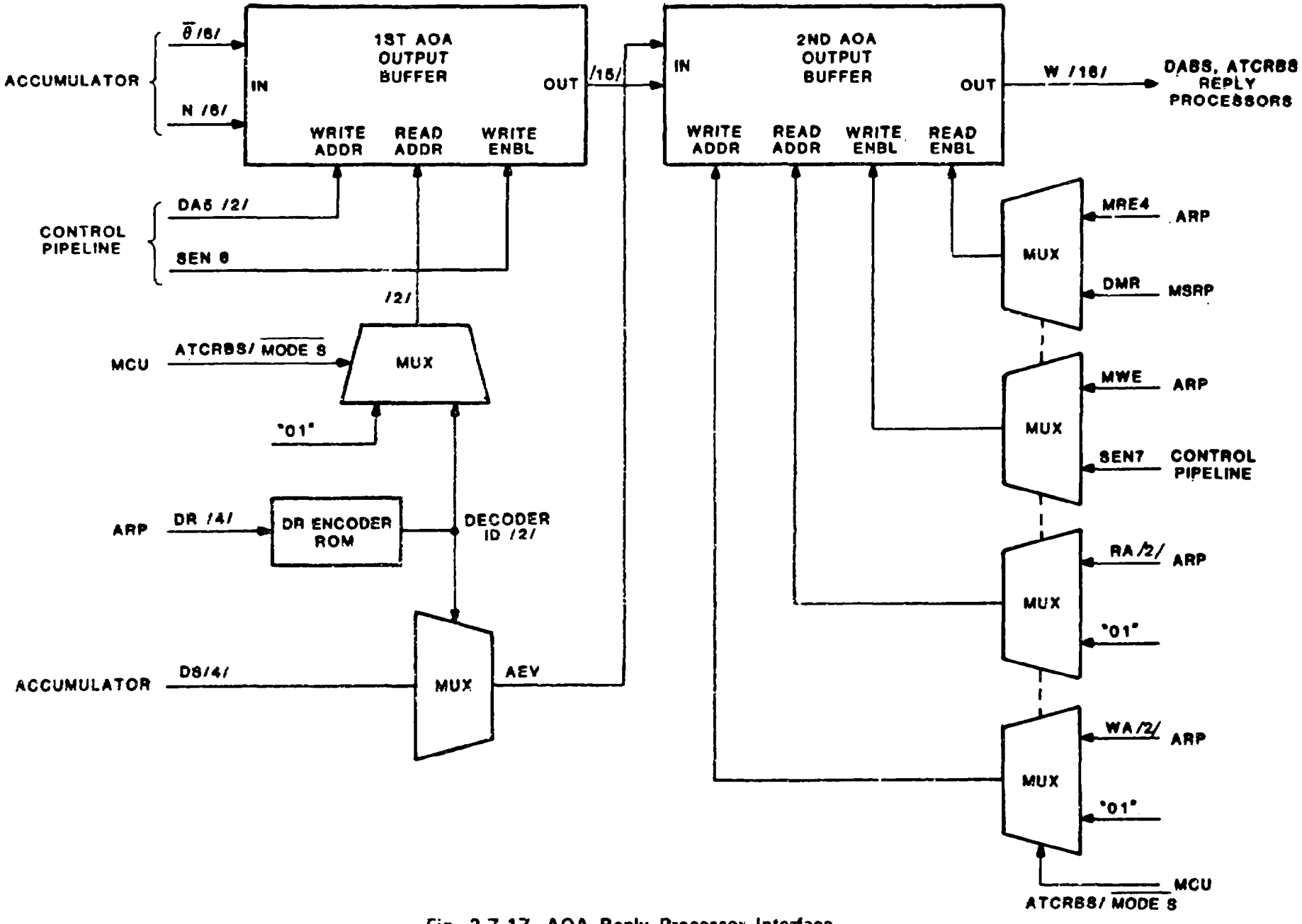


Fig. 2.7-17. AOA Reply Processor Interface

In Mode S the reply data ($\bar{\theta}$, N) is read out of location "01" in the first buffer (default DA value for Mode S), and written into location "01" of the second buffer during SEN7. It is held there until the Mode S reply processor requests the data by setting DMR, and the data is read out of location "01" and sent to the Mode S reply processor via the 16-bit W bus. This transfer occurs when the Mode S reply processor is transmitting its 9-word reply buffer to the ROLM CPU. The contents of the W bus are added to the transfer as the tenth word. The Mode S reply processor message transfer state controller is modified to provide the tenth data transfer. The revised state diagram for that controller is shown in Fig. 2.7-18.

In ATCRBS mode, when a reply decoder finishes processing a reply, the data ready (DR) flag is set. There is one for each reply decoder. The DR encoder ROM is used to encode these four lines to a 2-bit binary code, decoder ID. This signal is used to address the first output buffer so that the angle data for the correct decoder is read out and into the second buffer. Decoder ID is also used to generate AEV from the appropriate DS flag. The second output buffer is an extension of the 3-word reply buffer in the ATCRBS Reply Processor. Data from the first buffer is written into, and read out of, exactly the same locations as in the 3-word buffer in the ATCRBS Reply Processor, these locations being contained in WA and RA respectively. Data is written into the second buffer at exactly the same time as data is being placed in the ARP Buffer, strobed by the MWE signal. Data is read out of the buffer as requested by the ATCRBS Reply Processor with the MRE4 line, and the contents of the W bus is appended to the normal 3-word data transfer as a fourth word. The ATCRBS reply processor output state controller was modified to provide the fourth data transfer. The revised state diagram for that controller is shown in Fig. 2.7-19.

The format of the AOA word is the same for both ATCRBS and Mode S and is given in Fig. 2.7-20. In both cases this word is appended to the reply formats used in non-AOA TEU's.

2.8 Mode S Transponder Interface

2.8.1 Purpose

TCAS units, as defined by Ref. 6, perform air-to-air coordination among units involved in a collision-avoidance situation. This assures that aircraft will make complementary maneuvers. For example, in an encounter between two TCAS units, the coordination will assure that one pilot is given a "climb" or "don't descend" command, while the other pilot is given a "descend" or a "don't climb". Mode S ground sensors may also communicate with TCAS units in order to set their sensitivity level (a parameter which controls the range/range rate criteria used to detect threatening situations, and which may be used to inhibit TCAS commands or to disable the TCAS unit in certain regions of airspace) and to read out the coordination information developed during an encounter. The link between the TCAS unit and its associated Mode S transponder is used in both the air-air and air-ground protocols.

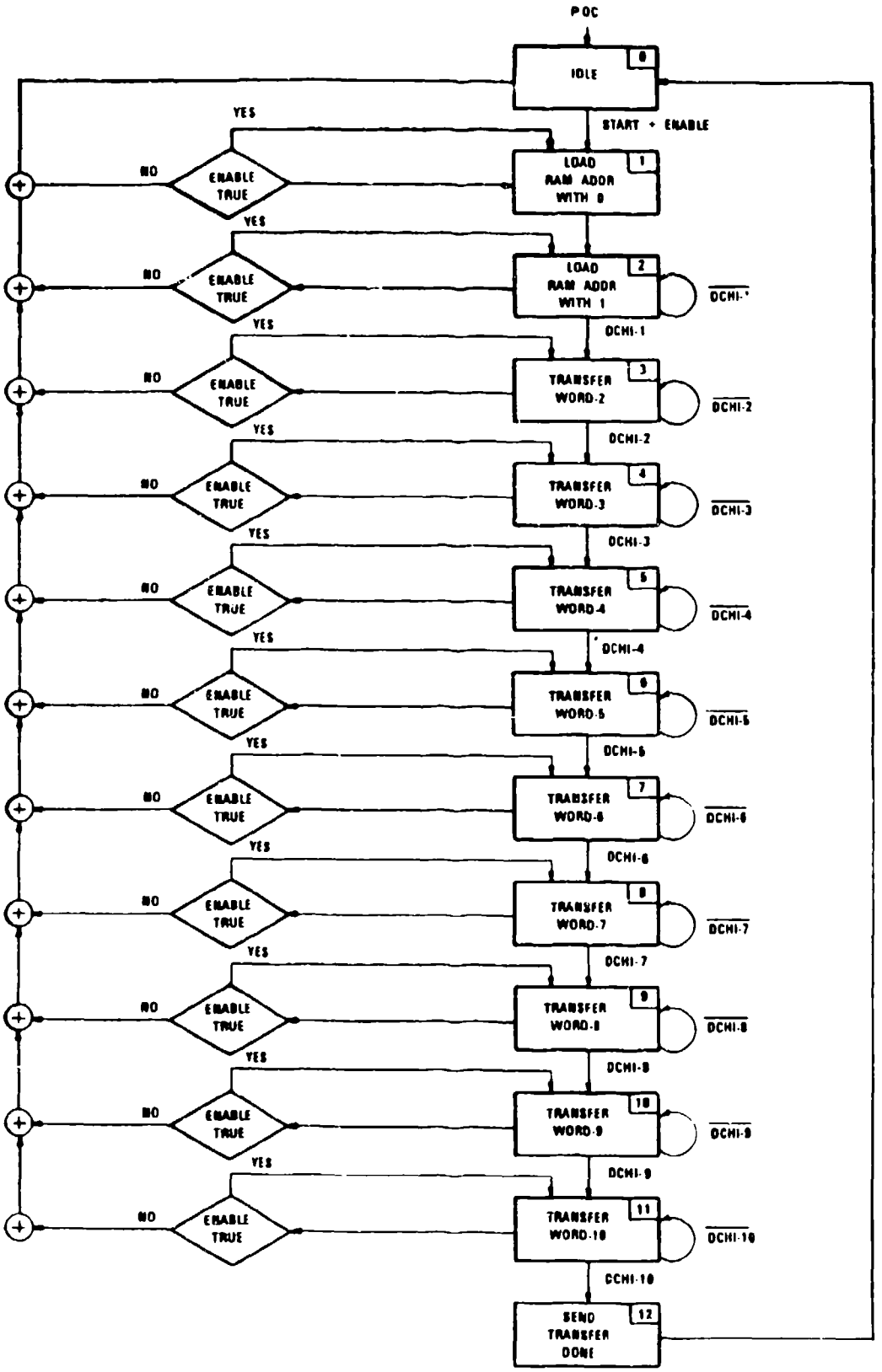


Fig. 2.7-18. Modified Mode S I/O controller state diagram.

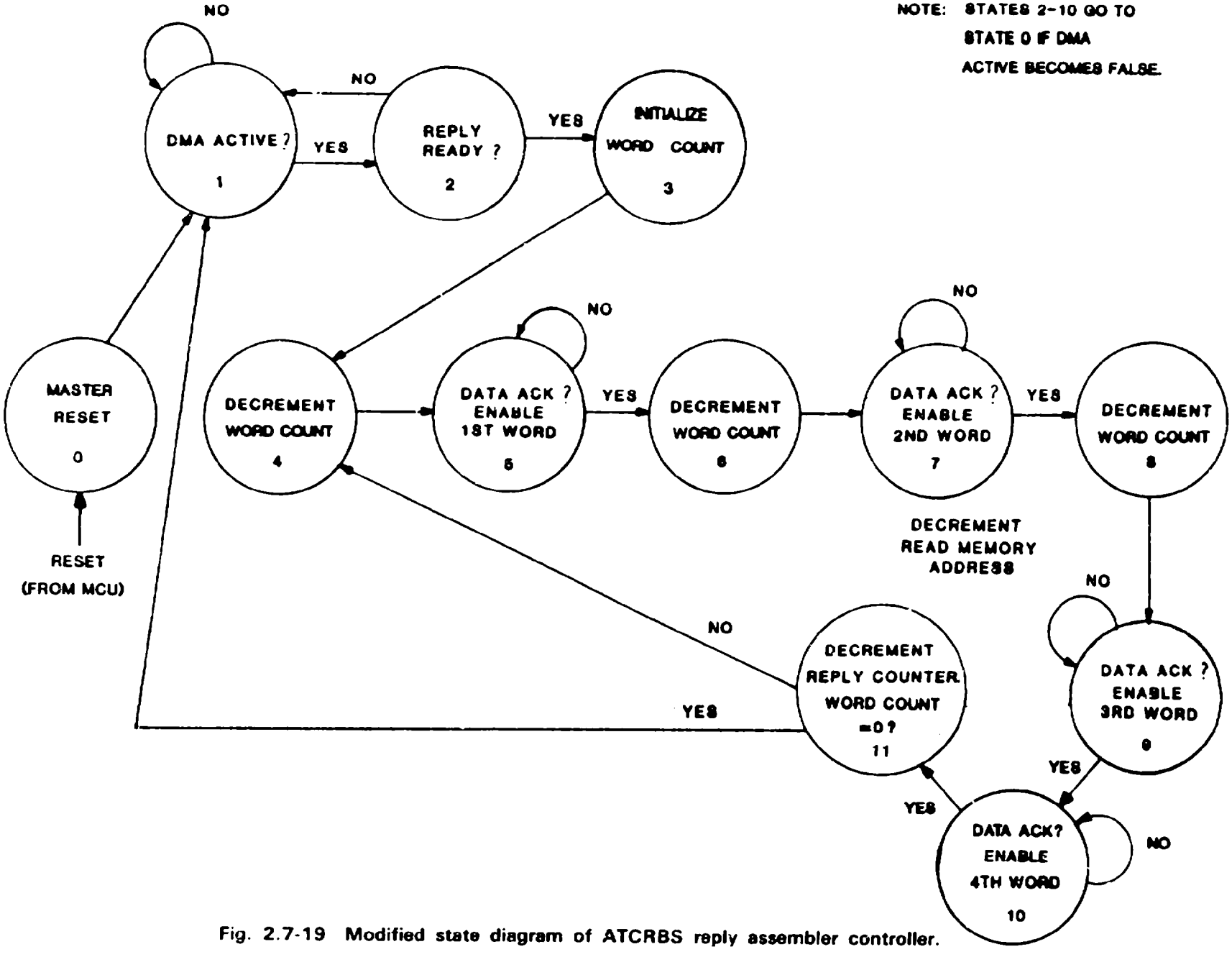
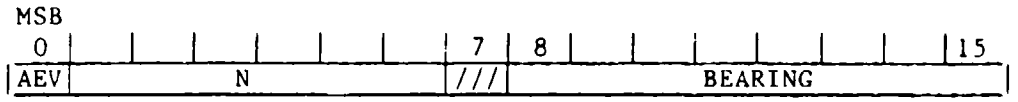


Fig. 2.7-19 Modified state diagram of ATCRBS reply assembler controller.



AEV	Angle Estimate Valid (1=valid)
N	Number of bearing samples (pulses) averaged to obtain the bearing field. This is always 32 for Mode S replies. For ATCRBS replies it normally corresponds to the number of ungarbled "1" bits in the data field plus one for the F1 pulse sample (F2 is not sampled). It may be less than this due to the F1 garble situations or to wide pulse situations where a DPLE pulse causes DALE-triggered sampling to be abandoned.
Bearing	Bearing of threat, LSB 360°/256. The bearing as reported by the AOA hardware needs to be corrected by the software before being used as the desired clockwise bearing relative to the aircraft's nose (i.e., right wing = 90°, tail = 180°). The first correction is for antenna orientation. A bottom mounted antenna with the same connections as the top antenna will give increasing bearings for counter clockwise movement of the target, and so the bearing from this antenna must be negated (2's complement). The second correction is for any bias between zero bearing and the nose of the aircraft. The third correction is for any non-linearity in the bearing measurement due to antenna or receiver effects. This may be done by table lookup, but such a table in general must be a function of both bearing and elevation angle. The procedure followed in the TEU's, given the $\pm$15° accuracy requirement and that the purpose of the bearing measurement was to allow pilots to visually locate threatening aircraft, was to adjust the overall bias to roughly minimize the bearing error over the forward hemisphere and within $\pm$10° elevation angle. The top and bottom antennas require separate corrections.

Fig. 2.7-20. AOA word format for ATCRBS and Mode S replies.

2.8.2 TEU Implementation

A Mode S transponder interface was implemented for the TEU's and used in experiments. However, the protocols implemented corresponded to earlier versions of the TCAS design and transponder data formats, and are not relevant either to current use of the TEU's or to the design of commercial TCAS systems. Therefore, they will not be discussed in detail. The TEU digital card slot originally used for the Mode S transponder interface has since been taken up by the AOA processor in the two TEU's with AOA capability.

2.8.3 General Principles

When the Mode S transponder receives an interrogation with the appropriate Mode S address, it transmits the entire message content (except for the address) over the data link interface (see Fig. 2.1-1) to all devices on board the aircraft which use the data link capability of the transponder.

The Mode S transponder interface must process all such messages and take action on any whose message type field indicates it is of interest to TCAS. Processing may include, depending on the message type, the receipt and buffering of the message, interrupting the TCAS software to inform it of the arrival of a message, modifying air-to-air coordination data maintained by the transponder interface board, and generating data bits to be sent over the data link interface for inclusion in the transponder's reply to that interrogation. Coordination data must be maintained by the transponder interface due to the required timing. The data in the interrogation and the current state of the coordination data is used to determine the content of the reply, which the transponder starts sending 128 sec after the arrival of the leading edge of the interrogation.

Since the TCAS software must also access and modify this coordination data, the Mode S transponder interface must provide these functions to the software and also locking primitives sufficient to lock the coordination data base according to the prescribed locking protocols [Ref. 6]. The locking functions are necessary to prevent the software in the TCAS unit from modifying the coordination data simultaneously with modifications caused by interrogations from other TCAS units. Also, in case of multiple intruder encounters, the locking functions prevent messages from the other TCAS units from interfering with each other.

2.9 General Purpose I/O Interface

2.9.1 Function

The General Purpose I/O (GPIO) interface provides the control logic necessary to interface the following TEU devices to the ROLM 1650 processor:

Direct Memory Access Devices
- Modulation Control Unit (MCU)
- ATCRBS Reply Processor
- Mode S Reply Processor
- Mode S Transponder Interface

Programmed I/O Devices
- Pressure Altimeter
- IVSI Display
- Sensitivity Level Selector
- Radar Altimeter

The general purpose I/O interface provides for both block data transfers (direct memory access) and single data word transfers (programmed I/O) between TEU system devices and the ROLM 1650 processor. In the block data transfer mode transfers are sequential to or from a block of memory defined by a starting address and a word count (block length). The programmed I/O logic section of the interface transfers a single 16-bit word at a time either from an external device to one of the processor's registers or from a register to an external device.

Normally in a system of this sort each device using the direct memory access mode of data transfer would interact directly with the ROLM I/O bus rather than through an intermediary such as the GPIO interface. However, this requires a significant amount of circuitry on each device. The GPIO interface consolidates all the data channel circuitry for the devices, and in turn presents these devices with a much simpler data channel interface. The disadvantage is that only one of these devices has access to the data channel at any time, and parallel independent operation is not possible. In this application that is not a severe restriction.

2.9.2 Summary of I/O Interface Conventions

The TEU employs a ROLM 1650 computer and associated I/O devices and interfaces. ROLM interfaces have two modes of data transfer. One is programmed I/O, where the transfer of each word is controlled by the software. The other is data channel I/O, where the software merely specifies to the device the address and length of a buffer area in memory (via programmed I/O). All transfers to or from that buffer are controlled by the device interface, independently of the software.

Programmed I/O is conducted between a CPU register and one of three "registers" on the device interface, the A, B, and C registers. These may or may not be implemented as unique hardware registers. It is better to regard A, B, and C as device addresses of particular components of the interface, rather than as actual registers. Output to these registers occurs via the DOA, DOB, and DOC instructions, respectively. The corresponding input instructions are DIA, DIB, DIC. Each programmed I/O instruction has as arguments the CPU register to/from which data is to be transferred, and the device address. The A, B, and C registers read on input may or may not be the same as those written to by the output instructions.

As mentioned above, data channel I/O is initialized by means of programmed I/O. Normally a DOA instruction specifies the starting address of the buffer to be read from or written to. A DOB instruction is used to transfer the length of the buffer. By convention the value actually passed is the two's complement of the word count. The block transfer is initiated by a START pulse, described below, appended to the DOB instruction.

Each device interface has three status flip-flops which are controlled by the software, the BUSY, DONE, and INTERRUPT DISABLE flip-flops. If BUSY is set, the device is active. DONE is always zero in that circumstance. When the device completes its operation, it clears BUSY and sets DONE. This automatically causes an interrupt to be generated at the CPU, unless the INTERRUPT DISABLE flip-flop is set, in which case the interrupt will be inhibited until INTERRUPT DISABLE is cleared. If DONE is set when INTERRUPT DISABLE is zero, an interrupt will be generated immediately. A device is completely inactive when both BUSY and DONE are zero.

The software can test the status of BUSY and DONE by means of the I/O SKIP instructions. For example, SKPBN will skip the next instruction if BUSY is Non-zero for the specified device code.

There are several signals available to control these status flip-flops. The START line, when pulsed, will set BUSY and clear DONE. This usually has the effect of activating the device. The CLEAR line, when pulsed, will set both of these to zero. The IORST line is similar to CLEAR, except that the INTERRUPT DISABLE flip-flop is also cleared.

To pulse the START and CLEAR lines the letters S and C respectively are appended to the programmed I/O instruction mnemonics (e.g., DOAS, DIRC), or to a NO I/O TRANSFER instruction (NIOS, NIOC). There is a third control line, called the P line, which is operated similarly by appending a P to the I/O commands. It has no standard usage, and is interpreted differently by each device.

The IORST line is pulsed by the I/O RESET (IORST) instruction, or by toggling the RESET switch on the front panel. It is pulsed automatically by the CPU when power is first turned on. Note that the input and output instructions, and any START, CLEAR, or P pulse associated with them, affect only the addressed device, while an IORST affects all devices.

INTERRUPT DISABLE is normally set and cleared by a MASK OUT (MSKO) instruction. This instruction places the contents of a specified CPU register on the data lines, and then sends a special control pulse. Each device looks at only one of the 16 data bits, and loads the value of that bit into its INTERRUPT DISABLE flip-flop on receipt of the pulse. A bit value of 1 disables the interrupt.

The operation of the interrupt request logic and the data channel logic are controlled by separate priority chains. The priority signals originate at the CPU and are part of the ROLM I/O bus. When a device requests an interrupt or requests a word transfer over the data channel it disables the appropriate

priority signal for all devices farther from the CPU. When the CPU services such a request, therefore, it always services the requesting device nearest the CPU on the appropriate priority chain. It is not necessary that devices occupy the same positions on both the data channel and interrupt priority chains.

When an interrupt is requested, the CPU stops its normal processing at the next interruptable point (usually between instructions). Various modes of state saving are possible in the ROLM CPU. The TEU software uses a mode in which the interrupt return address and the current device interrupt mask are automatically pushed on the stack, a new interrupt mask is output, and the address of the interrupting device is determined by the CPU. The CPU then uses that device address to index an interrupt vectoring table and branch to the appropriate handler for that device. On releasing the interrupt the old mask is restored from the stack and the CPU returns to the interrupted program.

To determine the address of the interrupting device the CPU emulates an Interrupt Acknowledge (INTA) instruction. This pulses a control line on the I/O bus, and this in turn causes the highest priority device requesting an interrupt to place its device address on the data lines.

When a data channel transfer is desired, a device goes through a synchronization procedure with the CPU using several control signals. At the completion of this sequence, one device has been selected as the highest priority device requesting a transfer. The selected device indicates the desired transfer direction (in or out of memory). The CPU then requests the memory address of the word to be read or written, and finally either reads the word and places it onto the data lines of the I/O bus or stores the data on the bus into the memory word. Normally, only one word is transferred for each synchronization cycle. However, it is possible for a device to request a burst mode transfer where an arbitrary number of words can be transferred once access to the data channel has been obtained. This gives the maximum transfer rate, but has the effect of stopping all instruction processing by the CPU until the transfer is completed.

2.9.3 Programming the General Purpose I/O Interface

Device code 6 is used to control the programmed I/O devices and device code 7 controls the data channel I/O device. Mask bit 8 (0 is the MSB, 15 the LSB) controls the INTERRUPT DISABLE flip-flop for device 7. Device 6 never interrupts. The available I/O operations and their word formats are summarized in Tables 2.9-1 and 2.9-2.

To initiate a data channel transfer, one of the data channel devices controlled by the GPIO interface must first be selected. This is done via a DOC instruction. Then, the buffer starting address is initialized via a DOA instruction, and the word count (two's complement) by means of a DOBS instruction. The START pulse associated with the DOB instruction initiates the transfer. Note that the direction of the transfer is determined by the device selected, specifically by the least significant bit of the GPIO device code.

TABLE 2.9-1

GPIO PROGRAMMED I/O SUMMARY

Device: GPIO Programmed I/O
Device Number: 6
Mask Bit: None (no interrupts generated)

START CLEAR P IORST - No effect. There is no BUSY/DONE, interrupt, or data channel logic.

DOA AC,6: Loads the IVSI display register from accumulator AC. The word format is as follows:

0						7	8							15
SP		F	-2	-1	-5	DD	DC	+5	+1	+2	NT	D	L	C

SP: Spare bits
F: Flag bits. These control one or two mechanical flags on the face of the IVSI. Different IVSI's have different flag arrangements.
-2: Lights the yellow segment from -2000 fpm to the end of the IVSI range.
-1: Lights the yellow segment from -1000 fpm to -2000 fpm.
-5: Lights the yellow segment from -500 fpm to -1000 fpm.
DD: Lights the yellow segment from -200 fpm to -500 fpm. (Used by the software for "Don't Descend")
DC: Lights the yellow segment from +200 fpm to +500 fpm. (Used by the software for "Don't Climb")
+5: Lights the yellow segment from +500 fpm to +1000 fpm.
+1: Lights the yellow segment from +1000 fpm to +2000 fpm.
+2: Lights the yellow segment from +2000 fpm to the end of the IVSI range.
NT: Lights the two red NO TURN lights that appear in the lower left and lower right corners of the IVSI, outside the main circular display area.
D: Lights the red downward pointing DESCEND arrow just below the center of the IVSI.
L: Lights the level flight indicator that lies underneath the IVSI needle in its zero velocity (center) position.
C: Lights the red upward pointing CLIMB arrow above the center of the IVSI.

DIA AC,6: This loads data from own aircraft's altimeter, the sensitivity level selector switch, and the radar altimeter into accumulator AC. The word format is as follows:

0							7	8							15
R1	R2	SL			R3	D4	A1	A2	A4	B1	B2	B4	C1	C2	C4
						. . . Altitude									

TABLE 2.9-1

GPIO PROGRAMMED I/O SUMMARY (CONT'D)

R1,R2,R3: Radar altimeter data. Each line represents an altitude threshold, measured with respect to ground level. By convention, R1 repesents the lowest threshold and R3 the highest threshold. A 1 indicates that own aircraft is above the corresponding threshold altitude. The threshold levels are set on the radar altimeter, and are not under the control of the TEU. Not all aircraft are equipped with radar altimeters, in which case these lines default to 1. For some experiments these lines may be connected to other signals to record switch closings, etc.

SL: Sensitivity level switch value. Values 0 through 4 represent the five possible switch settings.

ALTITUDE: The pressure altitude of own aircraft. Measured relative to sea level, but without barometric correction. The standard altitude Grey code is used, with the bit positions assigned as shown. Note that the D2 bit is not read, as this is set only above 62750 feet. This is identical to the altitude code that is sent by the Mode S transponder, and this fact is critical to safe system operation.

TABLE 2.9-2

GPIO DATA CHANNEL SUMMARY

Device: GPIO Data Channel
Device Number: 7
Mask Bit: 8

START: Sets BUSY, clears DONE, starts the data channel transfer.
CLEAR: Clears BUSY, DONE, and both data channel and range window closed interrupts. Does not terminate pending data channel transfers, but clearing of BUSY will stop devices from initiating further transfers.
P: Clears range window closed interrupts.
IORST: Performs all the functions of CLEAR, and in addition clears any pending data channel activity.

DOA AC,7: Loads the GPIO address register from accumulator AC. This should contain the 16-bit address of the first word of the buffer.

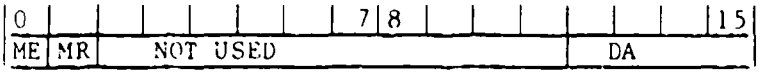
DOB AC,7: Loads the GPIO word count register from accumulator AC. This should contain the two's complement of the buffer length in words.

DOC AC,7: Loads the GPIO device address register from accumlator AC. This determines both the device which is to access the channel and the direction of transfer. The device address is contained in bits 12-15 of the accumulator (low order 4 bits). The remainder of the accumulator is ignored. Bit 15 determines the transfer direction, 0 indicating output from memory to the device. The device codes, in binary, are as follows:

Modulation Control Unit	0000
ATCRBS Reply Processor	0001
Mode S Reply Processor	0011
Mode S Transponder Interface, Output	0010
Mode S Transponder Interface, Input	0101

DIA AC,7: Reads the current value of the GPIO address register into accumulator AC. This represents the next address in the buffer to be read or written by the data channel. This value will increment after each word is transferred by the data channel.

DIB AC,7: Reads the GPIO status word into accumulator AC. Only the MCU device reports status. The format is as follows:



ME: Modulation Enabled from the MCU.
1 = Modulation is allowed.

TABLE 2.9-2

GPIO DATA CHANNEL SUMMARY (CONT'D)

0 = Modulation is not allowed. If the software attempts to generate an interrogation by sending data to the MCU, the MCU will accept the data normally, but no modulation waveform will be sent to the transmitter. The range counter will be cleared. The minimum allowed interval following interrogations is 2 msec following an ATCRBS interrogation and 16 msec following a Mode S interrogation. These are determined by the allowable transmitter duty cycle.

MR: MCU Ready

1 = The MCU is ready to accept a data channel transfer.

0 = The MCU is not ready to accept a data channel transfer, as it is still busy with the previous transfer.

DA: Current GPIO device address, coded as for the DOC instruction.

As data is transferred by the device, the GPIO address register increments. It always indicates the next address in memory to be read or written. To determine how much data has been transferred the software can read this address by means of a DIA instruction.

A DIB instruction is used to read device status. Currently only the MCU reports status data. In addition to the device status, this instruction reads the device select register on the GPIO interface, so that it is possible to determine which device was last selected.

Programmed I/O is done via the DOA and DIA instructions for device 6. The DOA instruction controls the lights on the IVSI display. Each lighted segment of the display is individually controlled. In addition to the lighted segments around the rim of the vertical speed indicator, there are up and down arrows used to indicate climb and dive commands. There are also controls which light the LEVEL light, the NO TURN lights, and which control the mechanical status flags on the IVSI.

The DIA instruction reads own aircraft altitude, the setting of the sensitivity selection switch, and the radar altimeter inputs. The pressure altitude is coded in the usual altimetry Grey code. It is identical to the altitude reported to the ground ATC system by the transponder. The radar data represents the status of the aircraft with respect to three altitude thresholds. These thresholds are manually settable on the radar altimeter. More abstractly, each bit represents the electrical state of a particular wire, and so these wire's can also be used to read switch settings or other such data. They have been used in flight testing to allow observers to mark significant events on the data tape by closing a switch.

The GPIO interface generates an interrupt from device 7 under two circumstances. The first is when a data channel transfer is in progress and the word count goes to zero. The second is when the MCU indicates that the range window has closed (i.e., the specified period for listening for replies to an interrogation has ended). The software can test which of these two conditions caused the interrupt by testing the GPIO interface DONE flip-flop using the I/O skip instructions. If DONE is set, this indicates a buffer overflow condition. The range window closed condition uses a second DONE flip-flop which cannot be directly tested by the I/O skip instructions. The buffer overflow condition can be cleared by the usual means, either a START or CLEAR pulse. The "range window closed" interrupt may be cleared either by CLEAR or by means of a P pulse (e.g., NIOP). IORST will clear either condition.

2.9.4 Operation

Figure 2.9-1 is a block diagram of the GPIO interface. The GPIO Control performs all the standard protocols with the ROLM I/O bus. At appropriate points in each protocol it generates control signals for the various registers and gates, as well as the necessary data channel control signals for the data channel devices. The control signals related to programmed I/O are labeled with the I/O instruction which triggers them. For example, the signal which

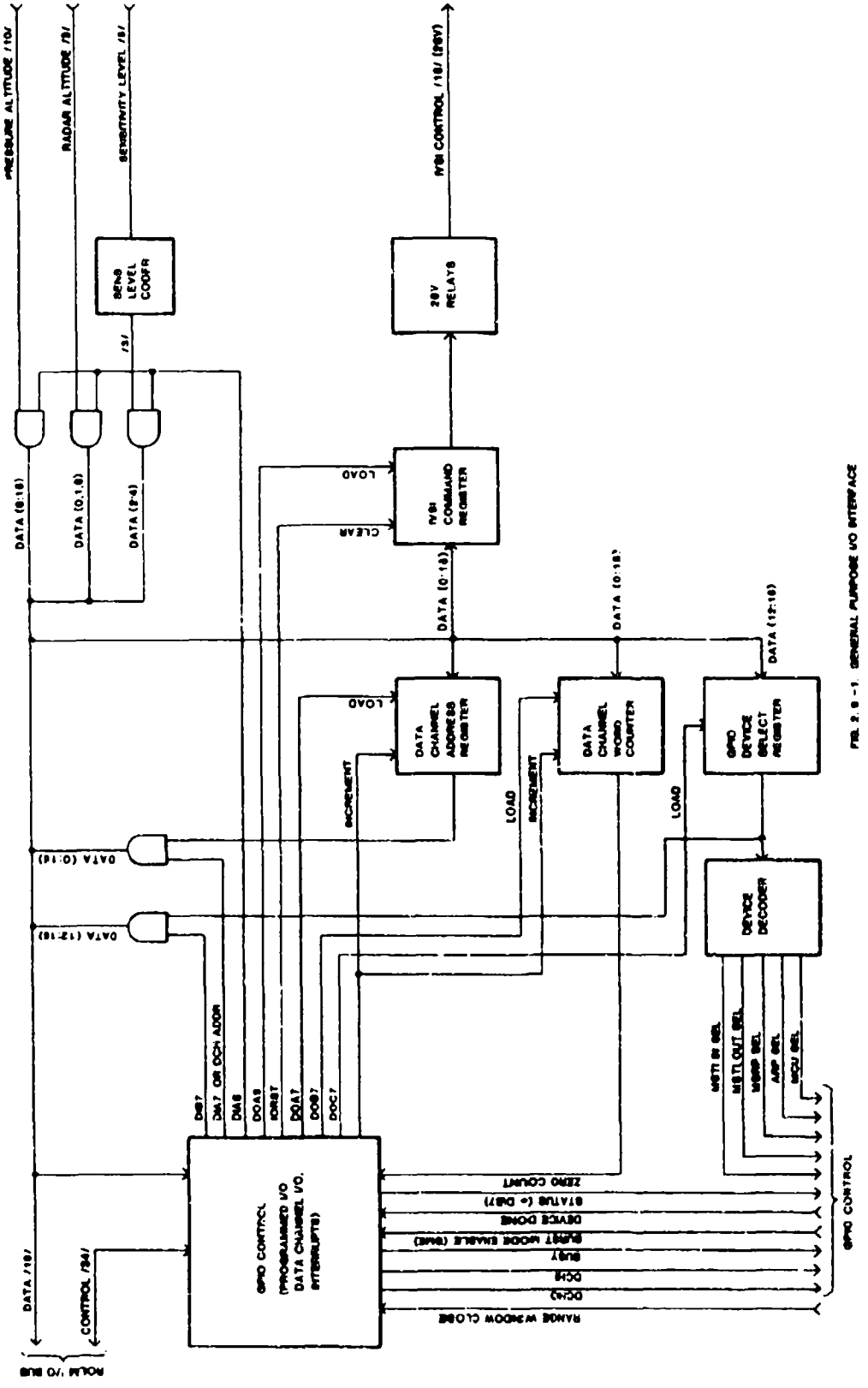


FIG. 2.8-1 GENERAL PURPOSE I/O INTERFACE

causes the data channel address register to be loaded is "DOA7", and is generated during a DOA operation to device 7. It causes the address register to be parallel loaded from the 16 data lines of the ROLM I/O bus. The other programmed I/O signals shown operate similarly.

When a GPIO data channel operation is initiated, a DOC7 instruction is used to load the GPIO Device Select Register. This is decoded to enable one of the five device select lines. The data channel address register and word count are next initialized via DOA and DOB commands, respectively, to device 7. A START pulse is then sent to the GPIO interfaces by the software. This sets the BUSY flip-flop in the GPIO Control, and this is reflected in the BUSY control line going to the GPIO data channel devices. The selected device uses this to trigger the beginning of its data channel operations. When it wishes to transfer a word, it indicates this via the DEVICE DONE line. The GPIO control then starts the data channel protocol with the ROLM CPU. When the CPU requests the memory address, the data channel address register is gated to the data lines. The GPIO Control then increments both the address and word count register. The CPU next initiates the data transfer by a pulse on either the DCHI or DCHO lines on the ROLM bus. Which line is pulsed depends on whether an input or output operation was requested by the GPIO Control, and this in turn depends on the least significant bit of the GPIO device address that was selected. The GPIO control parrots the DCHI and DCHO signals to the data channel devices, and they are responsible for either gating their data to the data bus for input operations, or for loading the data on the bus to a register for output operations.

Devices which have a requirement for rapid transfer of consecutive words of data can signal this via the BURST MODE ENABLE line. As long as this line is held, the GPIO Control will hold the ROLM data channel and perform consecutive transfers at the maximum possible rates (660,000 words/sec input, 525,000 words/sec output). During this time the CPU will be locked out of memory and will therefore not be able to process any instructions.

A status request (DIB to device 7) causes the STATUS line to be pulsed. A device can place any status information onto bits 0-11 of the data bus at that point (bits 12-15 are set to the device code of the currently selected GPIO device). Currently, only the MCU uses this feature.

2.10 Real-Time Clock

2.10.1 General Description

The TEU real-time clock generates an independent source of interval timing pulses used to initiate program interrupts at regularly recurring intervals. These intervals are selected under program control and may be 100 milliseconds (10 Hz), 10 milliseconds (100 Hz), or 1 millisecond (1000 Hz). The real-time-clock may be activated and deactivated under program control.

A crystal oscillator is used to generate a basic clock signal and is followed by a digital frequency divider which generates the three interrupt intervals.

2.10.2 Programming

The real-time-clock is a programmed I/O device using conventional Busy, Done, Interrupt Request, Interrupt Disable, and Device Select logic (see 2.9.2). The ROLM 1650 processor communicates with the real-time-clock using the DOA and NIO instructions. The real-time-clock device code is 14_8 and the interrupt disable mask bit is 13.

The DOA AC, 14 instruction is used to specify the desired interrupt frequency. Bits 14 and 15 of the accumulator selected by the DOA instruction contain the 2-bit code which selects the desired interrupt request interval.

Data 14	Data 15	Freq.
0	0	0
0	1	10 Hz
1	0	100 Hz
1	1	1000 Hz

The function codes for Start and Clear operate in the usual manner, as do the skip instructions sensing the interface Busy and Done conditions.

After the Start function causes the real-time-clock to enter the Busy state, the next clock pulse at the selected rate advances Busy to Done and initiates an interrupt request if interrupts are enabled. The interrupt request interval (DOA AC, 14) instruction need only be given once. Thereafter, each successive interrupt can be serviced using an NIOS instruction. The first time the program issues a Start function, the elapsed time before the first interrupt may be any amount up to the full period of the interrupt interval selected. Subsequent interrupts occur at the selected interval as long as the program issues an NIOS 14 before the next interval expires. IORST resets the interrupt interval rate to 0 as well as performing the usual reset function.

The real-time-clock logic resides on the same circuit card as the general purpose I/O interface.

2.11 Bootstrap Loading

The ROLM 1650 computer is boot loaded by setting the front panel switches to the device number of the device which contains the program to be loaded and depressing the BOOT LOAD switch. The following device codes are supported:

10 (octal)	Teletype paper tape reader
12	High speed paper tape reader
20	Fixed head disk
22	9-track magnetic tape
33	Moving head disk
77	Does not boot load -- instead performs a simple memory diagnostic
All others	High speed paper tape

When the BOOT LOAD switch is depressed, a loader appropriate for the indicated device is automatically read from a ROM in the 1650, placed in the upper 127 locations of memory, and executed.

In order to provide for boot loading from the cartridge tape drive, special boot loading circuitry was included as part of the general purpose I/O board. From the 1650 computer's point of view it behaves as device 12, a high speed paper tape reader. The actual implementation consists of a 512-word, 8-bit-wide ROM and an associated address register. The CLEAR and IORST lines clear the ROM address register to zero. A DIA reads the addressed ROM word into the least significant 8 bits of a register and increments the ROM address register. In this way up to 256 words (16-bit words) can be read into memory. A START pulse sets BUSY for 100 nsec and then BUSY is cleared and DONE is set. This allows I/O skip instructions to operate properly, as these are used by the paper tape reader boot load software. However, no interrupt is generated and there is no interrupt mask.

The program from device 12 (the ROM) is loaded beginning at location 100_8. The 1650 then jumps to its starting location. The program provided is one which reads the first file from the cartridge tape drive and places it in upper memory beginning at location 77500_8. It then jumps to this program, which must be the cartridge loader program. This has the name GTFEXEC on the TEU software development facility file system.

GTFEXEC first prompts on the terminal with "FILE?" The operator responds with "A", "B", "C", "D", etc., depending on which file is to be loaded. "A" corresponds to the first file on the cartridge following the cartridge loader program. The loader program will skip to the appropriate file and begin loading it starting at location zero in memory. The file must be short enough that it does not overwrite the cartridge loader program in upper memory. When end-of-file is detected, the loader jumps to whatever address is stored in location two of memory. This location must be initialized to the start address of the system when the binary file is created prior to being written to the cartridge.

If a cartridge read error is encountered during the loading process, the loader will halt and display an error condition code on the 1650's front panel display.

Bootable cartridge tapes are created on the TEU software development facility by using the CARTCREATE program. This prompts for the names of files to be written to the cartridge from the disk. To create a bootable tape these files must be executable binary (so-called "save") files, with a .SV suffix on the file name. The first file copied to the cartridge must be GTFEXEC.SV. Any other executable files may follow, and files may be repeated for the sake of redundancy. Before being written to cartridge tape, files must have location two set to the starting location of the program. This may be done by using the octal editor program or the patching facility provided on the software development facility.

REFERENCES

1. M.L. Wood, "TCAS II ATCRBS Surveillance Algorithms," Project Report ATC-131, Lincoln Laboratory, M.I.T., (9 November 1984), DOT/FAA/PM-84/19.

2. J.D. Welch, V.A. Orlando, "Traffic Alert and Collision Avoidance System (TCAS): A Functional Overview of Minimum TCAS-II," Project Report ATC-119, Lincoln Laboratory, M.I.T., (8 April 1983), DOT/FAA/PM-83/10.

3. W.H. Harman, et. al., "Active BCAS: Design and Validation of the Surveillance Subsystem," Project Report ATC-103, Lincoln Laboratory, M.I.T., (17 December 1980), FAA-RD-80-134.

4. W.H. Harman, R.S. Kennedy, "TCAS-II: Design and Validation of the High-Traffic-Density Surveillance Subsystem," Project Report ATC-126, Lincoln Laboratory, M.I.T., (7 June 1984), DOT/FAA/PM-84/5.

5. P.H. Mann, "Simulation of Surveillance Processing Algorithm Proposed for the DABS Mode of BCAS," Project Report ATC-82, Lincoln Laboratory, M.I.T., (16 February 1978), FAA-RD-77-138.

6. "Minimum Operational Performance Standards for Traffic Alert and Collision Avoidance System (TCAS) Airborne Equipment," RTCA/DO-185, Radio Technical Commission for Aeronautics (September 1983).

7. J. Grupe, et. al., "Active BCAS Detailed Collision Avoidance Algorithms," The MITRE Corporation, McLean, Virginia (October 1980), MTR-80W286.

8. "U.S. National Aviation Standard for the Traffic Alert and Collision Avoidance System II," Federal Aviation Administration (19 June 1984), DOT/FAA Order 6367.1.

9. R.G. Sandholm, "Effects of Interference on the Performance of a Minimum TCAS II," ATC Project Report ATC-132, Lincoln Laboratory, M.I.T., (5 September 1985), DOT/FAA/PM-84/33.

10. J.L. Gertz, "Fundamentals of Mode S Parity Coding," Project Report ATC 117, Lincoln Laboratory, M.I.T., (2 April 1984), DOT/FAA/PM-83/6.

11. W.H. Harman, "Effects of RF Power Deviations on BCAS Link Reliability," Project Report ATC-76, Lincoln Laboratory, M.I.T., (7 June 1977), DOT/FAA-RD-77-78.

12. A.C. Drumm, W.S. Heath, J.A. Richardson, "Airborne Intelligent Display (AID) Phase I Software Description," Project Report ATC-123, Lincoln Laboratory, M.I.T., (24 October 1983), DOT/FAA/PM-83/30.